TEAMCENTER

Vendor Management — Deployment and Administration

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

1. Overview of Vendor Management

What is Vendor Management?

To compete effectively in today's marketplace, many companies outsource the manufacturing of the component parts in their products in significant numbers. A rich ecosystem of suppliers, manufacturers, and distributors (collectively referred to as *vendors*) support the outsourcing process by providing parts to a manufacturing company. Vendors provide both direct materials (such as parts and assemblies) and indirect materials (such as tooling in the manufacturing process). Managing vendor and vendor part information is therefore critical to the outsourcing process. Using Teamcenter Vendor Management, you can integrate your PLM system and process knowledge with your vendor supply chain.

You can model vendor contributions to a product by capturing information about vendors and the parts that they provide to satisfy the form, fit, and function of commercial parts. You can also manage the changes in vendor information over time. Vendor Management is available in the My Teamcenter and Structure Manager applications. You require standard Teamcenter user and administrator accounts to use Vendor Management.

Vendors, vendor parts, and commercial parts

Vendors

A vendor supplies one or more parts and can be a supplier, manufacturer, or distributor. For example, Goodparts, a company that supplies parts to your company is a vendor. Two distinctive attributes of a vendor are the company location and the vendor contacts or company contacts.

A vendor's company location provides information about the vendor's site or facility, including the name (required), web site, address, region, location code, location type, and notes about that site. An employee designated as the one-point contact in the vendor's organization is referred to as a company contact or vendor contact. A company location can be related to any number of company contacts. However, a company contact can be related to only one company location.

Similarly, you can associate several company contacts with one vendor. For example, Natalie is a sales person from Goodparts' regional sales office in Detroit, MI, USA, and Sasha is a sales person from their regional sales office in Stuttgart, Germany.

Vendor parts

The part that a vendor supplies is called a **vendor part**.

A vendor part:

- Is obtained from a single vendor, and not from multiple vendors. However, a vendor may supply multiple vendor parts.

- Can be associated with one or more commercial parts.
- Has a vendor part name assigned by the vendor or assumes the company's name for the part.
- Has a vendor ID and vendor name, which identify the vendor supplying the part.
- Has a part number assigned by the manufacturer of the part.
- Is related to a vendor company location.

Vendor parts may also be classified based on their status, such as, preferred, backup, approved, or obsolete. This allows your company to always use the best source of vendor parts. It also allows you to grade the overall product bill of materials (BOM).

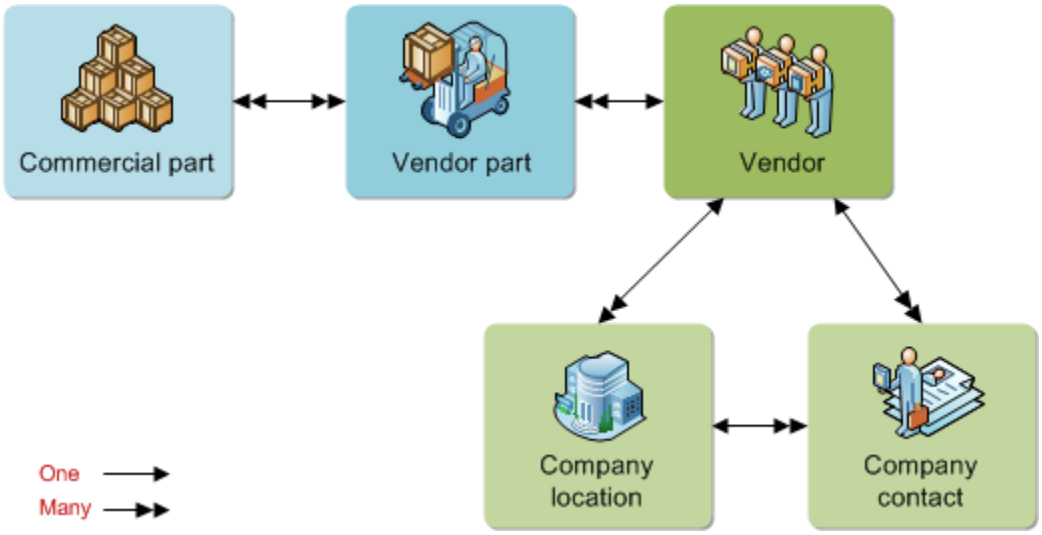
> Note:
>
> A **VendorPart** object cannot be sent to Structure Manager to be included in a product structure unless the vendor part is related to an Item Revision by a **Vendor Representation** relation. Additionally, you cannot add an object to a vendor part in the BOM structure because a vendor part is always the leaf node (lowest level node) in the hierarchy.

Commercial parts

A commercial part is a part that is made externally by a vendor who resells, builds, or designs and builds the part for your company. In this context, your company is referred to as the original equipment manufacturer (OEM). Commercial parts can be off-the-shelf market parts or custom-designed parts.

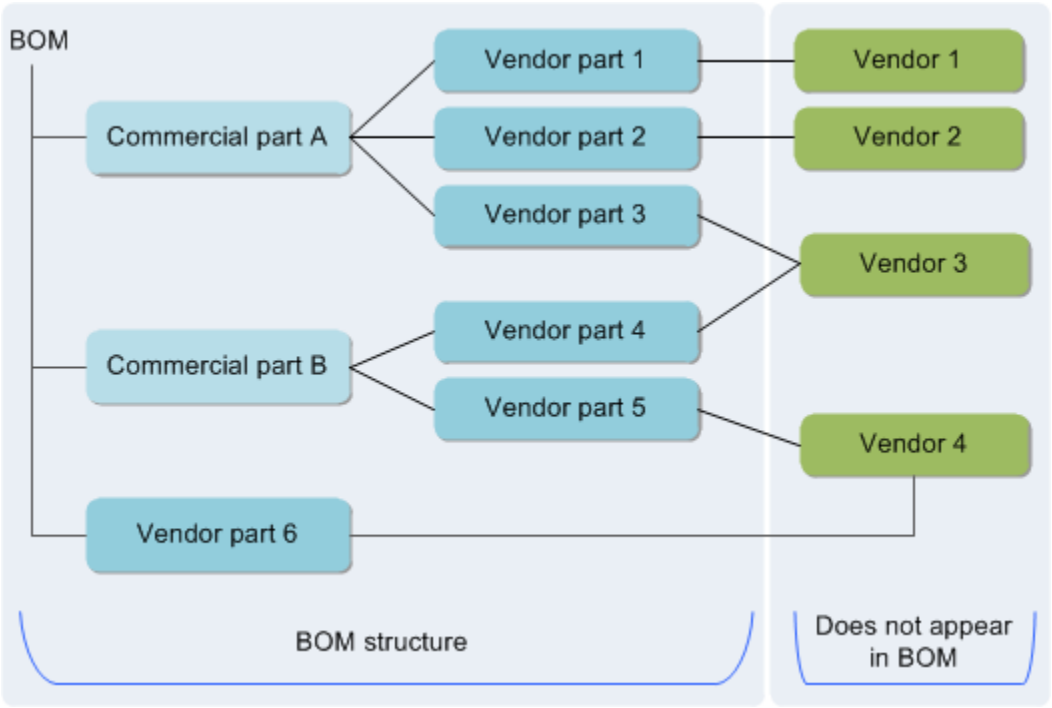
The form, fit, and function of a commercial part can be satisfied by one or more vendor parts that you may source from one or more vendors. Typically, a commercial part has one or more associated **VendorPart** objects.

In Vendor Management, you can classify commercial parts by status such as preferred, approved, or obsolete. This allows you to grade the overall product BOM.

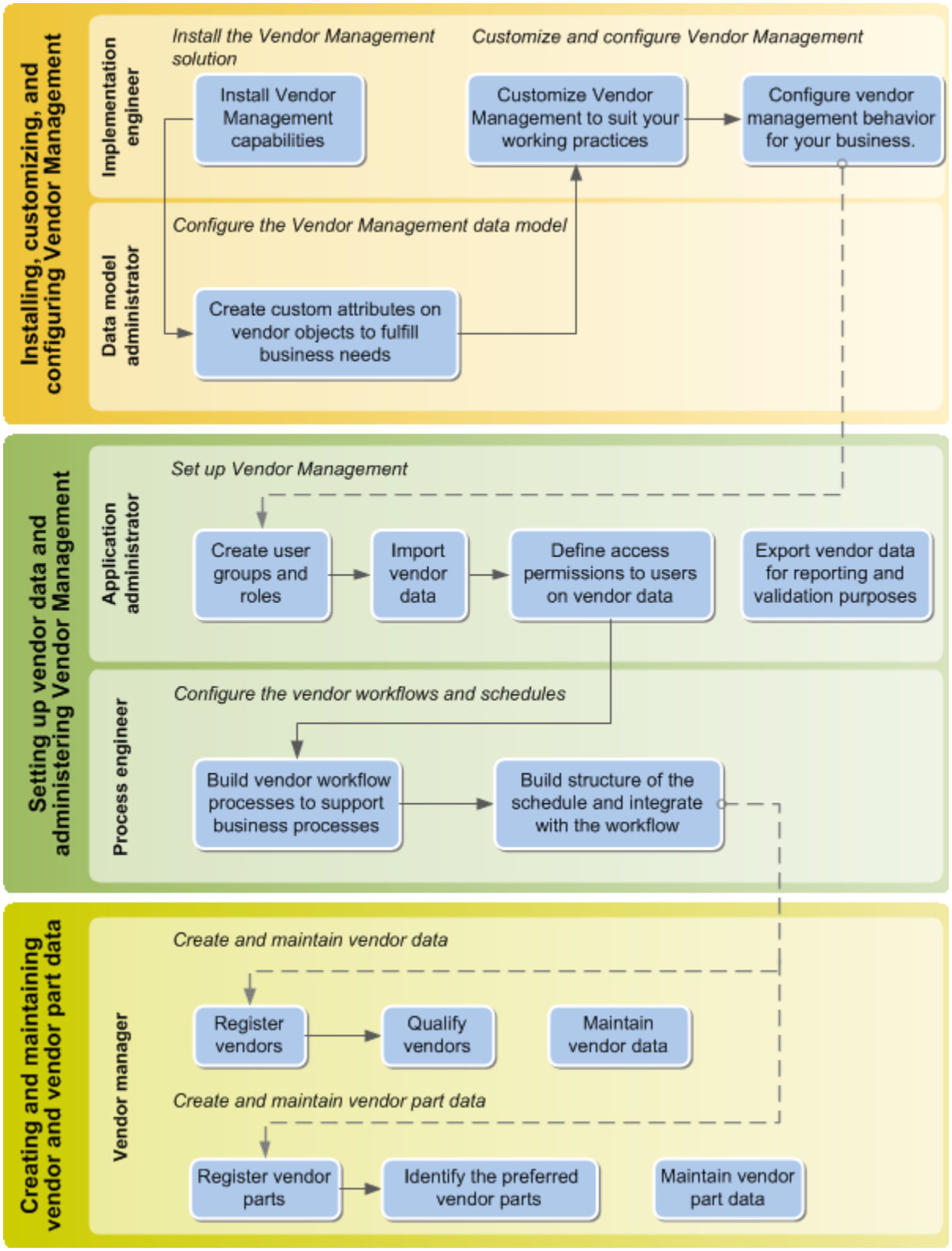


Relation between vendors, vendor parts, and commercial parts

You can view vendor data from both — a vendor-centric and a part-centric point of view. You relate vendors to the parts that they supply. Vendor parts in turn may be related to commercial parts. You can use both commercial parts and vendor parts in your BOMs.



Vendor Management process description and user roles



The following users are involved in the Vendor Management business process:

User role	Responsibilities	Tasks
Implementation Engineer	Is responsible for the upkeep, configuration, and reliable operation of computer systems, especially multiuser computers, such as servers. Also responsible for: • Product installation (for example, Teamcenter) • Application installation (for example, Vendor Management) • Upgrade from earlier releases • Dispatcher installation • Integrations • Server and client installation • FSC, and communications • Multisite, and managing servers	Install the Vendor Management solution Customize Vendor Management Configure Vendor Management
Data Model Administrator	Is responsible for the configuration of the data model to meet company requirements and user needs. Also responsible for: • Data model configuration • LOV definitions • Condition behavior settings • Rule configuration such as naming rules and deep copy rules	Configure the Vendor Management data model

User role	Responsibilities	Tasks
Application Administrator	Is responsible for the configuration of software and data to meet company requirements and user needs. Also responsible for: • User and data security • Data migration • Data import and export • Process configuration • Smart folder configuration • Document management settings • Publication structure definition	Creating user groups Creating vendor roles Importing vendor data Defining access permissions to users on vendor data Exporting vendor data for reporting and validation purposes
Process Engineer	Is responsible for the business process configuration in the software and process configuration. The process engineer builds vendor workflows and schedules as required.	Building vendor workflow processes to support business processes Building the schedule and integrating it with the workflow
Vendor Manager	Manages vendor process from requests through selection. Oversees the entire Vendor Management process. Tasks include: • Registering vendors, creating vendor parts, managing make/buy decisions, and importing vendor bulk data. • Deciding the workflow process to be used for the bid process, creating bid packages, submitting the bid to a workflow, conducting	Register vendors Qualify vendors Maintain vendor data Register vendor parts Identify preferred vendor parts Maintain vendor part data

User role	Responsibilities	Tasks
	bid review meetings, and monitoring the bid progress and making improvements to it. • Coordinating the status of vendor parts according to the decision of the review board and participating in the workflow.	

Understanding Vendor Management using an example

Let us consider a scenario where you are manufacturing the Nanobox, a compact and flexible embedded industrial PC.

Considering various factors, you have decided to outsource the connector and housing components of the Nanobox. You would therefore need to create and maintain vendor and vendor part information for these parts.

You have manufacturing plants for the Nanobox in Detroit, MI, USA, and in Stuttgart, Germany. Considering this, you have identified some new vendors. Bestparts and Goodparts are housing vendors, and Patriot is the vendor for connectors.

You start by registering new vendors, qualifying them, and then maintaining vendor data as needed.

You also register the vendor parts such as the connector and housing, and identify preferred vendor parts based on your procurement team's input. You can then maintain the vendor part data as required.

Vendor Management in conjunction with other Teamcenter applications

Vendor Management is available in the My Teamcenter and Structure Manager applications. It also interacts with the following Teamcenter applications:

- Aerospace and Defense

  Installing Vendor Management is a prerequisite if you implement the Aerospace and Defense solution.

- Industry-focused solutions such as Consumer Packaged Goods

You can optionally use Vendor Management with other industry-focused solutions such as Consumer Packaged Goods.

- Teamcenter Client for Microsoft Office

  You may use Vendor Management with Teamcenter Client for Microsoft Office to import vendor data in bulk as a spreadsheet, using Microsoft Excel.

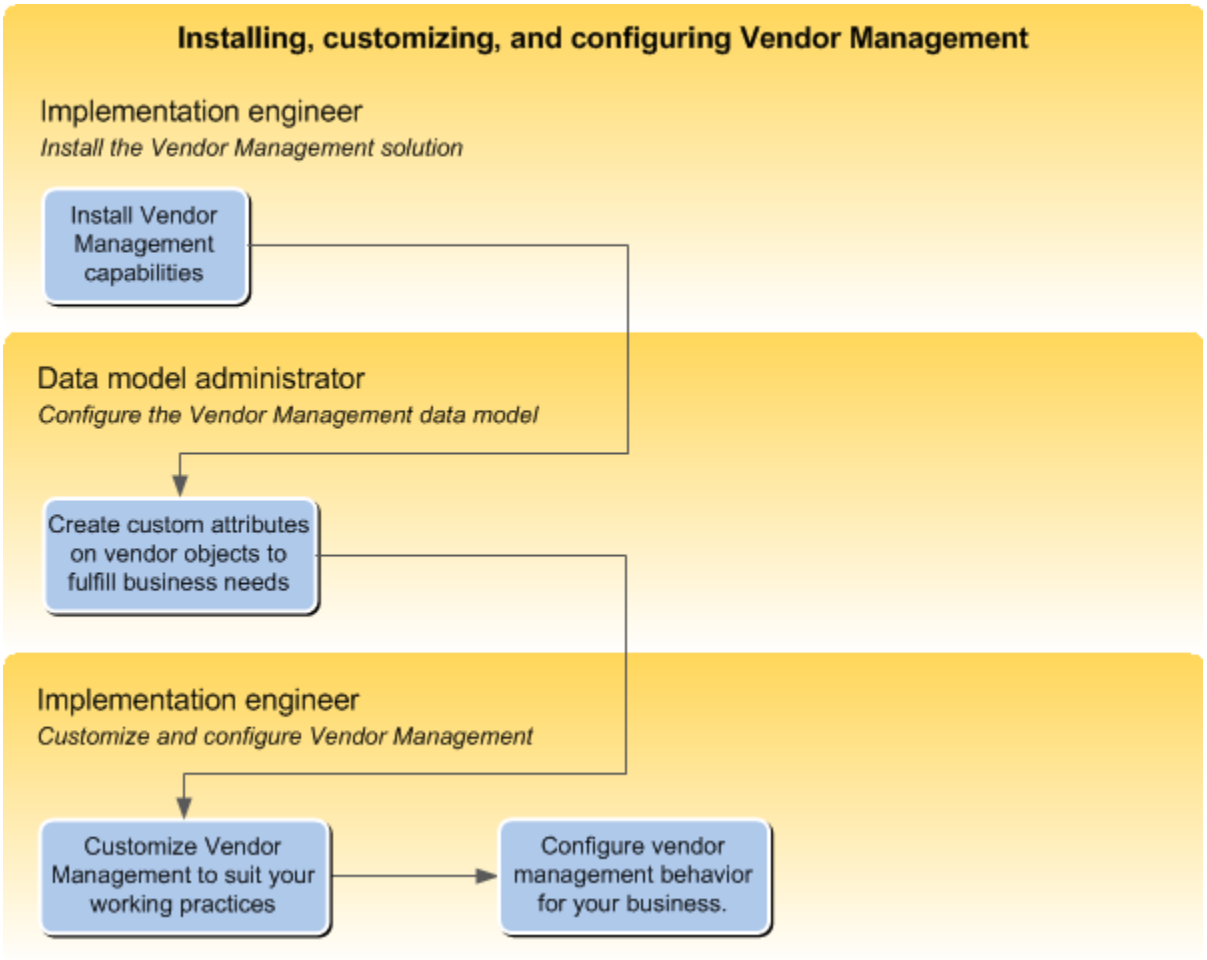
- Teamcenter Integration Framework

  You can optionally use Vendor Management with Teamcenter Integration Framework for exchanging vendor data with Supplier Collaboration.

2. Installing, customizing, and configuring Vendor Management

Installing, customizing, and configuring Vendor Management: process overview and associated user roles

Before end users can start working with Vendor Management, it must be installed, customized, and configured as required by your organization.



User role	Description	Tasks
Implementation engineer	Is responsible for the upkeep, configuration, and operation of computer systems, especially multiuser computers, such as servers. Also responsible for:	Install the Vendor Management solution Customize Vendor Management Configure Vendor Management

User role	Description	Tasks
	• Product installation (for example, Teamcenter) • Application installation (for example, Vendor Management) • Upgrade from earlier releases • Dispatcher installation • Integrations • Server and client installation • FSC and communications • Multisite and managing servers	
Data model administrator	Is responsible for the configuration of the data model to meet company requirements and user needs. Also responsible for: • Data model configuration • LOV definitions • Condition behavior settings • Rule configuration such as naming rules and deep copy rules	Configure the Vendor Management data model

Install Vendor Management

Vendor Management is installed as part of the standard Teamcenter core installation. To install Vendor Management, select the **Vendor Management** subfeature under **Supplier Relationship Management** when you run Deployment Center or Teamcenter Environment Manager (TEM).

For the installation considerations and procedures, see *Teamcenter Installation Using Deployment Center* or the TEM-based Teamcenter installation guides (for *Windows* or *Linux*) and the rich client installation guides (for *Windows* or *Linux*) in the Teamcenter documentation.

Configure the Vendor Management data model

You can configure the Vendor Management data model to meet company requirements and user needs by creating:

- Custom attributes for vendor objects

  The data model administrator creates custom attributes for vendor objects to fulfill business needs. Attributes are added or changed added or changed as required using Business Modeler IDE.

  For example, if you need to store some additional performance-related data for a vendor that does not map to the vendor attributes provided out-of-the-box, you can create additional attributes such as **Response Time**.

- LOV definitions

  You may require some additional values such as **Most preferred** and **On hold** to be specified as options in the **Vendor Status** list because your company has its own list of vendor status values.

- Condition behavior settings

  You can configure the system such that if a vendor has a status such as **Obsolete**, you cannot create vendor parts for that vendor.

- Rule configurations such as naming rules and deep copy rules

  You can configure the naming rule such that the vendor name is specified as per your company's policy.

Customize Vendor Management

You can customize Vendor Management to suit your working practices. For example, you can write Integration Toolkit (ITK) programs to do the following automatically:

- Create a commercial part as a component of your product assembly.

  Use the **VMS_create_commercial_part** function.

- Create vendors, specify vendor roles, and edit the vendor list.

Use the **VMS_create_vendor** function. You can also create vendor roles, such as distributor or supplier, with the **VMS_create_vendor_role** function.

You can also copy a vendor with the **VMS_copy_vendor** function or delete a vendor with the **VMS_delete_vendor** function.

Include the **vm\vms.h** file in your program and call the **VMS_init_module** function first to initialize the Vendor Management classes. For an example that uses the Vendor Management functions, refer to the *TC_ROOT***sample\vms\vms_util.c** file.

You can also use the service-oriented architecture (SOA) service APIs to customize Vendor Management.

Configuring Vendor Management

Configure how to highlight changes made to Vendor and Vendor Part objects

You can configure the Teamcenter environment to highlight changes made to the **Vendor** or **Vendor Part** objects after you add these objects as **Solution Items** or **Problem Items** in a change notice. This allows you to track changes to the selected properties of these objects. The changes are highlighted as follows:

- Deleted values of the selected properties are highlighted using a red strikethrough.
- Additions to the selected properties are highlighted in green and are italicized.
- For replaced or revised values of the selected properties, the old and the new values are shown side by side.

> Note:
>
> The change notice must be open and executing changes for it to be tracked.
>
> The changes are highlighted only if your site administrator has set the **AWC_Enable_RedLine_feature** preference to **TRUE**.
>
> You can share the changes made to a vendor or vendor part with another site in a Multi-Site environment by using the **TIECMOptionSetDefault** option set to export the **Vendor** and **Vendor Part** objects. When the vendor or vendor part is opened at the destination site, the changes are highlighted at that site as well.

To configure the Teamcenter environment to highlight changes made to **Vendor** or **Vendor Part** objects in a change notice, do the following:

1. In BMIDE, create a custom template that is dependent on the following templates:
   - Foundation

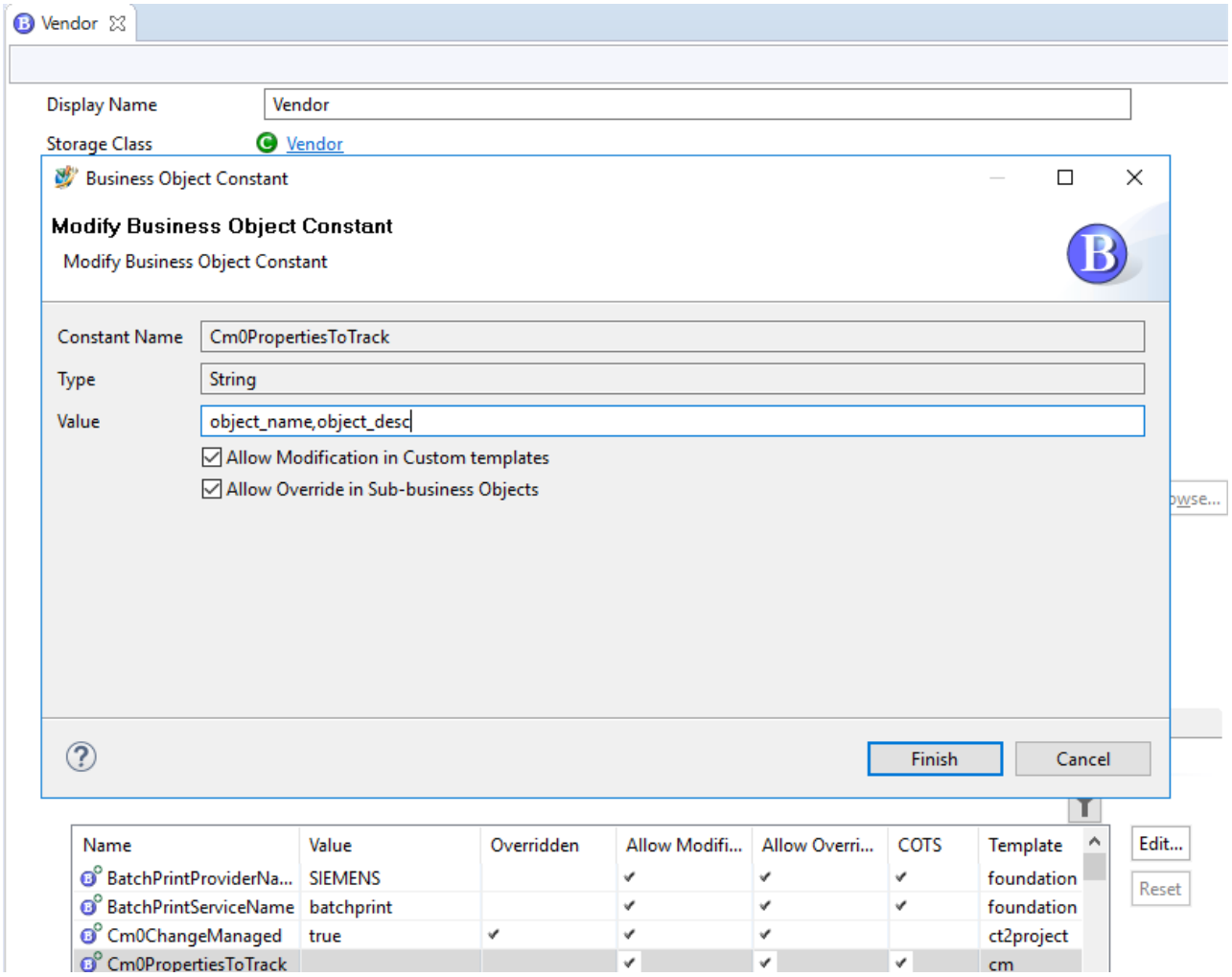
- Vendor Management
- Change Management

For more information about using BMIDE, see Configuring Your Business Data Model in BMIDE in the Teamcenter documentation.

2. In BMIDE, open the **Vendor** and **Vendor Part** objects and set the **Cm0ChangeManaged** business object constant to **True**.

3. In the **Cm0PropertiesToTrack** business object constant, add the internal names of the **Vendor** and **Vendor Part** properties that you want to track.

   For example, you can add these internal names in the **Value** field for tracking: first_name,last_name,email_address,fax_number,object_desc,phone_business,phone_home,phone_mobile,title



You can track only persistent properties of the following attribute types:

- Boolean
- Character
- Date
- Double
- ExternalReference
- Integer
- LongString

  > Note:
  >
  > A string length of up to 256 characters is supported.

- String

  > Note:
  >
  > By default, a string length of up to 256 characters is supported.

- TypedReference
- UntypedReference

4. To save the changes to the data model, choose **BMIDE→Save Data Model**, or click the **Save Data Model** button on the main toolbar.

5. On the menu bar, choose **BMIDE→Deploy Template**. Type the password, click the **Connect** button, and when a connection is established, select the **Generate Server Cache?** check box and click **Finish**.

6. When deployment is done, check the status in the **Console** view.

7. For each custom template that you have created, run the **install_callback** utility from a Teamcenter command prompt to enable highlighting the changes in Teamcenter.

   - To enable highlighting for the **Vendor** object in the custom template, run the **install_callback** utility as follows:

     ```
     install_callback -u=<User Id of an administrator> -p=<Password of
     the administrator> -g=group -mode=create -type=CompareTo_Callback
     ```

```
-library=libcm1cmaws -function=CM_get_otherside_object_for_compare
-name="<ObjectType>"
```

Vendor is an example of *<ObjectType>*.

- To enable highlighting for the **ManufacturerPart** object in the custom template, run the **install_callback** utility as follows:

  ```
  install_callback -u=<User Id of an administrator> -p=<Password of
  the administrator> -g=group -mode=create -type=CompareTo_Callback
  -library=libcm1cmaws -function=CM_get_otherside_object_for_compare
  -name="<ObjectType>"
  ```

  ManufacturerPart is an example of *<ObjectType>*.

8. If you select a vendor or vendor part and create a new change notice, you must define the default relationship between the **ChangeNoticeRevision** object and the **Vendor** or **Vendor Part** object. To do this, create the following preferences:

   - Create the **ChangeNoticeRevision_Vendor_default_relation** preference and set the value of this preference to **CMHasSolutionItem** to define the default relationship as **Solutions** between the **ChangeNoticeRevision** object and the **Vendor** object.

     Set the value of this preference to **CMHasProblemItem** to define the default relationship as **Problems** between the **ChangeNoticeRevision** object and the **Vendor** object.

   - Create the **ChangeNoticeRevision_ManufacturerPart_default_relation** preference and set the value of this preference to **CMHasSolutionItem** to define the default relationship as **Solutions** between the **ChangeNoticeRevision** object and the **Vendor Part** object.

     Set the value of this preference to **CMHasProblemItem** to define the default relationship as **Problems** between the **ChangeNoticeRevision** object and the **Vendor Part** object.

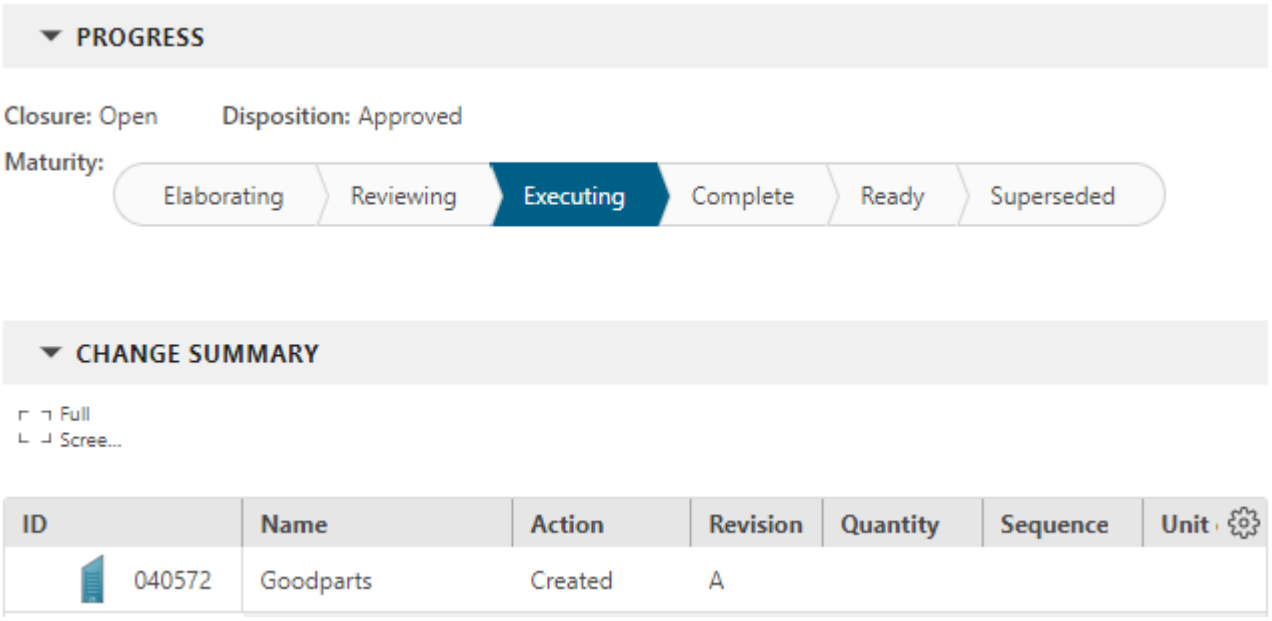
   The valid values are **CMHasSolutionItem** and **CMHasProblemItem**.

9. To define which properties of the **Vendor** or **Vendor Part** object appear as columns in the change notice's **Change Summary Table** in the **Overview** tab, add the properties to the **ChangeSummaryTableColumns** preference in the following format:

   ```
   Type:TYPE_NAME,Property:PROPERTY_NAME,Width:PIXEL_WIDTH
   ```

   In this example, the property name for the column comes from **PROPERTY_NAME** and the name for the column is displayed from the owning type, **TYPE_NAME**.

   ```
   Type:Vendor,Property:contact_name,Width:128
   ```



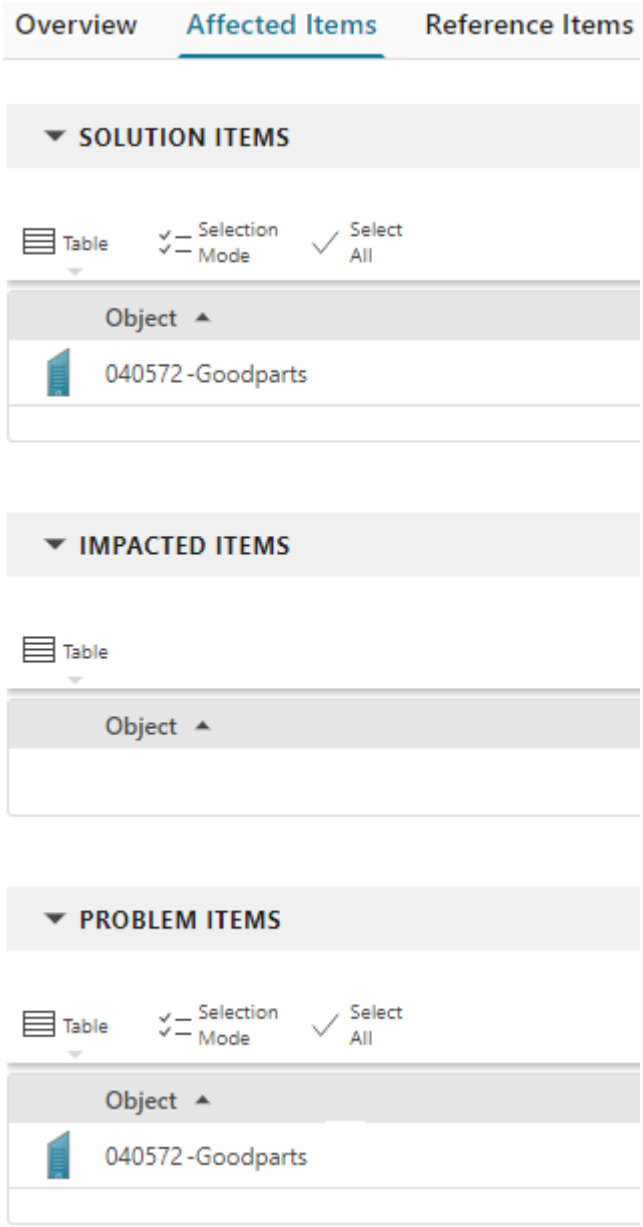
10. To display vendors and vendor parts in the **Solution Items** and **Problem Items** sections of the change notice's **Affected Item's** tab, add the **Vendor** or **Vendor Part** objects to the style sheet for the **Problem Items** and **Solution Items** sections. In the **XRT Editor**, set the source of the **objectSet** property to **CMHasSolutionItem.Vendor,CMHasSolutionItem.ManufacturerPart**.

    In the style sheet for the **Solution Items** section, add the following:

```
 <page titleKey="tc_xrt_AffectedItems">
  <section titleKey="tc_xrt_SolutionItems">
    <objectSet
source="CMHasSolutionItem.Vendor,CMHasSolutionItem.ManufacturerPart,CMHasSolu
tionItem.ItemRevision,CMHasSolutionItem.Mdl0ModelElement,CMHasSolutio
nItem.Cfg0AbsConfiguratorWSO,CMHasSolutionItem.Bom0ConfigurableBomEle
ment,CMHasSolutionItem.Fnd0AbstractOccRevision"
defaultdisplay="tableDisplay" sortdirection="ascending"
sortby="object_string">
```

    In the style sheet for the **Problem Items** section, add the following:

```
<section titleKey="tc_xrt_ProblemItems">
    <objectSet
source="CMHasSolutionItem.Vendor,CMHasSolutionItem.ManufacturerPart,CMHasProb
lemItem.ItemRevision,CMHasProblemItem.Mdl0ModelElement,CMHasProblemIt
em.Cfg0AbsConfiguratorWSO,CMHasProblemItem.Bom0ConfigurableBomElement
,CMHasProblemItem.Fnd0AbstractOccRevision"
defaultdisplay="tableDisplay" sortdirection="ascending"
sortby="object_string">
```



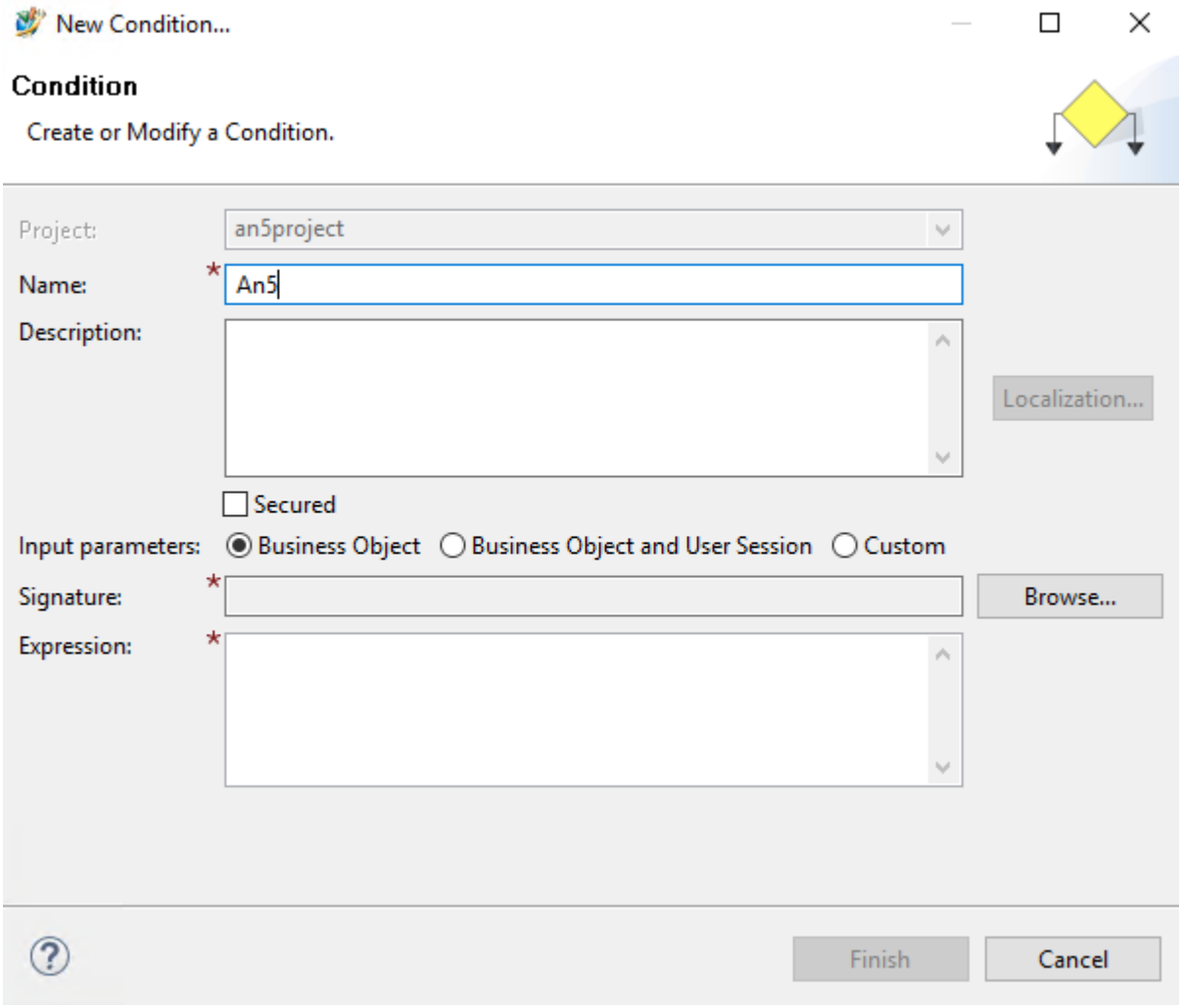
11. After the change notice is closed, the changes are not highlighted in the vendor or vendor part. To view the changes in the vendor or vendor part, from the **Primary** toolbar, click the **View** > **Show Redlines**.

Configure which users can create vendors

To allow only certain users to create vendors, you must configure Teamcenter to display the **Create Vendor** command to only those specific users. In BMIDE, create a condition that restricts which users can view the **Create Vendor** command, and assign this condition to the **Vm1CreateVendor** command. Do the following:

1. In the Business Modeler IDE, start the new condition wizard in one of these ways:

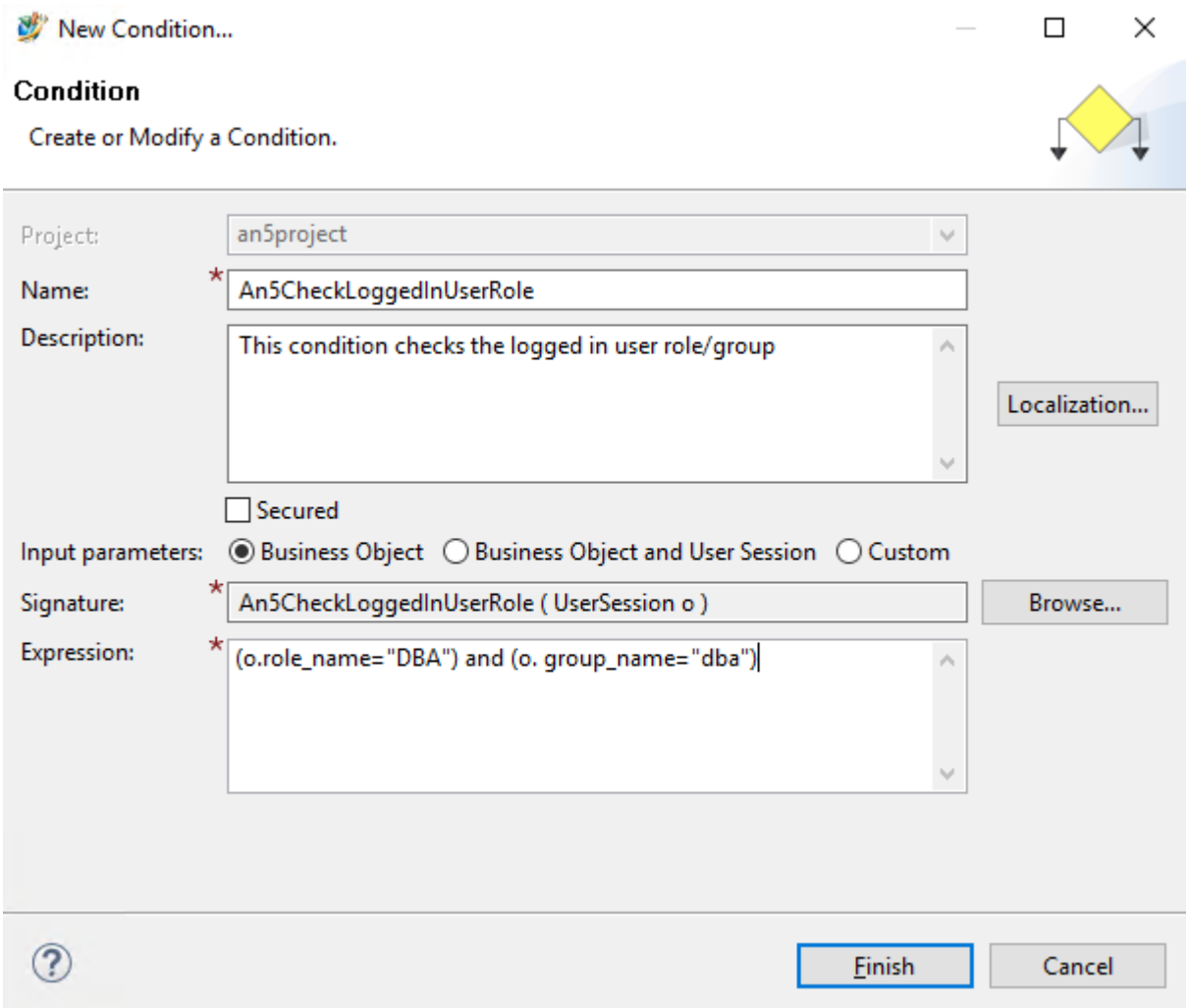
- On the menu bar, choose **BMIDE→New Model Element**, in the **Wizards** box type **condition**, and click **Next**.
- Open the **Extensions\Rules** folders, right-click the **Conditions** folder, and choose **New Condition**.



2. In the **Condition** dialog box, define the condition parameters.

For this parameter	Do this
Name	Type **An5CheckLoggedInUserRole** as the name that you want to assign to the condition in the database.
Description	Type a description of the condition.

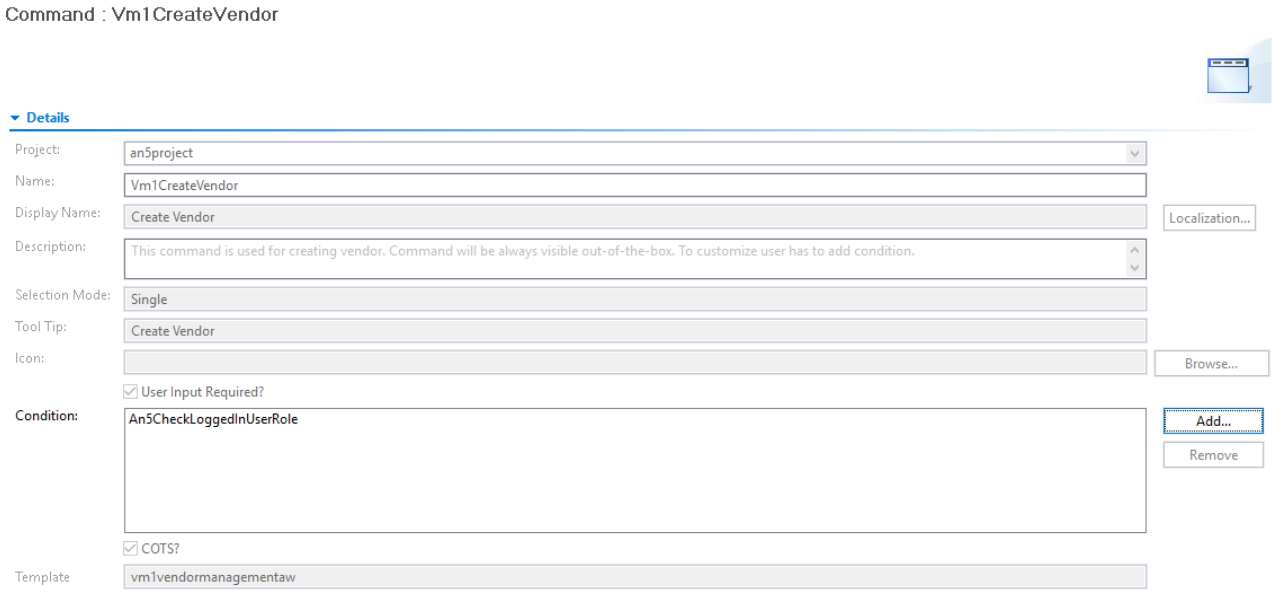
For this parameter	Do this
Input parameters	Select **Business Object** to apply the condition to a business object.
Expression	Type `(o.role_name="DBA") and (o.group_name="dba")` to allow only users with the **DBA** role and **dba** group to view the **Create Vendor** command from the **Primary** toolbar.



3. Click **Finish**.

4. Open the **Vm1CreateVendor** command.

5. In the **Condition** box, click **Add** and search for and add the **An5CheckLoggedInUserRole** condition to the **Vm1CreateVendor** command.



6. To save the changes to the data model, choose **BMIDE→Save Data Model**, or click the **Save Data Model** button on the main toolbar.

7. On the menu bar, choose **BMIDE→Deploy Template**. Type the password, click the **Connect** button, and when a connection is established, select the **Generate Server Cache?** check box and click **Finish**.

8. When deployment is done, check the status in the **Console** view.

Display only preferred vendor parts

Vendor Management is controlled by the **vendormanagement** template that is managed in Business Modeler IDE. To configure Vendor Management for your business, set the following:

- **showAllVendorParts** condition

  This rule displays all vendor parts in the product structure.

  A user can set this rule in Structure Manager by selecting from a list of valid conditions. Upon selection, the system filters the list of vendor parts displayed based on the selected condition. The criteria are evaluated against each potential vendor part in the structure, and only parts meeting the criteria are displayed.

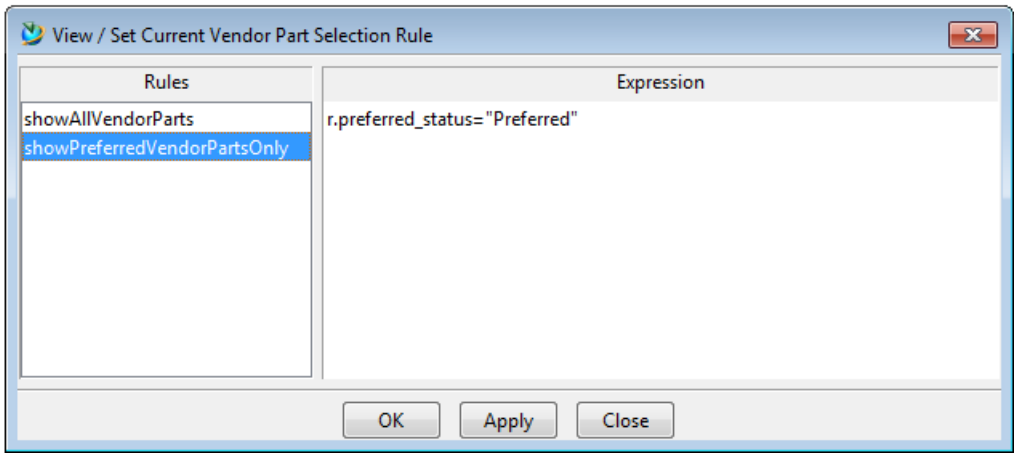
- **showPreferredVendorPartsOnly** condition

  This rule displays preferred vendor parts in the product structure. The **showPreferredVendorPartsOnly** rule displays only those parts that are marked as **Preferred** using the preferred status property on the vendor representation relation.

A user can set this rule in Structure Manager by selecting from a list of valid conditions. Upon selection, the system filters the list of vendor parts displayed based on the selected condition. The criteria are evaluated against each potential vendor part in the structure, and only parts meeting the criteria are displayed.

You can create a selection condition that limits the vendor parts that are displayed in the data pane. For example, you may see only preferred parts.

1. Choose **Tools→Vendor Parts Selection Rule**.



2. In the **View/Set Current Vendor Parts Selection Rule** dialog box, select the **showAllVendorParts** or **showPreferredVendorPartsOnly** and click **OK**.

   Teamcenter applies the selected rule to future actions.

For example, in the **Nanobox example**, you may configure the business rules such that the vendor manager can view all vendor parts, but other users can view only the preferred vendor parts.

Set preferences

You can set the following preferences before using Vendor Management:

- **CommercialPart_DefaultChildProperties**

  Specifies each property that can be displayed as a child of a commercial part, for example, **vendorparts**.

- **CommercialPart_PseudoFolder**

  Specifies each property that can be displayed as a pseudofolder under a node of a commercial part, for example, **vendorparts**. This preference must be set to allow users to view and manage the dataset associated with a commercial part.

- **CompanyLocation_DefaultChildProperties**

  Specifies each property that can be displayed as the child of a company location, for example, **ContactInCompany**.

- **CompanyLocation_PseudoFolder**

  Specifies each property that can be displayed as a pseudofolder under a node of a company location object, for example, **ContactInCompany**. Each entry must be a vendor location property.

- **ContactInCompany_relation_primary**

  Enables the **ContactInCompany** relation for each business object specified in this preference, for example, **CompanyLocation**.

- **ICS_classifiable_types**

  Defines all business object types that can be classified. Add any Vendor Management types you want to classify.

- **LocationInCompany_relation_primary**

  Enables the **LocationInCompany** relation for each business object specified in this preference, for example, **ItemRevision**.

- **ManufacturerPart_DefaultChildProperties**

  Specifies each property that can be displayed as the child of a manufacturer part, for example, **commercialparts**. Each entry must be a property of **ManufacturerPart**.

- **ManufacturerPart_DefaultPSEProperties**

  Specifies the properties of vendor parts that are visible in Structure Manager, for example, **vendor_part_num**. Each entry must be a property of **ManufacturerPart** or a property of the **VMRepresents** relation.

- **ManufacturerPart_PseudoFolder**

  Specifies each property that can be displayed as a pseudofolder under a node of a **ManufacturerPart** object, for example, **ProvidedFrom**. Each entry must be a property of **ManufacturerPart**.

- **ManufacturerPartRevision.REGISTEREDTO**

  Specifies the business object types to which the vendor part revision properties style sheet is applied.

- **ManufacturerPartRevision.RENDERING**

Registers the properties style sheet for vendor part revision objects.

- **ManufacturerPartRevision.SUMMARYRENDERING**

  Registers the summary style sheet for vendor part revision objects.

- **ManufacturerPartRevSummary.SUMMARY_REGISTEREDTO**

  Specifies the business object types to which the vendor part revision summary style sheet is applied.

- **ManufacturerPart.REGISTEREDTO**

  Specifies the business object types to which the vendor part properties style sheet is applied.

- **ManufacturerPart.RENDERING**

  Registers the properties style sheet for vendor part objects.

- **ManufacturerPart.SUMMARYRENDERING**

  Registers the summary style sheet for vendor part objects.

- **ManufacturerPartSummary.SUMMARY_REGISTEREDTO**

  Specifies the business object types to which the vendor part summary style sheet is applied.

- **ProvidedFrom_relation_primary**

  Specifies the business object types for which the **ProvidedFrom** relation is enabled, for example, **ManufacturerPart**.

- **PseudoFolder.ContactInCompany_ColumnPreferences**

  Specifies the properties that are displayed when a **ContactInCompany** pseudofolder is selected in the **Details** view of the rich client, for example, **CompanyContact.title**.

- **PseudoFolder.LocationInCompany_ColumnPreferences**

  Specifies the properties that are displayed when a **LocationInCompany** pseudofolder is selected in the **Details** view of the rich client, for example, **CompanyLocation.city**.

- **PseudoFolder.ProvidedFrom_ColumnPreferences**

  Specifies the properties that are displayed when a **ProvidedFrom** pseudofolder is selected in the **Details** view of the rich client, for example, **CompanyLocation.street**.

- **Restricted_Vendor_Part_Statuses**

  This site preference is optionally defined for commercial parts and revisions. It specifies restricted statuses that prevent users from associating commercial parts or part revisions with vendors, for example, **Obsolete**. This allows you to configure a business rule so that, with certain release statuses on vendor parts, users are not allowed to associate a commercial part or commercial part revision with vendor parts or vice versa.

  > Note:
  >
  > After you freeze a commercial part, you cannot change the preferred status for the vendor part.

- **Restricted_Vendor_Statuses**

  This site preference is optionally defined for vendors. It specifies restricted statuses that prevent users from associating vendor parts with vendors, for example, **Obsolete**. This allows you to configure a business rule so that, with certain release statuses on vendor objects, users are not allowed to create vendor parts for a vendor.

- **TC_Distributor_info_form**

  Specifies the **Distributor** role default form type, for example, **DistributorInfo**.

- **TC_Manufacturer_info_form**

  Specifies the **Manufacturer** role default form type, for example, **ManufacturerInfo**.

- **TC_Quality_Document_relation_primary**

  Specifies the primary types of the **TC_Quality_Document** relation, for example, **Vendor**.

- **TC_Supplier_info_form**

  Specifies the **Supplier** role default form type, for example, **SupplierInfo**.

- **TC_vendor_part_rel_relation_primary**

  Specifies the primary object types of the **TC_vendor_part_rel** relation, for example, **Item**.

- **TC_runtimeTypeToExpand**

  Specifies the list of runtime properties to be expanded in the deep copy tree. Valid values are runtime properties internal names. By default, **vendorparts** and **vendors** are the values specified. This ensures that copy options are displayed for vendor parts and vendors on the Deep Copy page. If you do not want copy options to be displayed for vendor objects, remove the **vendors** entry from the **Values** list. This is a site preference.

- **Vendor_DefaultChildProperties**

  Specifies the properties that can be displayed as children of vendors, for example, **vendorparts**.

- **Vendor_Part_BOMLine_Properties**

  Specifies which Vendor Part attributes you want to add to the default Vendor ID in the **Vendor Parts BOMLine** attribute. This preference allows you to customize the Bill Of Materials (BOM) export templates that generate BOMs for suppliers.

- **Vendor_PseudoFolder**

  Specifies each property that can be displayed as a pseudofolder under a node of a **Vendor** object, for example, **commercialparts**. Each entry must be a property of **Vendor**.

- **VendorRevision_DefaultChildProperties**

  Specifies the properties that can be displayed as children of vendor revisions, for example, **vendor_role_info**. Each entry must be a property of **VendorRevision**.

- **VendorRevision_PseudoFolder**

  Specifies each property that can be displayed as a pseudofolder under a node of a **VendorRevision** object, for example, **LocationInCompany**. Each entry must be a property of **VendorRevision**.

- **VMRepresents_relation_primary**

  Specifies the primary object types of the **VMRepresents** relation, for example, **Item**.

- **VMS_Maintain_Vendor_History**

  Specifies whether Teamcenter moves or copies vendor data after it is changed.

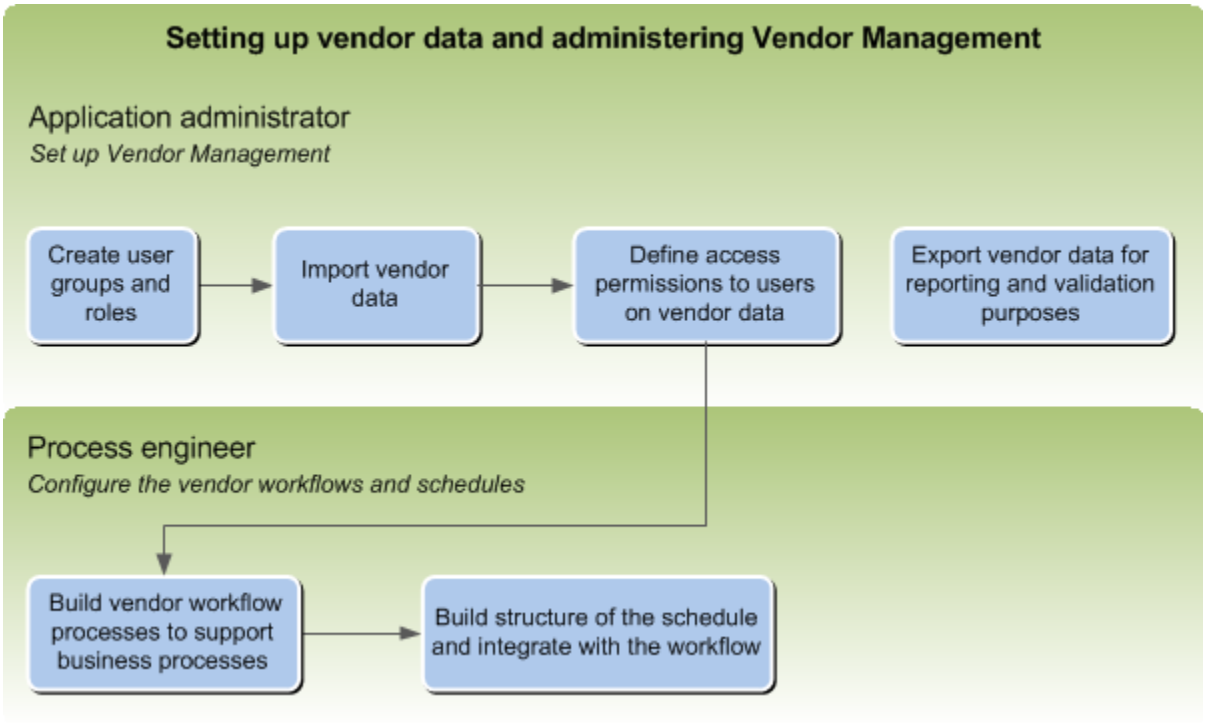
- **VMS_vendor_part_selection_rule**

  Specifies the current vendor part selection rule, for example, **showAllVendorParts**.

3. Setting up vendor data and administering Vendor Management

Setting up and administering Vendor Management: process overview and user roles

After installing, customizing, and configuring Vendor Management as required by your organization, you can set up and maintain the vendor and vendor part information.



Typically, the roles of users who set up vendor data and administer Vendor Management are:

User role	Description	Tasks
Application administrator	Is responsible for the configuration of software and data to meet company requirements and user needs. Handles: • user and data security • data migration	Creating user groups Creating vendor roles Importing vendor data Defining access permissions to users on vendor data Exporting vendor data for reporting and validation purposes

User role	Description	Tasks
	• data import and export • process configuration • smart folder configuration • document management settings • publication structure definition	
Process engineer (subrole of Application Administrator)	Responsible for the business process configuration in the software. The process engineer builds vendor workflows and schedules as required.	Building vendor workflow processes to support business processes Building the schedule and integrating it with the workflow

Set up Vendor Management

Workflow for setting up Vendor Management

To set up Vendor Management, you must create the required user groups and vendor roles. You must import the vendor data and define access permissions for it.

As and when required, you can export vendor data for reporting and validation purposes.

Creating user groups

Using the Organization application, you can create user groups to organize users into units for which you can grant access.

For example, in the **Nanobox scenario**, you would have different user groups for design engineers, who work on product design, and vendor managers, who manage vendor and vendor part data.

Creating vendor roles

In Teamcenter, a vendor can take on one or more roles. Role definitions are not fixed and can be customized as required. You can create a custom role, provided that the Teamcenter administrator has set this up. In a standard Teamcenter installation, the following vendor roles are provided:

- **Distributor**

  Resells but does not make parts.

- **Supplier**

  Makes parts to order for the OEM.

- **Manufacturer**

  Makes parts for the general market.

You can create the necessary role types for your business with the Organization application.

For example, in the **Nanobox scenario**, Bestparts, Goodparts, and Patriot are **Suppliers**.

Importing vendor data

You set up Vendor Management by importing vendor data such as vendors, vendor parts, and commercial parts. The implementation team imports and sets up your vendor data for you. You may also create your own utilities or programs to import data using Teamcenter APIs.

> Note:
>
> If you are upgrading from pre-Teamcenter 10, Siemens recommends that you **run the migrate_vendor_part_locations utility** after you complete the Teamcenter upgrade. This utility migrates the locations of vendor parts from the **Provided From Relations** property to the **Typed Reference** property.

To import vendors, vendor parts, and commercial parts:

1. Prepare a list of vendors, vendor parts, and commercial parts to import.

2. Configure utility parameters.

3. Execute the custom import utility.

4. Validate that objects are imported correctly.

5. Relate the imported objects to each other.

   After importing the vendor data, you must set up all the associations between the imported objects. For example, you must relate a vendor to the parts that the vendor is capable of supplying (vendor parts), and the vendor parts in turn must be related to the commercial parts that represent them in the part structure.

Migrate location data of vendor parts

You can use the **migrate_vendor_part_locations** utility to migrate the locations of vendor parts from the **Provided From Relations** property to the **Typed Reference** property. You can specify whether to migrate only the first location by using the **migratefirst** argument, or make copies of vendor parts each time the utility finds multiple locations by using the **migrateall** argument.

Syntax

migrate_vendor_part_locations [**-u=***username* {**-p=***password* | **-pf=***password-file*} **-g=***groupname*] [**-option=[list|migratefirst|migrateall]**] [**-h**]

Arguments

-u

Specifies the user ID.

This is a user with Teamcenter administration privileges.

> Note:
>
> If Security Services single sign-on (SSO) is enabled for your server, the **-u** and **-p** arguments are authenticated externally through SSO rather than being authenticated against the Teamcenter database. If you do not supply these arguments, the utility attempts to join an existing SSO session. If no session is found, you are prompted to enter a user ID and password.

-p

Specifies the password.

This argument is mutually exclusive with the **-pf** argument.

-pf

Specifies the password file.

For more information about managing password files, see Manage password files.

This argument is mutually exclusive with the **-p** argument.

-g

Specifies the group associated with the user.

If used without a value, the user's default group is assumed.

-option

Use the **list** value to find applicable vendor parts and print their information in the log file. Use the **migratefirst** value to migrate only the first location of the applicable vendor parts. Use the **migrateall** value to make copies of vendor parts each time the utility finds multiple locations in the applicable vendor parts.

-h

Displays help for this utility.

Environment

As specified in Manually configuring your environment for Teamcenter utilities.

Files

As specified in Log files produced by Teamcenter.

Defining user-access permissions for vendor data

You can restrict permissions to vendor data by defining access rules. For example, you can restrict Vendor Management actions on released vendor parts.

For example, you can define access rules to restrict Vendor Management actions on released vendor parts. You need the write access on a commercial part or revision for adding or removing the vendor parts and editing **VM Represents** relation properties such as **Preferred Status**.

Configure vendor workflows and schedules

Building vendor workflow processes to support business processes

You can configure workflows to support the business processes that your organization uses and to ensure that your organization has up-to-date vendor information.

No specific workflows are provided in Teamcenter for managing vendor data. However, you may implement workflows for the following purposes:

- Releasing vendor parts
- Making vendor parts obsolete
- Approving new vendors using the vendor qualification process
- Onboarding vendors
- Monitoring and reviewing vendor performance

Building the schedule and integrating it with the workflow

You configure vendor schedules as required by your organization. You create a schedule for vendors, and add tasks and attachments to the schedule. You may have one schedule for a single task or for a group of tasks in a project, or you may set up a schedule for each supplier. You may send a schedule task through a workflow by assigning a workflow template to a schedule task. From Schedule Manager, the workflow may be triggered automatically based on the workflow trigger defined, or you can trigger the task by manually launching the workflow.