



TEAMCENTER

**Teamcenter
Automotive Edition**

Teamcenter 2412

Unpublished work. © 2025 Siemens

This Documentation contains trade secrets or otherwise confidential information owned by Siemens Industry Software Inc. or its affiliates (collectively, "Siemens"), or its licensors. Access to and use of this Documentation is strictly limited as set forth in Customer's applicable agreement(s) with Siemens. This Documentation may not be copied, distributed, or otherwise disclosed by Customer without the express written permission of Siemens, and may not be used in any way not expressly authorized by Siemens.

This Documentation is for information and instruction purposes. Siemens reserves the right to make changes in specifications and other information contained in this Documentation without prior notice, and the reader should, in all cases, consult Siemens to determine whether any changes have been made.

No representation or other affirmation of fact contained in this Documentation shall be deemed to be a warranty or give rise to any liability of Siemens whatsoever.

If you have a signed license agreement with Siemens for the product with which this Documentation will be used, your use of this Documentation is subject to the scope of license and the software protection and security provisions of that agreement. If you do not have such a signed license agreement, your use is subject to the Siemens Universal Customer Agreement, which may be viewed at <https://www.sw.siemens.com/en-US/sw-terms/base/uca/>, as supplemented by the product specific terms which may be viewed at <https://www.sw.siemens.com/en-US/sw-terms/supplements/>.

SIEMENS MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS DOCUMENTATION INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF INTELLECTUAL PROPERTY. SIEMENS SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL OR PUNITIVE DAMAGES, LOST DATA OR PROFITS, EVEN IF SUCH DAMAGES WERE FORESEEABLE, ARISING OUT OF OR RELATED TO THIS DOCUMENTATION OR THE INFORMATION CONTAINED IN IT, EVEN IF SIEMENS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TRADEMARKS: The trademarks, logos, and service marks (collectively, "Marks") used herein are the property of Siemens or other parties. No one is permitted to use these Marks without the prior written consent of Siemens or the owner of the Marks, as applicable. The use herein of third party Marks is not an attempt to indicate Siemens as a source of a product, but is intended to indicate a product from, or associated with, a particular third party. A list of Siemens' Marks may be viewed at: www.plm.automation.siemens.com/global/en/legal/trademarks.html. The registered trademark Linux® is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a world-wide basis.

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

Getting started with Teamcenter Automotive Edition and GM Overlay

Getting started with Teamcenter Automotive Edition and GM Overlay	1-1
Getting started with Teamcenter Automotive Edition	1-1
Overview of getting started with Teamcenter Automotive Edition	1-1
Teamcenter Automotive Edition prerequisites	1-1
Install Teamcenter Automotive Edition using Teamcenter Environment Manager	1-1
Install Teamcenter Automotive Edition using Deployment Center	1-2
Enabling Teamcenter Automotive Edition	1-2
Using Teamcenter Automotive Edition	1-3
Getting started with GM Overlay	1-6
What is GM Overlay?	1-6
GM Overlay prerequisites	1-6
Install GM Overlay using Teamcenter Environment Manager	1-7
Install GM Overlay using Deployment Center	1-7
Enabling GM Overlay	1-8
Using GM Overlay	1-9

Administering Teamcenter Automotive Edition

Overview of administering Teamcenter Automotive Edition	2-1
Configuring Teamcenter Automotive Edition–GM Overlay features	2-1
Localizing Teamcenter Automotive Edition–GM Overlay	2-1
Using Teamcenter Automotive Edition–GM Overlay utilities	2-2
Using GM Overlay subscription events	2-2
Configuring business rules	2-3
Overview of configuring business rules	2-3
Naming rules	2-4
Extension rules	2-4
Deep copy rules	2-6
Compound properties	2-6
Generic Relationship Management rules	2-7
Display rules	2-7
Property rules	2-8
Configuring lists of values	2-8
Modifying preference files	2-8
Overview of modifying preference files	2-8
Add a new GM Overlay preference	2-9
Modify GM Overlay preferences	2-9
Override Teamcenter preference values	2-9
Using workflow handlers	2-10
Overview of using workflow handlers	2-10
Action handlers	2-10
Rule handlers	2-10
Configuring Repeatable Digital Validation	2-11

Configuring tessellation	2-11
Overview of configuring tessellation	2-11
Background tessellation	2-12
Foreground tessellation	2-12
Placement of the tessellation handler	2-13
Batch tessellation	2-13
Configuring the AliasStudio encapsulation	2-15
Tasks to be complete before using the AliasStudio encapsulation	2-15
Enable the AliasStudio launcher	2-15
Configure the default AliasStudio encapsulation tool	2-15
Edit the alias.bat file	2-16
Start the AliasStudio application from the rich client interface	2-16
Setting the environment variables for the AliasStudio encapsulation	2-16
Business rules	2-17
AliasStudio encapsulation preferences	2-17
AliasStudio encapsulation utilities	2-18
Configuring materials engineering data management	2-18
Overview of configuring materials engineering data management	2-18
Material form attribute synchronization – Teamcenter to NX	2-18
Materials engineering data management utilities	2-21
Materials engineering data management preferences	2-21

Using GM Overlay preferences [A-1](#)

1. Getting started with Teamcenter Automotive Edition and GM Overlay

Getting started with Teamcenter Automotive Edition and GM Overlay

Teamcenter Automotive Edition is a component of an industry-focused solution intended to meet the needs of design and manufacturing organizations within the automotive industry. Teamcenter Automotive Edition provides the GM Overlay installation that serves as a base implementation for automotive industry users. Other customer-specific implementations can then be layered on top of the Teamcenter Automotive Edition overlay.

This guide provides an overview of the commercial Teamcenter Automotive Edition and GM Overlay features and how these are used by manufacturers and suppliers.

This guide is for persons responsible for installing and configuring Teamcenter Automotive Edition and GM Overlay features and for those who use these features, specifically General Motors, its suppliers, and partners.

Getting started with Teamcenter Automotive Edition

Overview of getting started with Teamcenter Automotive Edition

Teamcenter Automotive Edition adds data model extensions, such as tool types and database utilities necessary to support the automotive industry.

Teamcenter Automotive Edition prerequisites

The Automotive Edition installation requires:

- Teamcenter database
- Base Teamcenter installation

For information about adding a database and these prerequisites, see *Teamcenter Installation on Linux Using TEM* or *Teamcenter Installation on Windows Using TEM*.

Install Teamcenter Automotive Edition using Teamcenter Environment Manager


Install the Teamcenter Automotive Edition using Teamcenter Environment Manager.

Feature	Description
Teamcenter Automotive Edition	Installs the optional Teamcenter Automotive Edition feature.

Install Teamcenter Automotive Edition using Deployment Center

You can install Teamcenter Automotive Edition using Deployment Center as follows:

Procedure

1. Log on to Deployment Center and select the environment to which you want to add Teamcenter Automotive Edition.
2. Go to the **Applications** task. Click **Add or Remove Selected Applications** .
3. In the **Available Applications** panel, use the web browser search to find the **Teamcenter Automotive Edition** application. Select the application, and then click **Update Selected Applications**.

Deployment Center automatically selects any additional dependent applications.

4. In the **Selected Components** list, note any components whose configuration status is not **100%**. Select each incomplete component, enter required parameters, and save component settings until all components in the environment show a configuration status of **100%**.

When all components are fully configured, the **Deploy** task is enabled.

5. Go to the **Deploy** task. Click **Generate Install Scripts** to generate deployment scripts you will use to update affected machines.

When script generation is complete, note any special instructions in the **Deploy Instructions** panel.

6. Locate deployment scripts, copy each script to its target machine, and then run each script on its target machine.

For more information about running deployment scripts, see *Deployment Center — Usage*.

Enabling Teamcenter Automotive Edition

To enable Teamcenter Automotive Edition functionality, you must install the required license files that allow you to run Teamcenter Automotive Edition. Siemens Digital Industries Software products are licensed using the Macrovision FLEXIm licensing system. For licensing information, contact your Siemens Digital Industries Software representative.

The **TCAE_installation_level** preference is automatically inserted into the database and set to **2** upon installation of Teamcenter Automotive Edition.

For information about installing a license file, see *Teamcenter Installation on Linux Using TEM* or *Teamcenter Installation on Windows Using TEM*.

Using Teamcenter Automotive Edition

Overview of using Teamcenter Automotive Edition

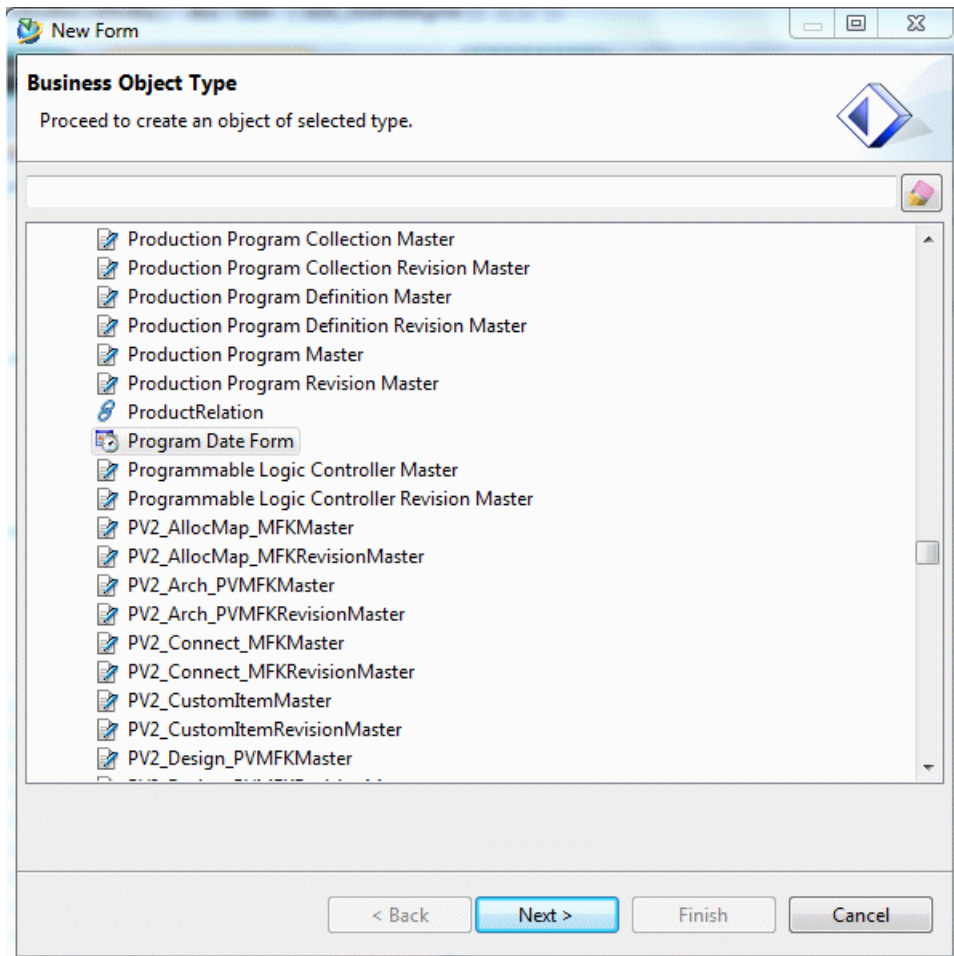
Teamcenter Automotive Edition provides features in the rich client for these applications:

- My Teamcenter
- Database Utilities (an administrator application)

Using New Form wizard in My Teamcenter

Teamcenter Automotive Edition uses the New Form wizard in My Teamcenter.

The New Form wizard (**File→New→Form**) is available in My Teamcenter when you use the Teamcenter Automotive Edition. It populates the form name using the Teamcenter Automotive Edition naming conventions.



The program date form, which is used for project and program management purposes to set milestones, allows you to prepopulate the fields of the form during creation.

The **Relation** list allows you to select a relation based on business rules specific to Teamcenter Automotive Edition.

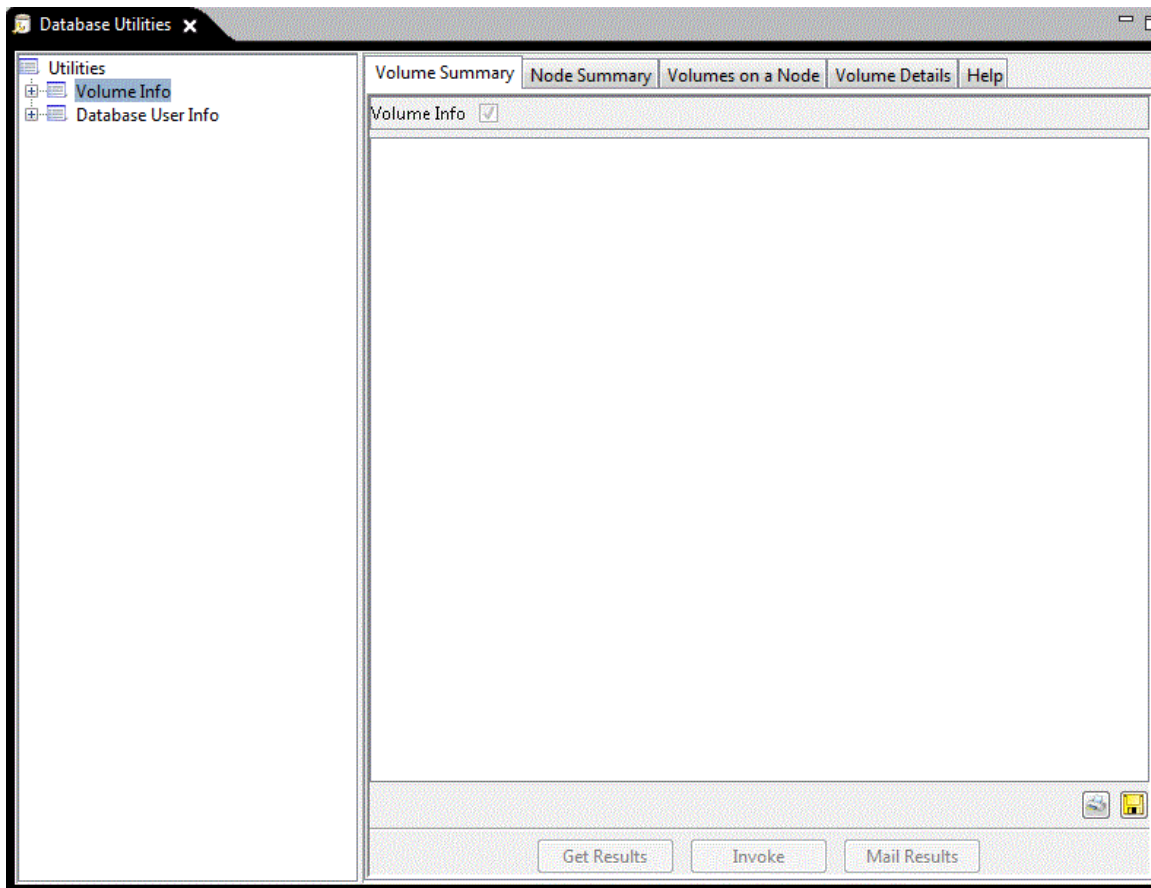
Using the Database Utilities application

Note:

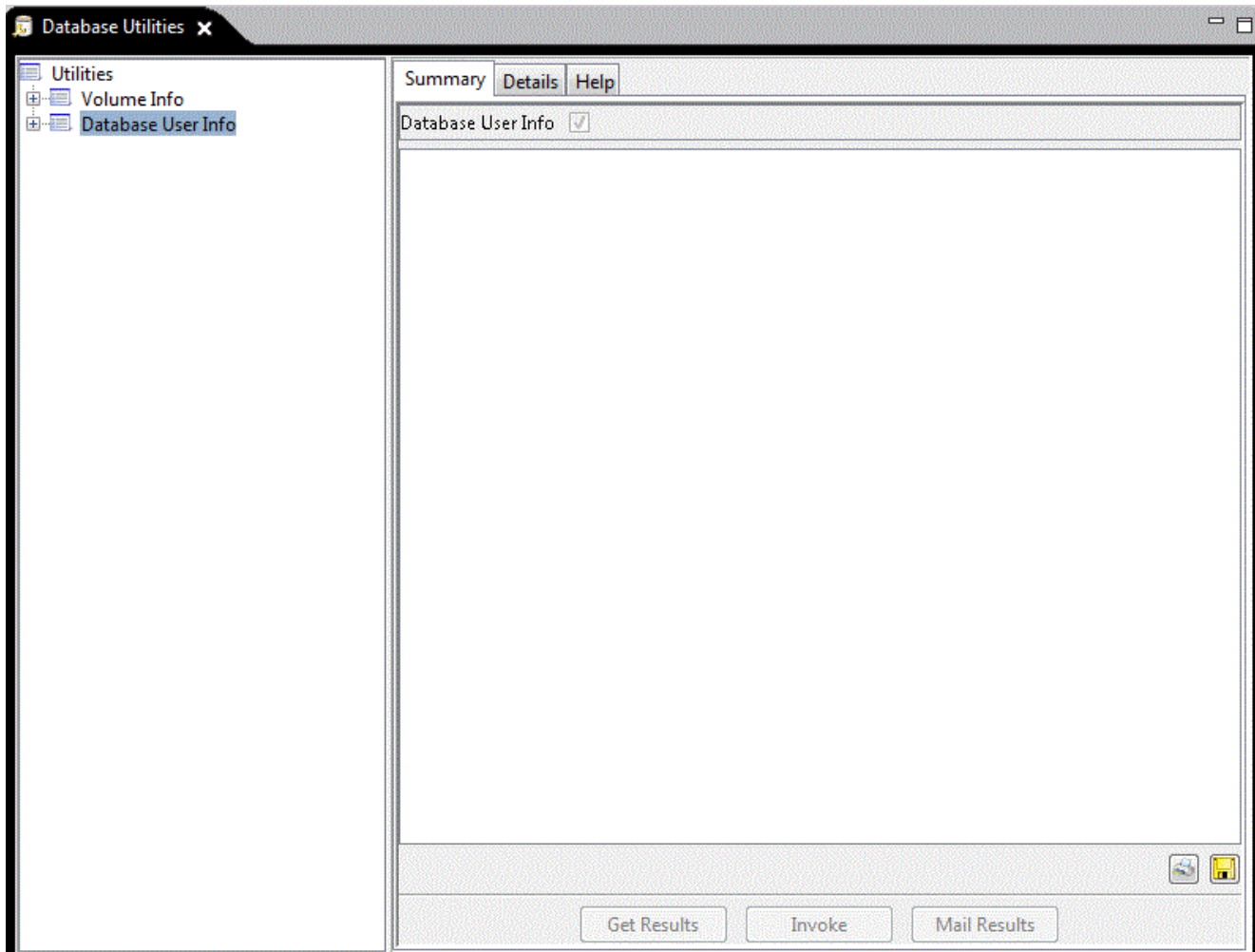
Because of security reasons, the Database Utilities application can only be used in a two-tier environment.

Select the Database Utilities application to access:

- Teamcenter volume information (for example, on which node the volume exists, node name, volume name)



- Teamcenter database user information (for example, number of users connected to the database)



Getting started with GM Overlay

What is GM Overlay?

The GM Overlay adds data model extensions, such as business rules and part and tool item types, specifically to support General Motors, its suppliers, and partners.

GM Overlay prerequisites

The GM Overlay installation requires:

- Teamcenter database
- Base Teamcenter installation
- Installed Teamcenter Automotive Edition feature

For information about adding a database and these prerequisites, see *Teamcenter Installation on Linux Using TEM* or *Teamcenter Installation on Windows Using TEM*.

Install GM Overlay using Teamcenter Environment Manager

Install the GM Overlay using Teamcenter Environment Manager.

Feature	Description
GM Overlay	Installs the GM Overlay feature. Installing GM Overlay requires that you also install Teamcenter Automotive Edition.

Optionally, you can install two other Teamcenter features.

Feature	Description
GM PAD/TWP Customization	Installs additional data types for GM PAD/TWP Customization. This feature requires the Teamcenter Foundation , Teamcenter Automotive Edition , GM Overlay , Customization for eMServer Integration , and PAD/TWP Customization features.
Configure AutoCAD Integration for GM Overlay	Configures AutoCAD Integration/AutoCAD Manager to operate in a Teamcenter Automotive Edition–GM Overlay environment. This feature requires the Teamcenter Foundation , Teamcenter Automotive Edition , and GM Overlay features.

For information about installing these features on the server, see *Teamcenter Installation on Linux Using TEM* or *Teamcenter Installation on Windows Using TEM*.


For information about installing the GM Overlay feature on the client, see *Teamcenter Rich Client Installation on Linux* or *Teamcenter Rich Client Installation on Windows*.

Install GM Overlay using Deployment Center

You can install GM Overlay using Deployment Center as follows:

Procedure

1. Log on to Deployment Center and select the environment to which you want to add GM Overlay.

2. Go to the **Applications** task. Click **Add or Remove Selected Applications** .
3. In the **Available Applications** panel, use the web browser search to find the following applications: **Teamcenter Automotive Edition** application.

- **GM Overlay**

- (Optional) **GM PAD/TWP Customization**

Installs additional data types for GM PAD/TWP customization.

- (Optional) **Configure AutoCAD Integration for GM Overlay**

Configures AutoCAD Integration/AutoCAD Manager to operate in a Teamcenter Automotive Edition–GM Overlay environment. This application requires that the AutoCAD application is installed.

Select the application, and then click **Update Selected Applications**.

Deployment Center automatically selects any additional dependent applications.

4. In the **Selected Components** list, note any components whose configuration status is not **100%**. Select each incomplete component, enter required parameters, and save component settings until all components in the environment show a configuration status of **100%**.

When all components are fully configured, the **Deploy** task is enabled.

5. Go to the **Deploy** task. Click **Generate Install Scripts** to generate deployment scripts you will use to update affected machines.

When script generation is complete, note any special instructions in the **Deploy Instructions** panel.

6. Locate deployment scripts, copy each script to its target machine, and then run each script on its target machine.

For more information about running deployment scripts, see *Deployment Center — Usage*.

Enabling GM Overlay

To enable the GM Overlay functionality, you must install the required license files that allow you to run GM Overlay. Siemens Digital Industries Software products are licensed using the Macrovision FLEXlm licensing system. For licensing information, contact your Siemens Digital Industries Software representative.

The **TCAE_installation_level** preference is automatically inserted into the database and set to **2** upon installation of Teamcenter Automotive Edition. After installing the GM Overlay feature, this preference

is automatically set to **3** to indicate you have licenses for both Teamcenter Automotive Edition and GM Overlay features.

For information about installing a license file, see *Teamcenter Installation on Linux Using TEM* or *Teamcenter Installation on Windows Using TEM*.

Using GM Overlay

Overview of using GM Overlay

GM Overlay provides features in the rich client for these applications:

- My Teamcenter
- Structure Manager

GM Overlay features in My Teamcenter

Overview of GM Overlay features in My Teamcenter

GM Overlay uses the following features in My Teamcenter:

- New Item wizard
- New Dataset wizard
- **GM Task Queries** dialog box
- **File Checker** command
- **Site Specific** command
- **Set Dataset Name** command
- **GM Task Report** command

New Item wizard

The New Item wizard (**File**→**New**→**Item**) is available in My Teamcenter when you use GM Overlay. It lists only the items that you can create using GM Overlay, for example:

- **CORP_Part**

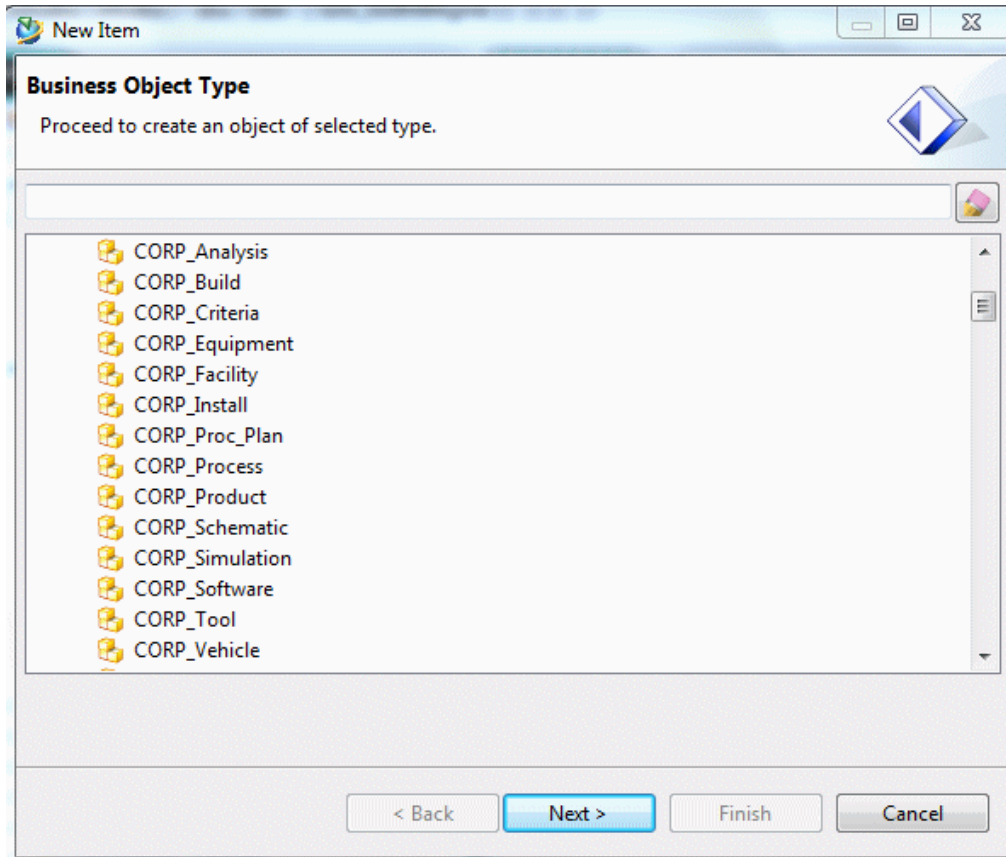
Identifies a product, component, or commodity identified by a unique number.

- **CORP_Process**

Represents a process used in the manufacturing environment.

- **CORP_Product**

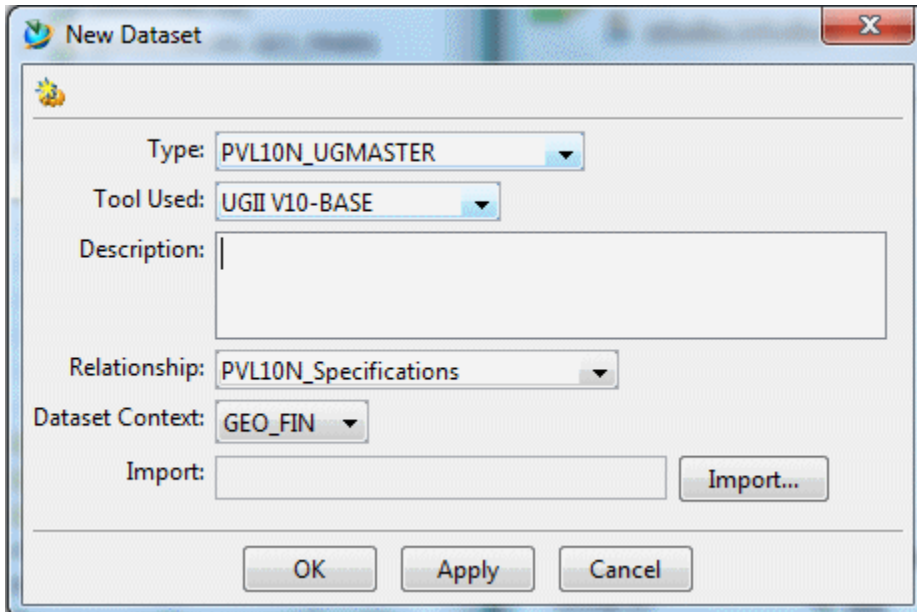
Identifies products that are not vehicles, but can be nested with the **CORP_Vehicle** item.



The New Item wizard automatically assigns the item ID and revision ID using the GM Overlay naming conventions.

New Dataset wizard

The New Dataset wizard (**File**→**New**→**Dataset**) is available in My Teamcenter when you use GM Overlay. It creates a new dataset that conforms to the GM Overlay conventions.

**Note:**

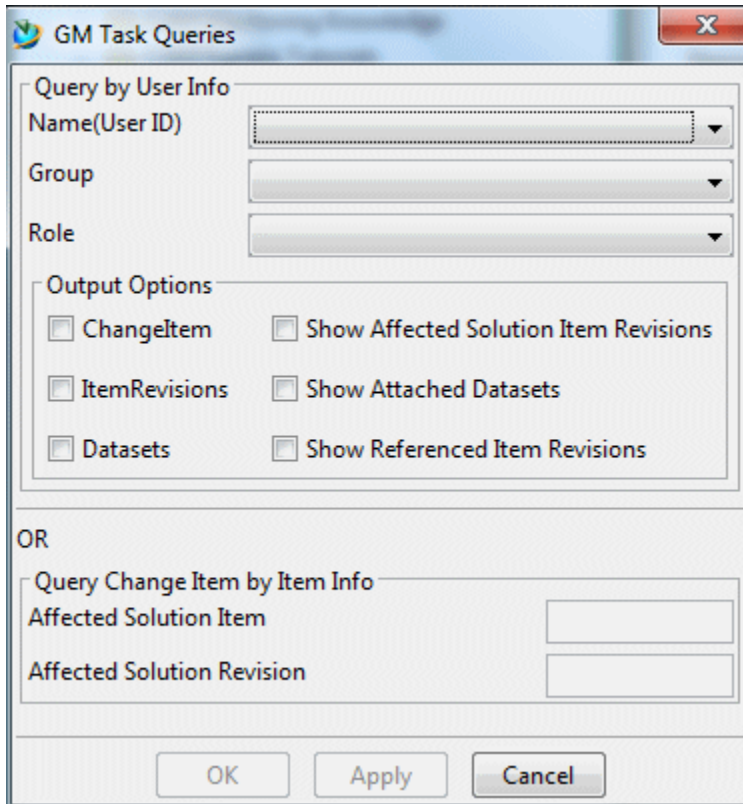
You can import any named references using the **Import** button.

The name of the new dataset is automatically generated using the GM Overlay naming conventions.

GM Task Queries dialog box

The **GM Task Queries** dialog box is available in My Teamcenter when you use GM Overlay. You can use it to query all the in-process change items, item revisions, or datasets that belong to a specific user ID, role ID, or group. You can also use this dialog box to query for all the in-process change items associated with a particular solution item or solution revision. This dialog box allows you to configure a report showing the related objects based on the selected item, item revision, or dataset.

Choose **Tools**→**GM Task Report** to display the **GM Task Queries** dialog box.

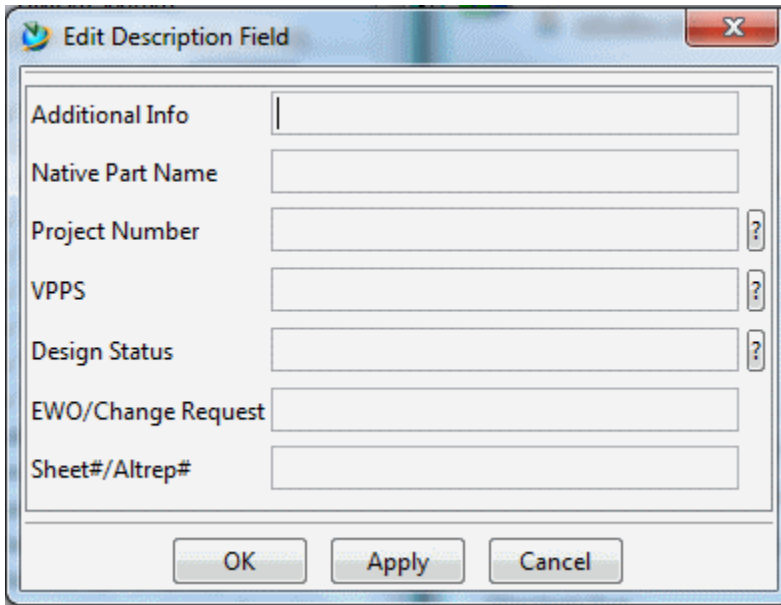


File Checker command

Choose **Tools**→**File Checker** to check whether the dataset complies with the GM Overlay conventions.

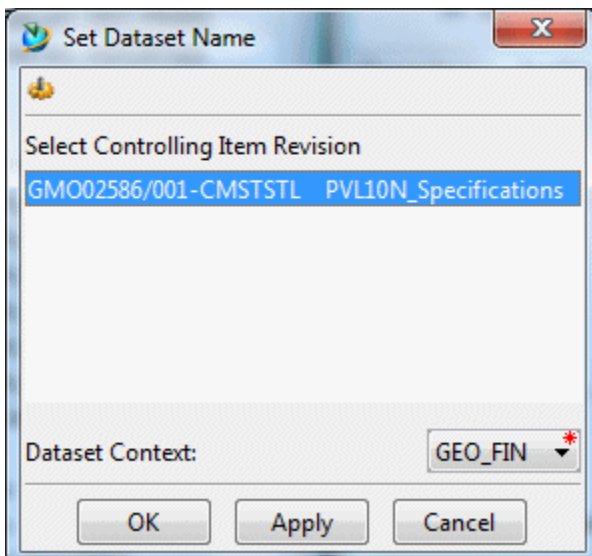
Site Specific command

Choose **Site Specific**→**Edit Description** to display the **Edit Description Field** dialog box.



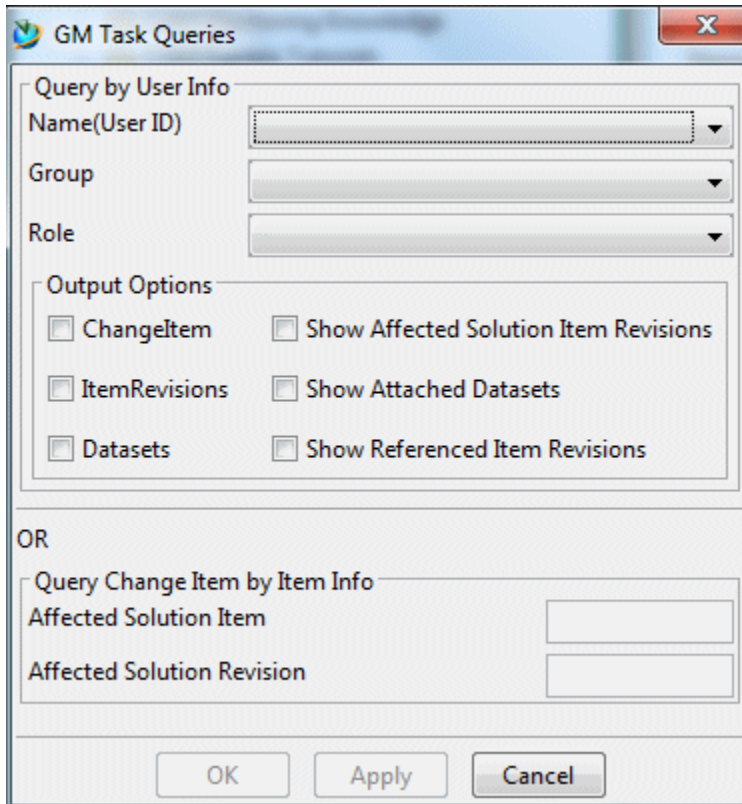
Set Dataset Name command

Choose **File**→**Set Dataset Name** to display the **Set Dataset Name** dialog box.



GM Task Report command

Choose **Tools**→**GM Task Report** to display the **GM Task Queries** dialog box.



GM Overlay features in Structure Manager

The different GM Overlay features in Structure Manager

There are two GM Overlay features available in Structure Manager:

- Export NX assembly
- Send to external system

In GM Overlay, when you open a BOM structure in Structure Manager for the first time, you get a **java.lang.NullPointerException** error because GM Overlay sets the default revision rule to **LatestReleased**, which is not part of the default configuration.

Either create a revision rule called **LatestReleased** or set the **TC_config_rule_name** preference to any valid existing revision rule (for example, **LatestWorking**).

Export NX assembly

A feature in Structure Manager for Teamcenter Automotive Edition–GM Overlay allows users to export any NX® assembly from a Teamcenter database to local files by choosing **Tools→Export NX Assembly**. This feature exports the part/assembly without going through Teamcenter Integration for NX.

Send to external system

Choose **Tools**→**Send to External System** to have the system send calibration information generated from the selected installation assembly to a URL (defined using the **TC_external_system_url** preference). An external system can then access this information from a Web browser.

2. Administering Teamcenter Automotive Edition

Overview of administering Teamcenter Automotive Edition

After you install the Teamcenter Automotive Edition and GM Overlay features, you can:

- Configure Teamcenter Automotive Edition–GM Overlay features.
- Localize Teamcenter Automotive Edition–GM Overlay.
- Use Teamcenter Automotive Edition–GM Overlay utilities.
- Configure business rules.
- Configure lists of values.
- Modify preference files.
- Use workflow handlers.
- Configure Repeatable Digital Validation.
- Configure tessellation.
- Configure the AliasStudio encapsulation.
- Configure materials engineering data management.

Configuring Teamcenter Automotive Edition–GM Overlay features

After installing the Teamcenter Automotive Edition and GM Overlay features on the server and the **GM Overlay for Rich Client** feature on two-tier rich client or four-tier rich client hosts, you must complete the required postinstallation and configuration steps.

For more information about postinstallation and configuration steps for the **GM Overlay for Rich Client** feature, see *Teamcenter Rich Client Installation on Linux* or *Teamcenter Rich Client Installation on Windows*.

Localizing Teamcenter Automotive Edition–GM Overlay

Using the Business Modeler IDE, you can localize content for the following elements:

- Type names
- Relation names
- Property names
- LOV values and LOV value descriptions

For more information about localizing Teamcenter Automotive Edition–GM Overlay using the Business Modeler IDE, see *Teamcenter Localization*.

Using Teamcenter Automotive Edition–GM Overlay utilities

You can use Teamcenter Automotive Edition–GM Overlay utilities to administer the Teamcenter Automotive Edition and/or Teamcenter Automotive Edition–GM Overlay.

For detailed information and syntax for these utilities, see *Teamcenter Utilities*.

Using GM Overlay subscription events

GM Overlay installs the following subscription events allowing you to subscribe to an event and receive notifications when the event occurs. These events are controlled as per the following:

Event type	Controlled via	Configurations
IRCreated	GMO_notify_on_ir- _creation extension	None
DrawingCreated- InIR	GMO_GRM_notify_on- _creation extension	Update the preference GMO_notify_item_on_dataset_attachment
ImageCreatedInI R	GMO_GRM_notify_on- _creation extension	Update the preference GMO_notify_item_on_dataset_attachment
IRReleasedWith- StatusAny	GMIMAN- invoke-subscription-event- on-item workflow handler	Attach the handler to a workflow
IRReleasedWith- StatusAnyExcept X		
IRReleasedWith- Status0		
IRReleasedWith- Status1		

Event type	Controlled via	Configurations
IRReleasedWith-Status2		
IRReleasedWith-Status3		
IRReleasedWith-StatusX		

The events are attached to the following objects:

- CORP_Part
- CORP_Criteria
- CORP_Install
- CORP_Proc_Plan
- CORP_Vehicle
- CORP_Product
- CORP_Analysis
- CORP_Equipment
- CORP_Facility
- CORP_Process
- CORP_Simulation
- CORP_Tool
- CORP_Schematic
- CORP_Build Item types

Configuring business rules

Overview of configuring business rules

Teamcenter Automotive Edition–GM Overlay provides several types of configurable business rules:

- Naming rules
- Extension rules
- Deep copy rules
- Compound property rules
- Generic Relationship Management (GRM) rules
- Display rules
- Property rules

For more information about configuring business rules, see *BMIDE for Data Model Design*.

Naming rules

Naming rules define the data entry format for a business object property. After you create a naming rule, you must attach it to the business object property. For example, you can create a rule to define how new item business objects are named and attach it to the **item_id** property.

The GM Overlay feature has several naming rules that are imposed on the **Item ID**, **ItemRevision ID**, **Item Name**, and **Baseline Revision ID** attributes.

The following information pertains to GM Overlay naming rules:

- Item IDs must be eight characters. The auto assign feature creates items with alphanumeric item IDs, beginning with three letters; the remaining characters are numbers.
- Other patterns listed by the attached item ID rule are allowed.
- There are 15 item types. **CORP_Tool** and **CORP_Facility** use a different item ID rule that allows 17 characters, which can be alphanumeric. Hyphens and underscores can be used except as the first or last character. These item types can also be created using the eight-character alphanumeric rule.

The remaining item types (**CORP_Analysis**, **CORP_Build**, **CORP_Criteria**, **CORP_Equipment**, **CORP_Install**, **CORP_Part**, **CORP_Proc_Plan**, **CORP_Process**, **CORP_Product**, **CORP_Schematic**, **CORP_Simulation**, **CORP_Software**, and **CORP_Vehicle**) allow creation of items with an item ID of eight characters. These item IDs can be a numeric ID of eight characters or an alphanumeric item ID of eight characters beginning with three letters followed by five numbers. Many other patterns listed by the attached naming rule are allowed.

- The item revision ID uses a three-digit pattern for all item revisions in GM Overlay. The initial value for the **ItemRevision ID** attribute is 001; the maximum value is 999. An assign button generates the item revision ID.
- The item name entered for all items in GM Overlay is converted to uppercase characters.
- The baseline suffix for all items starts with **.0001**; the maximum value is **.0099**.

For more information about naming rules, see *BMIDE for Data Model Design*.

Extension rules

Extension rules define the actions that need to be done in different time phases for different operations, such as create, save as, and delete. The GM Overlay-specific extension rules are installed in the database when you install the GM Overlay feature.

Extension rule	Description	Where used
checkLatestReleased	Ensures that during creation of a revision there are no more than the configured number of allowable working revision.	Used as a precondition on all CORP_ item revisions.
createObjects	Creates specified objects and attaches them with the specified relation to the item type for which this method is configured.	Used as a postaction on the following item types: <ul style="list-style-type: none"> • CORP_Analysis • CORP_Build • CORP_Install • CORP_Part • CORP_Product • CORP_Software • CORP_Tool • CORP_Vehicle
createCAEObjects	Creates specified CAE objects and attaches them with the specified relation to the item type for which this method is configured.	Used on postaction of create on CORP_Analysis revision.
GMO_notify_on_IR_creation	Notifies subscribers when an item revision is created.	Used on postaction of revise on all CORP_ item revisions.
GMO_check_baseline_preaction	Performs a preaction check, whether the base IR dataset is filechecked or not, before creating the baseline.	Used on preconditions of Create Baseline Revision for the following item types: <ul style="list-style-type: none"> • CORP_Analysis • CORP_Criteria • CORP_Equipment • CORP_Facility • CORP_Install • CORP_Part • CORP_Proc_Plan • CORP_Process • CORP_Product • CORP_Simulation • CORP_Tool • CORP_Vehicle
GMO_add_status_list_prop	Adds the status_list property value on UGMaster , UGPart , UGAltrep , and UGScenario .	Used as the base action on the status_list property of UGAltrep , UGMaster , UGPart , and UGScenario .

Extension rule	Description	Where used
GMO_user_new_form_name	Generates the new form name using the GM Overlay naming convention.	Used on the property object_name of form type.
GMO_user_new_dataset_name	Generates the new dataset name using the GM Overlay naming convention.	Used on the property object_name of dataset type.

For more information about extension rules, see *BMIDE for Data Model Design*.

Deep copy rules

Deep copy rules provide a method to specify which objects are copied from one revision to the next.

Deep copy rules also provide a method to set relation/type objects that you can copy from one revision to another using the **Save As** or **Revise** commands. GM Overlay-specific deep copy rules are installed in the database as part of the GM Overlay installation. For example, performing a **Save As** operation on a **CORP_Part** revision to a new **CORP_Part** item copies forward all attachments under **CORP_Part** revision with a **Specification** relation to the new **CORP_Part** revision as a new object.

For more information about deep copy rules, see *BMIDE for Data Model Design*.

Compound properties

A *compound property* is a property that can be displayed on one object although it is defined and resides on a different object.

GM Overlay-specific compound properties are installed in the database as part of the GM Overlay installation. The following are examples of compound properties:

- **dls**, a **CORP_PartRevMaster** attribute, is displayed as a compound property of **CORP_Part** revision.
- **mass**, a **CORP_PartRevMaster** attribute, is displayed as a compound property of **CORP_Part** revision.
- **gauge**, a **CORP_PartRevMaster** attribute, is displayed as a compound property of **CORP_Part** revision.
- **change_desc**, a **CORP_PartRevMaster** attribute, is displayed as an compound property of **CORP_Part** revision.
- **tool_level**, a **CORP_ToolMaster** attribute, is displayed as a compound property of **CORP_Tool** revision.
- **local_tool_id**, a **CORP_ToolMaster** attribute, is displayed as a compound property of **CORP_Tool** revision.

For more information about compound properties, *BMIDE for Data Model Design*.

Generic Relationship Management rules

Use Generic Relationship Management (GRM) rules to apply constraints on objects based on the relationship between primary and secondary objects.

GM Overlay-specific GRM rules are installed in the database as part of the GM Overlay installation. The following are examples of GRM rules:

- Creating more than one UGMaster dataset with a **Specification** relation under any item revision is restricted using a GRM rule on an item revision.
- Only one Program Date Form is allowed under a **CORP_Part** revision and with a **CORP_Schedule** relation. This is restricted using a GRM rule on a **CORP_Part** revision.
- Creating UGPart datasets under **CORP_Facility** revision is restricted using a GRM rule implemented on a **CORP_Facility** revision.
- Creating more than one **ALIAS_PROJECT** dataset with a **Specification** relation under a **CORP_Criteria** revision is restricted using a GRM rule implemented on a **CORP_Criteria** revision.
- Creating more than one **DesignSpace** dataset with a **Manifestation** relation under a **CORP_Tool** revision is restricted using a GRM rule implemented on a **CORP_Tool** revision.

For more information about GRM rules, see *BMIDE for Data Model Design*.

Display rules

By default, all business objects can be created by all users. Business object display rules suppress the display of objects in the creation dialog boxes, preventing the users in a specific group or role from creating objects of those types.

GM Overlay-specific type display rules are installed in the database as part of the GM Overlay installation. Examples of type display rules used by the GM Overlay feature include:

- Items of type **Item**, **Document**, **EngChange**, and **Functionality** are hidden from the **Create Item** dialog box.
- **Mail Folder** and **Newstuff Folder** are hidden from the folder creation dialog box.
- All the item master forms and item revision master forms are hidden from the **New Form** dialog box.

For more information about type display rules, see *BMIDE for Data Model Design*.

Property rules

Using property rules, you can control whether a property is modifiable, visible, enabled, or required. In addition, you can set an initial value for a property and define complex properties comprised of a combination of properties and constant strings that assign a value to the property.

GM Overlay-specific property rules are installed in the database as part of the GM Overlay installation. Examples of property rules used by the GM Overlay feature include:

- The **ucl** attribute on the **CORP_Criteria** revision master is modifiable using property rules.

Attributes (**mass**, **area**, **density**, **volume**, **material**, **gauge**) on the **CORP_Install** revision master and **CORP_Part** revision master are modifiable using property rules.

The **dls** attribute on the **CORP_Install** revision master and **CORP_Part** revision master is made visible using property rules.

The **product** and **year** attributes on the **CORP_Product** revision master and **CORP_Vehicle** revision master are modifiable and enabled using property rules.

For more information about property rules, see *BMIDE for Data Model Design*.

Configuring lists of values

Lists of values (LOVs) are used to ensure consistent data entry in Teamcenter. The list entries are created either by referencing existing data or by entering customized site data. Once created, the LOV is implemented throughout the interface by attaching the LOV to one or more properties. Creating LOVs and implementing them throughout Teamcenter can greatly improve productivity at your site and help prevent incorrect user entries. The GM Overlay-specific LOVs are installed in the database as part of the Teamcenter Automotive Edition–GM Overlay configuration.

For more information about LOVs, see *BMIDE for Data Model Design*.

Modifying preference files

Overview of modifying preference files

You can modify the GM Overlay and CMTPADTWP preference files to add, modify, or delete preferences.

- GM Overlay-specific preference files are located in the *Teamcenter-install-directory*\gmo directory.
- CMTPADTWP-specific preference files are located in the *Teamcenter-install-directory*\cmtpadtwp directory.

Note:

The following procedures are specific to GM Overlay preference files. However, the same procedures apply when modifying the CMTPADTWP preference files.

Add a new GM Overlay preference

To add a new GM Overlay preference in the **gmo_preferences_override.xml** file, the preference name must be new and must not exist in the **tc_preferences.xml** file. Follow the naming convention of the preference prefixed with **GMO**, for example, **GMO_test=**:

```
GMO_test=
First
Second
```

Modify GM Overlay preferences

To add existing Teamcenter preferences (already in the **tc_preferences.xml** file) to the GM Overlay preference file, add the new values to the **gmo_preference_merge.xml** file. For example, in the **tc_preferences.xml** file:

```
TC_test=
First
Second
```

To add another value, **third**, for GM Overlay functionality, in **gmo_preference_merge.xml** file, add the preference as follows:

```
TC_test=
Third
```

After an installation or upgrade, the preference in the database contains all the three values:

```
TC_test=
First
Second
Third
```

Override Teamcenter preference values

To override the Teamcenter preference values with GM Overlay values, add the preference in the **gmo_preference_override.xml** file. For example, the **tc_preferences.xml** contains the following:

```
TC_test=
First
Second
```

To have only one value, **third**, for GM Overlay functionality, in the **gmo_preferenec_override.xml** file, add the following preference:

```
TC_test=
Third
```

After installation and upgrade, the preference in the database has only one value:

```
TC_test=
Third
```

Using workflow handlers

Overview of using workflow handlers

Handlers are the lowest level building blocks in the workflow. They are small Integration Toolkit (ITK) programs used to extend and customize tasks. There are two kinds of handlers:

- *Action handlers* perform an action, such as attaching objects or sending an e-mail.
- *Rule handlers* confirm that a defined rule has been satisfied. If the rule is met, the handler returns the **EPM_go** command, allowing the task to continue. If the rule is not met, it returns the **EPM_nogo** command, preventing the task from continuing.

For more information about workflow design, see *Workflow Designer on Rich Client*.

Action handlers

The GM Overlay feature provides the following action handler:

- **GMPDM-add-baseline-release-status**

Replaces the baseline status with *base-item-revision-status_baseline-status* if the base item revision is in the process of being released.

Rule handlers

The GM Overlay feature provides the following rule handlers:

- **GMPDM-PDS-upload_with-url**

Uploads Teamcenter attributes to GPDS (Global Product Development System).

- **GMPDM Enforce Rules**

Validates the dataset before it progresses to a new release level.

The **-filecheck** argument checks whether the dataset passed filecheck. Based on this result, workflow progresses to a new release level. This argument is optional if it is not required to enforce that files have passed the File Checker program.

- **GMPDM-GRP-validation-check**

Performs a validation check.

Configuring Repeatable Digital Validation

Repeatable Digital Validation (RDV) is a solution that automates some of the design and modeling activities that occur during product development. The RDV tools allow you to accurately model specific product assembly configurations and use the results with analytical and visualization tools to facilitate learning and decision making. You use these tools to create digital mockups of a product or part of a product, rather than having to create expensive physical models and mockups. You can filter product or product family data, so that the user only works in an area of interest and does not have to deal with the complexity of the full product.

Configuring tessellation

Overview of configuring tessellation

Tessellation is the process of creating a lightweight representation of a CAD model that generates a JT file that can be visualized in Teamcenter Lifecycle Visualization.

The **EPM-invoke-system-rule** and **EPM-invoke-system-action** workflow handlers provide the facility to initiate and control run-time intensive (external) calculations and updates, such as the creation of lightweight 3D models in asynchronous (background) mode. This feature allows the tessellation process to run in the background, allowing you to interact with Teamcenter while the tessellation is in progress. You can use this feature in workflow process templates to ensure that a valid lightweight 3D model exists before the process advances to the next stage or before a release status is applied.

You must set the following **PATH** statement in the startup script to be able to execute workflow scripts by their base name only to avoid platform-specific settings for the **-system=parameter** option, such as **-system=gmo_check_renderings.pl**:

Windows:

```
set PATH=%TC_ROOT%\bin;%PATH%
```

Linux

```
PATH=$TC_ROOT/bin:$PATH;export PATH
```

Background tessellation

To enable the background tessellation feature, the review tasks in process templates must use the following rule handler in the **Complete** action of the perform-signoff task:

```
Check-signoff QUORUM=-1 Designer
EPM-invoke-system-rule -system=gmo_check_renderings_pl -signoff
-EPM_undecided
```

This starts an asynchronous process to run the tessellator if no up-to-date **DirectModel** dataset (JT dataset) exists. The following message is displayed even though it appears with all required signoffs (quorum reached).

```
The process will not be completed until the background
tessellation job is completed
```

The signoff task automatically completes and advances to the next step when the tessellation is successfully completed. Otherwise, the background process automatically reverts the user's signoff to **EPM_undecided**, which reduces the quorum, or rejects the signoff with an indication of the problem in the signoff comment depending on whether **-EPM_undecided** is specified as an argument for the **EPM-invoke-system-rule** handler argument.

Foreground tessellation

To allow the tessellation process to run in the foreground, perform one of the following actions when using the Teamcenter Automotive Edition–GM Overlay solution:

- Set the **TC_FOREGROUND_TESSELLATION** environment variable to a nonempty string (true/yes) in the startup script.

Windows:

```
set TC_FOREGROUND_TESSELLATION=true
```

Linux:

```
TC_FOREGROUND_TESSELLATION=true ; export
TC_FOREGROUND_TESSELLATION
```

Use the **EPM-invoke-system-rule** handler in the **Complete** action of the perform-signoff task, as follows:

```
EPM-invoke-system-rule -system=gmo_check_renderings_pl
-EPM_undecided -signoff
```

- Use the **EPM-invoke-system-rule** handler in the **Complete** action of the perform-signoff task, as follows:

```
EPM-invoke-system-rule -debug -system=gmo_check_renderings_pl
-EPM_undecided -signoff -f
```

Note:

Do not use the **-signoff** argument if the **EPM-invoke-system-rule** or **EPM-invoke-system-action** handlers are used in zero-step release procedures.

Placement of the tessellation handler

The background tessellation feature allows tessellation to occur in asynchronous mode so the workflow process can continue upon initiation of tessellation. The following changes are required to the procedure definition template to run in this mode:

- Place the **EPM-invoke-system-action -system=gmo_check_renderings.pl** in the **Complete** action of the root task of the release procedure. This handler should be placed last in the **Complete** action after the **add-status** handler.
- Remove all other placements of the previous background tessellation handler in the same procedure.
- Remove all instances of **EPM-attach-related-objects -relation=IMAN_Rendering -type=DirectModel** from the same procedure.

Batch tessellation

Overview of batch tessellation

The batch tessellation utility is used to tessellate released **UGMASTER** and **UGALTREP** datasets in a batch using the **gmo_batch_tessellation_script.pl** utility.

The utility runs in two parts. The first part runs an ITK utility, **TC_ROOT/bin/find_released_datasets**, which queries the database for all **UGMASTER** and **UGALTREP** datasets that were released within a give time. The results of this query are written to the output file in the following format:

```
@DB/item-id/rev-id/relation/dataset-name
```

The second part invokes the **ugtopv** translator for each part. Release status of the tessellated dataset is applied to the newly created **DirectModel** datasets.

gmo_batch_tessellation.pl

SYNTAX

```
gmo_batch_tessellation_script.pl [-u=user-id -p=password
-g=group ][-bypass][-released_date=DD-MMM-YYYY][-out_file=output-file-name] -h
```

ARGUMENTS

- u** Specifies the Teamcenter user. This is a user with Teamcenter administration privileges.
- p** Specifies the Teamcenter password associated with the **-u** argument.
- g** Specifies the Teamcenter group associated with the **-u** argument.
- bypass** Turns the bypass on.
- released_date** Specifies the released date in *DD-MMM-YYYY* format. If not specified, the default is today's date. The **UGMASTER** and **UGALTREP** datasets released after this date form the result set.
- out_file** Specifies the name of the output file. The default file is **query_output.txt**.
- h** Displays help for this utility.

FILES

The parts on which tessellation failed are logged in the **/tmp/year_MM_DAY.log** file.

EXAMPLES

- To tessellate all **UGMASTER** and **UGALTREP** datasets that were released today, enter the following command on a single line:

```
$TC_ROOT/bin/gmo_batch_tessellation_script.pl
```

- To tessellate all **UGMASTER** and **UGALTREP** datasets that were released after September 20, 2002 using the **results.txt** file as the intermediate file, enter the following command on a single line:

```
$TC_ROOT/bin/gmo_batch_tessellation_script.pl
-released_date=20-Sep-2002 -out_file
```

Configuring the AliasStudio encapsulation

Tasks to be complete before using the AliasStudio encapsulation

Configuration tasks must be completed before using the AliasStudio encapsulation in Teamcenter Automotive Edition.

Enable the AliasStudio launcher

The following steps must be performed on each machine on which the AliasStudio application is running, for each user, before launching AliasStudio from Teamcenter Automotive Edition:

Solaris:

1. Export the **ALIAS_LOCATION**=*AliasStudio-installation-directory*, for example, */usr/law/alias 12.0*.

On the Solaris platform, export:

```
LD_LIBRARY_PATH=$ALIAS_LOCATION/lib:/usr/openwin/lib:/usr/dt/  
lib:$UGII_BASE_DIR/ugii/uglibs:$LD_LIBRARY_PATH
```

2. Run **appLaunch** (*ALIAS_LOCATION/bin/AppLaunch*).

The system launches the Alias Wavefront Application Launcher.

3. Select the **Select Default** check box and click **Save**.

The default values are saved.

Windows:

1. Ensure that the path to the AliasStudio application, for example, **C:\AliasWavefront\StudioTools12.0\bin**, is in the PATH.

2. Run **appLaunch** (*ALIAS_LOCATION/bin/AppLaunch*).

3. Select the **Select Default** check box and click **Save**.

The default values are saved.

Configure the default AliasStudio encapsulation tool

1. Open the Type application in Teamcenter Automotive Edition.
2. Expand the **Dataset Type** node in the **Types** tree.

3. Select the **ALIAS_PROJECT** dataset type.
4. In the right pane of the window, ensure that **AUTOSTUDIO_TOOL** is displayed first in the **List of selected tools** list.

Edit the alias.bat file

- Edit the **alias.bat** file located in the *TC_ROOT/bin* directory, replacing **<Enter the path of the Alias Application>** with the path of the AliasStudio application.

Start the AliasStudio application from the rich client interface

Linux:

- To start the AliasStudio application from the rich client interface on Linux, ensure that the **mailcap** file contains the following entry:

```
application/alias;\
<Enter the path of alias.bat (TC_ROOT/bin/alias.bat)>
```

Windows:

1. Edit the **alias.reg** file located in the *TC_ROOT/data* directory by replacing **<Enter the path of the Alias Application>** with the path of AliasStudio application.
2. Update the registry editor by running *TC_ROOT/data/alias.reg*.

Setting the environment variables for the AliasStudio encapsulation

The following variable settings must be made prior to running the AliasStudio application from within Teamcenter Automotive Edition:

- **ALIAS_WANT_IMAN**

Set to a value (any value) to display the Teamcenter Automotive Edition interface within the AliasStudio application.

- **ALIAS_IMAN_DATA**

Controls which Teamcenter Automotive Edition database is accessed.

It must point to the directory where **tc_profilevars.bat** (Windows) or **tc_profilevars** (Linux) can be found. In a default Teamcenter Automotive Edition installation, it is usually the same location as **TC_DATA**.

- **ALIAS_IMAN_LIB**

Points to the directory containing Teamcenter Automotive Edition library.

- **TC_ROOT**

Generally set when Teamcenter Automotive Edition is installed. On a Linux platform, this environment must be set correctly for Studio/Engineering Process Management.

- **ALIAS_DEBUG**

Set to =1.

- **ALIAS_IMAN_DATABASE_NAME**

Specifies the name of the Teamcenter Automotive Edition database in use. AliasStudio can access only one database at a time and cannot switch databases without restarting the AliasStudio application.

Business rules

Business rules, including extension rules, canned methods, and action rules, relate to the AliasStudio encapsulation.

Canned methods

The following canned method is related to the AliasStudio encapsulation:

- **GMO_CM_alias_import_seed_part**

This method is registered on the postaction of the **ALIAS_PROJECT** dataset. It locates the named reference of the new dataset. If no dataset has named references, the method looks for a template seed part in the **Home** folder of the administrative user. The method obtains the named reference of the seed parts and imports it as a named reference to the new dataset.

AliasStudio encapsulation preferences

Use the following preference to configure the encapsulation of the AliasStudio application:

- **Alias_Project_Dataset_Reference_Name**

Specifies the name of the reference used to create an **ALIAS_PROJECT** dataset. Only one value is expected for this preference. If more than one value is supplied, only the first value is used.

AliasStudio encapsulation utilities

The utilities related to the AliasStudio encapsulation are as follows.

- **gmo_create_alias_seedpart**

Creates a new folder in the **infodbaHome** folder and inserts an AliasStudio seed part dataset, with a template seed part as a named reference, into the folder.

- **find_modified_datasets**

Searches for all latest working item revisions of a given type with a dataset of a given type attached by a specified relation that have been created or modified after the input date.

- **batch_export_translate_import**

Exports, translates, and then imports visualization data corresponding to the datasets in the input file.

- **gmo_check_renderings**

Starts the tessellation script in the background in asynchronous mode.

- **gmo_tessellate_renderings**

Used by the **gmo_check_renderings** script.

- **gmo_workflow_postprocess**

Initiates a configurable action in a given workflow task.

Configuring materials engineering data management

Overview of configuring materials engineering data management

You must have the following business rules, preferences, utilities, workflow handlers, and environment variables to configure materials engineering data management.

Material form attribute synchronization – Teamcenter to NX

The following table describes the mapping of material form attributes between Teamcenter and NX.

Teamcenter source property			UGMASTER/UGPART target attribute						
Object	Property	Import behavior	Item type	UG title	Display title	Type	Length	Desc	
CORP_Part Material Specification Form	Material Engineer	Update Teamcenter	CORP_Part	p_mateng	Material Engineer	String	36	Part Material Engineer	
	Appearance Engineer	Update Teamcenter		p_appeng	Appearance Engineer	String	36	Part Appearance Engineer	
	Paint Engineer	Update Teamcenter		p_pnteng	Paint Engineer	String	36		
	Part Material	Update Teamcenter		p_mat	Part Material	String	128	Part Material Spec (first line) [Note 1]	
				p_mat2	(none)	String	128	Part Material Spec (second line)	
				p_mat3	(none)	String	128	Part Material Spec (third line)	
				p_matr	(none)	String	128	Part Material Spec (fourth line)	
				p_pcoat	Paint/Coating	String	128	Part Paint or Coating Spec (first line)	
	Paint/Coating	Update Teamcenter		p_pcoat2	(none)	String	128	Part Paint or Coating Spec (second line)	
				p_pcoat3	(none)	String	128	Part Paint or Coating Spec (third line)	
				p_pcoat4	(none)	String	128	Part Paint or Coating Spec (fourth line)	
				p_perf	Part Performance	String	128	Part Performance Spec (first line)	
	Part Performance	Update Teamcenter		p_perf2	(none)	String	128	Part Performance Spec (second line)	
				p_perf3	(none)	String	128	Part Performance Spec (third line)	
				p_perf4	(none)	String	128	Part Performance Spec (fourth line)	
				p_pcperf	Paint/Coating Performance	String	128	Paint/Coating Performance Spec (first line)	
	Paint/Coating Performance	Update Teamcenter							

Teamcenter source property			UGMASTER/UGPART target attribute				
			p_pcperf2	(none)	String	128	Paint/Coating Performance Spec (second line)
			p_pcperf3	(none)	String	128	Paint/Coating Performance Spec (third line)
			p_pcperf4	(none)	String	128	Paint/Coating Performance Spec (fourth line)
Finish	Update Teamcenter		p_fin	Finish	String	128	Part Finish (first line)
			p_fin2	(none)	String	128	Part Finish (second line)
			p_fin3	(none)	String	128	Part Finish (third line)
			p_fin4	(none)	String	128	Part Finish (fourth line)
Part Appearance to Match	Update Teamcenter		p_appear	Part Appearance to Match	String	128	Appearance to Match (first line)
			p_appear 2	(none)	String	128	Appearance to Match (second line)
			p_appear 3	(none)	String	128	Appearance to Match (third line)
			p_appear 4	(none)	String	128	Appearance to Match (fourth line)
Service Part Requirements	Update Teamcenter		p_svcreq	Service Part Requirements	String	128	Service Part Reqts (first line)
			p_svcreq2	(none)	String	128	Service Part Reqts (second line)
Additional Requirements	Update Teamcenter		p_addreq	Additional Requirements	String	128	Additional Reqts (first line) (Pulled from CAD System Notes Library)
				(none)	String	128	Additional Reqts (second line)
				(none)	String	128	Additional Reqts (third line)
				(none)	String	128	Additional Reqts (fourth line)

Materials engineering data management utilities

Material form templates are imported into Teamcenter using the **gmo_create_material_form_templates** utility, which creates a folder structure with a root of **GM Preferred Materials Catalog** and creates folders of type **Material Template** into which forms of type **Material** are inserted based on the content of an input file. The forms are populated with the attribute values specified in the input file.

For more information and syntax for this utility, see *Teamcenter Utilities*.

Materials engineering data management preferences

Use the **GMO_MATFORM_AUTOCREATE** preference to configure materials engineering data management:

- **GMO_MATFORM_AUTOCREATE**

Enables automatic creation of a **Material** form in Teamcenter, which is attached to the **CORP_Part** revision, when users save a **UGMaster** file in NX for which material attributes are defined. This is a site preference.

A. Using GM Overlay preferences

The following table describes some of the GM Overlay preferences. Use these preferences to determine such behavior as how item revisions are revised and what types of relations are created when using Teamcenter Automotive Edition.

Preference	Description
TCAE_installation_level	Determines whether to check for Teamcenter Automotive Edition and GM Overlay licenses.
TCAE_autogen_name_form_types	Allows automatic generation of form names when a specified form type is created and attached to an item revision.
TC_Configuration_File	Determines the configuration file used to convert an input file.
TC_external_system_attr_value_separator	Determines the behavior of the Send to External System command on the Tools menu in Structure Manager.
TC_external_system_bom_attributes	Determines the behavior of the Send to External System command on the Tools menu in Structure Manager.
TC_external_system_item_type	Determines the behavior of the Send to External System command on the Tools menu in Structure Manager.
TC_external_system_url	Determines the behavior of the Send to External System command on the Tools menu in Structure Manager.
TC_external_system_url_value_name	Determines the name tag of the information posted to the URL defined in the TC_external_system_url preference. The cgi-bin/server script uses this name tag to search for the posted data.
TC_send_to_external_system	Determines whether the Send to External System command

Preference	Description
TC_Validation_Datasettype	displays on the Tools menu in Structure Manager.
TC_Validation_Utility	Determines which datasets types can be validated upon import.
EngChange Revision_Change-Type_Revise_no_copy	Determines the utility that performs the validation and conversion process.
EngChange Revision_Change-Type_SaveAs_no_copy	Specifies objects that are not copied forward to a new item revision when a change object of the specified type is revised or saved as a new revision.
GMO_enable_new_pnum_match_criteria	Specifies objects that are not copied forward to a new item revision when a change object of the specified type is saved as a new revision.
GMO_GRP_show	Determines whether the system checks for EWO_status during an audit.
GMO_GRP_hostname_default	Determines whether the Global Part Reuse Project (GRP) buttons and menu commands display in the My Teamcenter and Structure Manager toolbars.
GMO_GRP_browser_default	Specifies the host name of the Web server connected to via the GRP-I2 code.
GMO_GRP_PORT	Specifies the browser used by the Global Part Reuse Project (GRP) code.
GMO_GRP_req_ir_types	Specifies the port used by the Global Part Reuse Project (GRP) code.
	Specifies the item revision types supported by the Global Part Reuse Project (GRP) code. Allows users to edit and validate the item revision types specified in this preference.

Preference

Description

GMO_GRP_edit_prop_mapping_default

Specifies the Global Part Reuse Project (GRP) mapping values for Teamcenter Automotive Edition–GM Overlay property names. These names and the GRP property names are separated by the tilde (~) delimiter.

GMO_GRP_req_edit_props_default

Specifies the item revision (and item revision-related) properties whose values are extracted. The Global Part Reuse Project (GRP) code uses composite property syntax to retrieve these values.

GMO_MATFORM_AUTOCREATE

Determines whether the system automatically creates a **Material** form in Teamcenter and attaches it to the **CORP_Part** revision (if one is not already attached) when a user saves a **UGMaster** dataset in NX containing material part attributes.

RDV_build_intent_status

Defines the release status applied on the **CORP_Build** object when a new version of a tracking vehicle definition is imported.

RDV_default_options_for_tracking_vehicle

Defines the options created for a saved variant rule when creating a tracking vehicle.

RDV_default_option_values_for_tracking_vehicle

Defines the values for the options created using the **RDV_default_options_for_tracking_vehicle** preference.

RDV_usage_synchronized_architecture_type

Determines which architecture breakdown elements are created while importing usages with the **RDV_import_usage** utility.

IPVBOM_build_intent_status

Lists the statuses of a build intent to be skipped when exporting the build intent using the **gmo_ipvbom_export** utility.

Preference	Description
IPVBOM_compare_mode_var_level_RevisionCompare_ItemTypes	Compares the revision IDs of two selected BOM lines using the custom comparator.
IPVBOM_compare_mode_var_level_Occurrence_Notes	Compares quality and variant condition differences of two selected BOM lines using the custom comparator.
IPVBOM_UPCFNA_OccNoteName	Defines which occurrence notes are used with IPVBOM functionality.