Teamcenter manufacturing
process management 10.1

Getting Started with Manufacturing
Proprietary and restricted rights notice

This software and related documentation are proprietary to Siemens Product Lifecycle Management Software Inc.

© 2013 Siemens Product Lifecycle Management Software Inc. All Rights Reserved.

Siemens and the Siemens logo are registered trademarks of Siemens AG. Teamcenter is a trademark or registered trademark of Siemens Product Lifecycle Management Software Inc. or its subsidiaries in the United States and in other countries. All other trademarks, registered trademarks, or service marks belong to their respective holders.
# Contents

**Proprietary and restricted rights notice** ........................................... 2

**Getting started with manufacturing** ............................................. 1-1

- Getting started with manufacturing ........................................... 1-1
- Prerequisites .................................................................................. 1-1
- Enabling Manufacturing Process Management .............................. 1-1
- Configuring Manufacturing Process Management .......................... 1-1
- Starting Manufacturing Process Management ............................... 1-2
- Manufacturing Process Management applications ....................... 1-2
- Manufacturing Process Management symbols .............................. 1-4
- Rich client perspectives and views ............................................. 1-6
- Suppressing commands ................................................................. 1-6

**Basic concepts** .............................................................................. 2-1

- Basic concepts .............................................................................. 2-1
- Manufacturing planning model .................................................... 2-1
- Creating process structure and operations .................................... 2-5
- Aligning properties ...................................................................... 2-8
- Creating structures from templates ............................................. 2-11
- Configuring processes with rules ............................................... 2-16
- Configuring processes with variants .......................................... 2-17
- Creating in-context data (absolute occurrences) ......................... 2-18
- Using compositions ...................................................................... 2-18
- Revising processes and operations ............................................. 2-20
- Managing incremental changes .................................................. 2-21
- Understanding relations versus occurrences ............................. 2-23
- Reconciling broken links ............................................................... 2-25
- Controlling access ....................................................................... 2-25
- Managing resources ..................................................................... 2-25
- Releasing a manufacturing process ............................................ 2-26
- Creating product views ................................................................. 2-27
- Managing weld points (features) .................................................. 2-28
- Managing process design in the thin client ................................. 2-28
- Setting manufacturing preferences ............................................ 2-28
- Integrating through Multi-Structure Manager .............................. 2-29

**Basic tasks** ...................................................................................... 3-1

- Basic tasks .................................................................................... 3-1
- Creating a factory structure ......................................................... 3-1
- Creating a manufacturing process ................................................ 3-2
- Creating a preliminary data indicator (baseline) ......................... 3-8
- Create an intermediate data capture ............................................ 3-8
- Performing process simulations .................................................. 3-9
- Design a fixture for manufacturing ............................................. 3-11
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creating reports</td>
<td>3-11</td>
</tr>
<tr>
<td>Displaying GD&amp;T symbols and rich text in forms</td>
<td>3-19</td>
</tr>
<tr>
<td><strong>Planning assembly manufacture</strong></td>
<td>4-1</td>
</tr>
<tr>
<td>Setting manufacturing preferences</td>
<td>4-1</td>
</tr>
<tr>
<td>Associating the EBOM and MBOM</td>
<td>4-2</td>
</tr>
<tr>
<td>Designing processes for precise or imprecise structures</td>
<td>4-3</td>
</tr>
<tr>
<td>Working with relative occurrences</td>
<td>4-4</td>
</tr>
<tr>
<td>Managing studies</td>
<td>4-5</td>
</tr>
<tr>
<td>Body-in-white integrated workflow</td>
<td>4-6</td>
</tr>
<tr>
<td>Work with flexible components and NX</td>
<td>4-7</td>
</tr>
<tr>
<td>Integrating with a manufacturing execution system</td>
<td>4-8</td>
</tr>
<tr>
<td><strong>Planning part manufacture</strong></td>
<td>5-1</td>
</tr>
<tr>
<td>Planning part manufacture</td>
<td>5-1</td>
</tr>
<tr>
<td>Transfer data with the NX CAM Integration</td>
<td>5-1</td>
</tr>
<tr>
<td>Storing postprocessor and simulation drivers in Teamcenter</td>
<td>5-6</td>
</tr>
<tr>
<td>Copying MENC Machining operations</td>
<td>5-10</td>
</tr>
<tr>
<td><strong>Planning resource management</strong></td>
<td>6-1</td>
</tr>
<tr>
<td>Planning resource management</td>
<td>6-1</td>
</tr>
<tr>
<td>Setting access rights</td>
<td>6-1</td>
</tr>
<tr>
<td>Using the Manufacturing Resource Library</td>
<td>6-1</td>
</tr>
<tr>
<td>Install NX templates</td>
<td>6-2</td>
</tr>
<tr>
<td>Enable graphics creation</td>
<td>6-3</td>
</tr>
<tr>
<td>Enable tool retrieval from Teamcenter to NX</td>
<td>6-3</td>
</tr>
<tr>
<td>Exporting the classification hierarchy</td>
<td>6-4</td>
</tr>
<tr>
<td>Manually extend the Classification hierarchy with a new class</td>
<td>6-5</td>
</tr>
<tr>
<td>Building tool assemblies</td>
<td>6-6</td>
</tr>
<tr>
<td>Understanding propagated attributes</td>
<td>6-10</td>
</tr>
<tr>
<td>Adding components using the guided component search</td>
<td>6-11</td>
</tr>
<tr>
<td>Map vendor catalog tools to customer hierarchy</td>
<td>6-11</td>
</tr>
<tr>
<td>Retrieve a tool from Teamcenter in NX</td>
<td>6-12</td>
</tr>
<tr>
<td><strong>Managing the enterprise BOP</strong></td>
<td>7-1</td>
</tr>
<tr>
<td>Managing the enterprise BOP</td>
<td>7-1</td>
</tr>
<tr>
<td>Overview</td>
<td>7-2</td>
</tr>
<tr>
<td>Prerequisites for managing the enterprise BOP</td>
<td>7-5</td>
</tr>
<tr>
<td>Understanding logical designators</td>
<td>7-5</td>
</tr>
<tr>
<td>Understanding the generic BOP</td>
<td>7-7</td>
</tr>
<tr>
<td>Understanding the product BOP</td>
<td>7-8</td>
</tr>
<tr>
<td>Understanding the plant BOP</td>
<td>7-8</td>
</tr>
<tr>
<td>Understanding partitions</td>
<td>7-9</td>
</tr>
<tr>
<td>Understanding origins</td>
<td>7-9</td>
</tr>
<tr>
<td>Creating operations, processes, and partitions</td>
<td>7-10</td>
</tr>
<tr>
<td>Consuming parts</td>
<td>7-11</td>
</tr>
<tr>
<td>Allocating processes and operations to a plant BOP</td>
<td>7-11</td>
</tr>
<tr>
<td>Assigning tools</td>
<td>7-12</td>
</tr>
<tr>
<td>Visualizing a process or operation</td>
<td>7-12</td>
</tr>
<tr>
<td>Understanding time analysis</td>
<td>7-12</td>
</tr>
<tr>
<td>Comparing and propagating differences in enterprise BOP structures</td>
<td>7-13</td>
</tr>
<tr>
<td>Importing and exporting EBOP structures using PLM XML</td>
<td>7-14</td>
</tr>
</tbody>
</table>
## Configuring the manufacturing environment

- Configuring the manufacturing environment ........................................... 8-1
- Configure the update of multiple product views ........................................ 8-1
- Configuring the accountability check ...................................................... 8-2
- Configure new categories in the Time view ............................................. 8-4
- Configuring an MTM data card ............................................................... 8-6
- Configuring the TiCon integration ........................................................... 8-8
- Configuring publishing of manufacturing data ......................................... 8-9
- Configuring the user interface ................................................................. 8-27
- Customizing line balancing reports ......................................................... 8-28
- Configuring standard text ........................................................................ 8-32
- Configuring 3D PDF generation ............................................................... 8-47
- Configuring the comparison of design to manufacturing features ................. 8-50
- Configuring 4D planning .......................................................................... 8-52

### Glossary

- Index ........................................................................................................... Index-1

### Figures

- Example of fabrication process ................................................................. 2-4
- Cloning with incremental changes ........................................................... 2-15
- Using NX CAM Integration data in Teamcenter ....................................... 5-2
- Example result – table asset populated with child item names and form
descriptions ............................................................................................... 8-15
- Sample product structure ......................................................................... 8-15
- Publish Bind dialog box ............................................................................ 8-16
- Expanded ProductPartList alias showing FormDescription entry ............... 8-17
Chapter

1  Getting started with manufacturing

Getting started with manufacturing ........................................... 1-1
Prerequisites ............................................................................. 1-1
Enabling Manufacturing Process Management ............................ 1-1
Configuring Manufacturing Process Management ...................... 1-1
Starting Manufacturing Process Management ............................. 1-2
Manufacturing Process Management applications ...................... 1-2
Manufacturing Process Management symbols ............................. 1-4
Rich client perspectives and views ............................................. 1-6
Suppressing commands ........................................................... 1-6
Chapter

1  Getting started with manufacturing

Getting started with manufacturing

Teamcenter manufacturing process management (Manufacturing Process Management) manages data created in the manufacturing planning stage of the product life cycle. It allows you to manage assembly and fabrication of products whose design is managed in Teamcenter.

Prerequisites

You do not need any special permissions to run any of the Manufacturing Process Management applications.

Enabling Manufacturing Process Management

Several of the Manufacturing Process Management applications require licenses in addition to the standard Teamcenter product license. For details, contact your Siemens PLM Software representative.

Configuring Manufacturing Process Management

Manufacturing Process Management comprises several applications that extend the capabilities of Teamcenter to allow the planning of product manufacturing processes. Before using Manufacturing Process Management, you must install and configure the base Teamcenter product and its associated Oracle or SQL server database. You should refer to the Teamcenter installation and configuration documentation for information on how to install and configure the base product. Once you have installed Teamcenter and the necessary licenses, the basic Manufacturing Process Management applications are available for use.

If you use certain advanced functionality such as the collaboration context integration with Tecnomatix process planning and simulation software, you must install and configure additional components.

For more information, see the Installation on UNIX and Linux Servers Guide or the Installation on Windows Servers Guide.

Some of the data tabs you frequently use to manage manufacturing data may not be available in your default environment.
For more information about how to add and remove data tabs, see the Client Customization Programmer's Guide.

Starting Manufacturing Process Management

It is not necessary to start the Manufacturing Process Management environment separately. Once Teamcenter is running, you can create, view, and edit Manufacturing Process Management data by clicking the buttons for the rich client applications.

Manufacturing Process Management applications

Manufacturing Process Management includes the following rich client applications:

- Manufacturing Process Planner
  Allows you to design a plan that details how to manufacture a product that is an assembly. The manufacturing process plan includes a top-level structure of the process needed to manufacture the product, as well as a detailed design of the individual processes and activities included in the plan. As you build the process structure, you can assign resources to the various processes, operations, and activities. You can also identify the specific locations within the plant where each operation and activity is performed.

  You can also create collaboration contexts that allow you to capture multiple different Teamcenter structures in one container or to share Teamcenter data with another application such as the Tecnomatix process planning software. The shared data can be managed or modified by either application and any changes are available in both environments. This capability is also available in the Multi-Structure Manager application.

  For more information about how to use Manufacturing Process Planner, see the Manufacturing Process Planner Guide.

- Part Planner
  Allows you to design a plan that details how to manufacture a piece part product, such as a piston or engine fan blade. You can plan the production process from raw material to end product, including cutting, drilling, milling, turning, and quality checking operations. In this environment, there is typically no production line, but several resources such as machine tools, fixtures, cutting tools and gages are necessary.

  As in Manufacturing Process Planner, the process plan may include processes, activities and resources.

  For more information about how to use Part Planner, see the Part Planner Guide.

- Plant Designer
  Allows you to create, modify, import, and export a factory structure.

  For more information about how to use the Plant Designer, see the Plant Designer Guide.

- Resource Manager
Getting started with manufacturing

Allows you to store and retrieve resources such as tools, fixtures, machines, and process templates. Resource data is held in a database that is accessible to all users. Resource Manager stores this data in combination with classification information, and it is organized into a hierarchy that is specific to your company.

For more information about how to use Resource Manager, see the Resource Manager Guide.

- Resource Browser

Allows you to retrieve classification-related data such as a hierarchy with corresponding groups, classes, and instances from a Teamcenter database when working in an external application. Resource Browser is delivered as a dynamic link library (DLL). To use it within an external application, you must develop code that uses this library.

For more information about how to use the Resource Browser, see the Resource Browser Guide.

- Report Generator

Allows you to create reports about the process plan and related operations, activities, product structure, and plant structure.

For more information about how to use the Report Generator, see the Report Generator Guide.

- Multi-Structure Manager

Allows you to capture the data in a Teamcenter process or product structure into a collaboration context object for review, sharing, or comparison. You can also create compositions and perform basic structure editing tasks with this application.

You can share captured data with another application such as the Tecnomatix process planning software. The shared data can be managed or modified by either application and any changes are available in both environments.

You can also create a preliminary data indicator (PDI) of any structure in a collaboration context.

For more information about how to use Multi-Structure Manager, see the Multi-Structure Manager Guide.

- Process Simulate

The Process Simulate application allows you to design, analyze, simulate and optimize manufacturing processes from the factory level down to lines and workcells. It provides optional advanced capabilities that are not available in other Manufacturing Process Management applications. Process Simulate is fully integrated into Teamcenter, and you can send data to it from other applications.

You can also use the Manufacturing Structure Viewer in the thin client to view manufacturing data.

Manufacturing Process Management provides interfaces to external systems, allowing you to import and export manufacturing data. For example, you can:

- Import manufacturing feature definitions through the NX CAM Integration.
Share process structure information with Tecnomatix manufacturing process designer applications.

Manufacturing Process Management symbols

The process structure is presented in the Manufacturing Process Management applications in the form of a process structure (sometimes called the MBOM or manufacturing bill of materials), which is similar in concept to an assembly structure. A process structure shows all the operations, activities and other items that are assigned to the process, and their relationships. You create the process structure from the product structure, resequencing and rearranging it as necessary to create an efficient manufacturing process.

Each item in the structure is represented by a symbol according to its type.

<table>
<thead>
<tr>
<th>Item</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item (product)</td>
<td><img src="image1" alt="Symbol" /> (revision)</td>
</tr>
<tr>
<td>Process</td>
<td><img src="image2" alt="Symbol" /> (revision)</td>
</tr>
<tr>
<td>Operation</td>
<td><img src="image3" alt="Symbol" /></td>
</tr>
<tr>
<td>Activity</td>
<td><img src="image4" alt="Symbol" /></td>
</tr>
<tr>
<td>Workstation</td>
<td><img src="image5" alt="Symbol" /> (revision)</td>
</tr>
<tr>
<td>Plant or work area</td>
<td><img src="image6" alt="Symbol" /></td>
</tr>
<tr>
<td>Collaboration context</td>
<td><img src="image7" alt="Symbol" /></td>
</tr>
<tr>
<td>Structure context</td>
<td><img src="image8" alt="Symbol" /> (composition)</td>
</tr>
<tr>
<td>Configuration context</td>
<td><img src="image9" alt="Symbol" /></td>
</tr>
<tr>
<td>Standard text library</td>
<td><img src="image10" alt="Symbol" /></td>
</tr>
<tr>
<td>Standard text folder</td>
<td><img src="image11" alt="Symbol" /></td>
</tr>
<tr>
<td>Standard text library element</td>
<td><img src="image12" alt="Symbol" /></td>
</tr>
<tr>
<td>Consumed product</td>
<td><img src="image13" alt="Symbol" /></td>
</tr>
<tr>
<td>Linked object</td>
<td><img src="image14" alt="Symbol" /></td>
</tr>
<tr>
<td>Object linked but not loaded</td>
<td><img src="image15" alt="Symbol" /></td>
</tr>
<tr>
<td>Item</td>
<td>Symbol</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>Broken link</td>
<td><img src="image" alt="Broken link" /></td>
</tr>
<tr>
<td>Classified object</td>
<td><img src="image" alt="Classified object" /></td>
</tr>
<tr>
<td>Occurrence group</td>
<td><img src="image" alt="Occurrence group" /></td>
</tr>
<tr>
<td>Object checked out</td>
<td><img src="image" alt="Object checked out" /></td>
</tr>
<tr>
<td>Resource</td>
<td><img src="image" alt="Resource" /></td>
</tr>
<tr>
<td>Operation/Station/Activity flow</td>
<td><img src="image" alt="Operation/Station/Activity flow" /></td>
</tr>
<tr>
<td>Weld point</td>
<td><img src="image" alt="Weld point" /></td>
</tr>
<tr>
<td>Datum point</td>
<td><img src="image" alt="Datum point" /></td>
</tr>
<tr>
<td>Arc weld</td>
<td><img src="image" alt="Arc weld" /></td>
</tr>
<tr>
<td>Barrel tool</td>
<td><img src="image" alt="Barrel tool" /></td>
</tr>
<tr>
<td>CLSF data</td>
<td><img src="image" alt="CLSF data" /></td>
</tr>
<tr>
<td>PTP data</td>
<td><img src="image" alt="PTP data" /></td>
</tr>
<tr>
<td>Shop docs</td>
<td><img src="image" alt="Shop docs" /></td>
</tr>
<tr>
<td>Drill tool</td>
<td><img src="image" alt="Drill tool" /></td>
</tr>
<tr>
<td>Form tool</td>
<td><img src="image" alt="Form tool" /></td>
</tr>
<tr>
<td>Groove tool</td>
<td><img src="image" alt="Groove tool" /></td>
</tr>
<tr>
<td>Machining</td>
<td><img src="image" alt="Machining" /></td>
</tr>
<tr>
<td>Mill tool</td>
<td><img src="image" alt="Mill tool" /></td>
</tr>
<tr>
<td>NC machining</td>
<td><img src="image" alt="NC machining" /></td>
</tr>
<tr>
<td>Thread tool</td>
<td><img src="image" alt="Thread tool" /></td>
</tr>
<tr>
<td>Tool</td>
<td><img src="image" alt="Tool" /></td>
</tr>
<tr>
<td>Turning tool</td>
<td><img src="image" alt="Turning tool" /></td>
</tr>
</tbody>
</table>
By default, all Multi-Structure Manager object types are visible in Teamcenter, including in My Teamcenter and in the product structure in Structure Manager. This may result in a very complex structure displayed to the core Teamcenter users. If appropriate, you can hide the Multi-Structure Manager object types for core Teamcenter users by choosing the Edit→Options menu command in a rich client application (for example, Structure Manager or Manufacturing Process Planner), choosing the General→Item menu command, and selecting the types to hide.

Rich client perspectives and views

Within the Teamcenter rich client user interface, application functionality is provided in perspectives and views. Some applications use perspectives and views to arrange how functionality is presented. Other applications use a single perspective and view to present information.

Note: Your administrator can use the HiddenPerspectives preference to prevent the display of some Teamcenter perspectives in the rich client.

For information about editing preference values, see the Preferences and Environment Variables Reference.

If your site has online help installed, you can access application and view help from the rich client Help menu or by pressing F1.

For more information about rich client perspectives and views, see the Rich Client Interface Guide.

Suppressing commands

Not all the menu commands described in the Manufacturing Process Management documentation may be appropriate for all users. An administrator can restrict commands to certain groups or roles with the Command Suppression application, as described in the Application Administration Guide.
Chapter 2  Basic concepts

Basic concepts .............................................. 2-1

Manufacturing planning model ............................. 2-1
  Components of the manufacturing planning model .......... 2-2
  Example of product and process structures ................. 2-4

Creating process structure and operations ................. 2-5
  Defining process operations ................................ 2-6
  Defining process operation data ............................... 2-6
  Defining an NC machining process operation ............... 2-7
  Defining assembly and weld process operation .......... 2-8

Aligning properties ........................................ 2-8
  Assigning occurrences between structures ................. 2-9
  Checking alignment ........................................... 2-10
  Synchronizing aligned occurrences ......................... 2-10
  Managing kits ................................................. 2-11

Creating structures from templates ....................... 2-11
  Defining cloning rules ....................................... 2-12
  Carrying over future incremental change when cloning ........ 2-14
  Reusing process templates .................................. 2-15

Configuring processes with rules .......................... 2-16

Configuring processes with variants ....................... 2-17

Creating in-context data (absolute occurrences) .......... 2-18

Using compositions ......................................... 2-18
  Associating data in the context of a composition .......... 2-19
  Configuring referenced structures in a composition .... 2-20

Revising processes and operations ........................ 2-20

Managing incremental changes .............................. 2-21
  Tracking changes ............................................ 2-21
  Configuring changes ......................................... 2-22
  Configuring with intents .................................... 2-22
  Controlling access .......................................... 2-23

Understanding relations versus occurrences ............... 2-23
Reconciling broken links ........................................ 2-25
Controlling access ............................................... 2-25
Managing resources ............................................ 2-26
Releasing a manufacturing process ....................... 2-26
Creating product views ....................................... 2-27
Managing weld points (features) .......................... 2-28
Managing process design in the thin client ............. 2-28
Setting manufacturing preferences ....................... 2-28
Integrating through Multi-Structure Manager .......... 2-29
Chapter

2 Basic concepts

Basic concepts

Manufacturing Process Management manages data created in the manufacturing planning stage of the product lifecycle. It allows you to manage assembly and fabrication of products whose design is managed in Teamcenter. The following concepts apply to manufacturing planning of both assemblies and parts.

Manufacturing planning model

The product design and development life cycle can be broken into the following stages:

- **Design**
  The development team produces the information that defines the final product. This includes a complete description of the individual components (for example, geometry and tolerances) and an assembly structure. The design may require several analysis processes to ensure it meets the design specifications, for example, stress analysis and thermal analysis.

- **Manufacturing planning**
  In this stage, the manufacturing planning team determines how the product is manufactured. This may include designing an overall plan for the manufacture of the product, and the detailed design of each individual machining or assembly step. After the process design is complete, it may be simulated to verify the reliability of the process by completing activities such as NC tool path verification and weld point allocation.

- **Execution**
  You can now assign the individual steps in the manufacturing plan to specific resources, using manufacturing orders or other criteria. When manufacturing commences, manufacturing execution systems (MESs) collect and manage real-time manufacturing information, perform inventory control, and schedule activities.

  **Note** These stages are not necessarily sequential. For example, the manufacturing process may be designed before the product design is complete. This allows the components of a process to be manufactured before the design of the entire product is released.

Manufacturing Process Management allows you to manage manufacturing planning and execution, using information generated in the design stage.
Components of the manufacturing planning model

The manufacturing planning model includes several interrelated process and operation types, as follows:

**Manufacturing process**
A set of manufacturing process operations and other manufacturing processes that are closely related to each other. It is a mechanism to group operations into logical groups. It also maintains constraints on the order of execution between subprocesses and process operations.

The process structure is the model of the complete manufacturing plan, and describes how and where the product is manufactured. It contains information about which plant structure elements are used and where. It establishes links between the product, plant layout, and the resources necessary to produce the product.

Operations and resources can be shared by several processes. You can define multiple views of a process to allow for different manufacturing requirements.

You can also vary the generic bill of process (BOP) by defining and applying variant or revision rules.

**Process operation**
A process operation represents one controlled step in the process and contains the work done in one work area; technically, it is a leaf node in the process structure. Optionally, you can break a process operation into steps called activities.

Examples of process operations include an NC machining operation on a single machine tool and an operation to assemble several components into a structure.

**Operation activities**
A breakdown of manufacturing process operations into activities, each with an associated start time and duration. The total time of an operation is calculated from the start time and duration of each activity.

**Workstation**
A unique location within the factory to which you can assign manufacturing operations, for example, a paint spraying booth.

**Plant**
A manufacturing facility in which operations and processes are executed.

The plant structure is a hierarchical structure of work areas of the factory in which the product is manufactured. Its definition is independent of, but related to, the manufacturing processes that are performed in it.

**Work area**
Any element in the plant structure such as a factory, line, section, work cell, or station. Typically, a work area is assigned to one process operation. A work area is defined by a location on the shop floor and the process capability it provides.

To group resources further, your organization may define a hierarchy of work areas. For example, a plant may include a welding line, a lathe workstation, a milling workstation, and an inspection workstation. The welding line may contain a welding workstation and a welding work cell.

You configure each work area with standard equipment, and you can further configure equipment that is needed to execute a specific process operation. The
process designer tries to use as much standard equipment in the work area as possible. If additional equipment is needed, the process designer provides work instructions on how to configure the work area with the additional equipment. For example, an NC machining center may be preloaded with standard tools that are commonly used on the machine. If a process operation requires tools that are not preloaded, the operator loads them on the machine when the particular process operation is executed.

**Product structure**
The *product structure* defines the *as-designed* product. It is created by the design engineers in a CAD system such as NX and managed by Teamcenter. Manufacturing Process Management allows you to take the as-designed product structure, create alternative manufacturing views of the product and attach process definitions as necessary. (Optionally, the product structure may contain definitions of manufacturing features.)

**Operation setup**
The model of an operation-specific configuration of an environment in which processes and operations are executed. It describes how consumed items, resources, and a standard work area are utilized to perform an operation.

**Consumed items**
Components in the product definition that are *consumed* by a manufacturing operation.

**Consumed material**
Items that are consumed by a manufacturing operation but are not part of the product definition, for example, glue or paint.

**Raw material**
The initial in-process model before any manufacturing operation is performed.

**In-process model**
The state of the product at some stage in the manufacturing process after an operation is executed. The raw material is the initial in-process model for the first process operation in the manufacturing process. Any subsequent in-process model is created by attaching a work instruction to the in-process model.

**Resource**
The equipment needed for the execution of manufacturing processes, and may include machine tools, robots, and weld guns. A single resource can be used in several operations and processes.

**Work instructions**
Documents that describe how work should be performed. A work instruction documents the procedure by which an operator should perform an activity. Instructions may be printable or accessed from a Web page or are available in Word or PDF format. Manufacturing Process Management allows you to create and manage work instructions in many of the applications using different methods. You can also create templates to ensure work instructions appear in a standard format.
Manufacturing feature
A set of entities and parameters that define generic manufacturing data. For example, a weld point and its normal location comprise a feature that describes where two points may be welded together.

Process plan
The process structure and operations, and the sequence in which they are performed which together describe how a product is manufactured.

Example of product and process structures
The following figure shows a simplified representation of a product and the process to manufacture it. The product contains a component called Part I and the process for manufacturing this component is currently revised to Rev B. The manufacturing process comprises five operations, and operation Op 20 has two tools and a fixture assigned to it. These tools and the fixture are stored in a Resource Manager library, allowing you to reuse them in other operations. The plant where the manufacturing process takes place is also managed in Teamcenter, so that operation Op 20 is assigned to Cell B and operation Op 30 is performed by Lathe 2 in the same cell.

Example of fabrication process
Creating process structure and operations

Manufacturing Process Management provides a view of the manufacturing process that may be structured differently from the product structure. It allows you to link defined processes to the product and provide constraints on the order of process execution.

The design and manufacturing representations of a product structure may be configured differently. For example, the manufacturing view may be divided by areas for a car (front, rear, middle and interior), while the design view is structured functionally. However, the identity of each component remains the same in each view. For example, the left front wheel is part of the wheel subassembly in the design representation; in the manufacturing representation, it appears in the front section of the car, but the absolute occurrence identity remains.

However, each representation may contain information that is specific to design or manufacturing and does not appear in the other representation. For example, the design representation of a car has the doors closed, but in the manufacturing representation, they are open. Different information is stored against the same components, although the components and their absolute identities are unchanged.

The following example shows the process for manufacturing a crankshaft. The manufacturing process comprises assembly and manufacturing steps, which must be executed in the correct sequence.

The hierarchy of the processes and operations is as follows:

Crankshaft Assy

Rod and Piston - a subprocess to capture all activities of manufacturing the rod and piston.

  Assembly - assembly process operation to assemble the rod and piston.
  Rod - a subprocess to capture the manufacturing process of the rod.
    Raw material - a process operation to cut the raw material.
    NC machining rough - an NC machining process for rough cut.
    Heat treatment - the rough part goes through heat treatment.
    NC machining finish - the part is machined to its final dimensions.
    Inspection - the part is inspected.
  Painting - the part is painted.
  End caps - a subprocess to manufacture the end caps.
  Piston - a subprocess to manufacture the piston.

Crankshaft - a subprocess to manufacture the crankshaft.

  Crankshaft assembly
  Cylinder 1 - a subprocess to manufacture cylinder 1.
  Assembly level 1 - a subprocess to manufacture assembly level 1.
    Raw material - cutting the initial raw material.
**Machining** - machining the part to its final shape.

**Inspection** - inspecting the part.

**End 1 [2]** - the alternate process to manufacturing End 1.

**End 2** - the manufacturing process of End 2.

**Assembly Operation** - an assembly process to assemble the four pistons to the crankshaft.

Processes may vary according to the location of their execution, availability of resources, and time. Processes may also be revised independently of the product, in response to feedback from the shop floor personnel when executing the process.

**Defining process operations**

A process operation describes one step in the manufacturing process of a product, and is executed at one work area. Each manufacturing process includes many types of process operations, examples of which include:

- Machining steps
- Painting
- Quality control
- Heat treatment
- Assembly
- Welding

Each process operation has documentation in an appropriate format that describes how the operation is performed, for example, an NC program, work instructions document, or tool list. The manufacturing engineer produces the work instructions as part of the process operation design activity. Documentation can be generated by a software application that is integrated with Teamcenter, for example, you can generate NC programs in NX. The manufacturing engineer may provide tooling instructions and setup procedures with the NC programs. All this information is collectively managed as part of the NC machining process operation.

Similarly, when the manufacturing engineer designs an assembly process operation, the documentation may be in the form of a robot program. The engineer also produces work instructions for setting up the robot and procedures for the operator that must be completed before and after the robot executes the program.

**Defining process operation data**

Each process operation may be associated with one or more of the following data types:

- **Input data**
  
  This data is obtained from one or more of the previous steps of the in-process model.
• **Work instruction data**
  This information is created by the designer of the process operation and contains all the information necessary to complete the work. It may include tooling information, setup information, and the actions necessary to execute the operation. If you do not explicitly define the work area, it may be specified in the work instructions.

• **Output data**
  Output data is generated by applying the work instructions to the input data. The result is a modified in-process model and any instructions necessary for the next steps.

• **Analysis data**
  Analysis data is derived from the work instructions and may include cost, time and a manufacturing features list. You can perform analysis for one process operation or many process operations. The process designer may also set analysis data prior to completing the design to estimate time and cost from the available high-level information.

• **Product**
  You can associate one or more items in the product structure with each process operation. Each item in the product structure should be associated with a manufacturing model that the manufacturing engineer uses to develop the work instructions.

• **Work area**
  You can associate each process operation with one work area. If you do not explicitly assign it to a work area, the requirements may be included in the work instructions and the assignment may be made on the shop floor when the process operation is executed. For example, the process may be performed on a three axis machine, but a specific machine may not be preassigned; shop floor personnel assign a machine according to their availability when the process operation is executed.

• **Equipment**
  The process operation references all equipment required, both the standard equipment of the work area and any additional equipment needed for the specific process operation.

**Defining an NC machining process operation**

When you design an NC machining process operation, you create an NC program, which is utilized by the NC machine to machine manufacturing features. You may also generate setup instructions and a tool list for the machine operator. The generic items associated with an NC process operation include:

• **Machine tools**
  An NC machine that executes the NC program. It must be included in the equipment definitions.

• **Tools**
The cutting tools used in manufacturing operations to cut features.

- **Machining manufacturing features**
  The geometry definitions of the manufacturing operations, such as pocket, hole, slots, ream, and tap.

- **Machining method**
  The types of cuts that are performed during manufacturing operations, such as mill, lathe, mill-rough, mill-finish, center-drill, reaming, and tapping.

- **Program**
  A collection of manufacturing operations in the order of execution.

- **NC manufacturing operation**
  A description of how one tool moves in relation to a feature’s geometry. A tool path can be automatically generated from this description; the tool path, in turn, generates the NC program.

**Defining assembly and weld process operation**

An assembly process operation provides the work instructions for assembling parts in a work area that is designated for assembly operations. Robot programs may also be provided if the assembly process operation uses a robot to perform some assembly actions. The generic items associated with an assembly and weld process operation include:

- **Robots and weld guns**
  This equipment performs assembly instructions.

- **Assembly components**
  Create assembly manufacturing operations to move these items to the correct positions.

- **Assembly method**
  You can define general assembly methods such as weld, glue, and move.

- **Assembly sequence**
  Describes the order in which assembly manufacturing operations are executed.

- **Assembly manufacturing operation**
  Describes how a component moves with one tool or operator during the assembly sequence.

**Aligning properties**

The manufacturing engineer performs certain actions when creating and maintaining a manufacturing bill of materials (MBOM) from an engineering bill of materials (EBOM), namely:
Basic concepts

- Links an EBOM root with an MBOM root in Manufacturing Process Planner or Multi-Structure Manager. When these structures are linked, all further assignments to the MBOM from the EBOM create logically equivalent MBOM occurrences. You can subsequently validate these occurrence links with an alignment check.

You must configure the EBOM and MBOM appropriately, as the root lines of the structures provide the context for assignments. If you override data, the displayed override data is used or updated during the alignment check and synchronization.

This feature uses the AbsOccID property to establish logical equivalence between aligned occurrences. If you want to use this feature, you cannot use the AbsOccID property for any other purpose.

- Assesses the alignment (logical equivalence) of occurrences in the EBOM and MBOM. The results highlight all pairs of occurrences with mismatched data. To do this, you must open the two structures in companion windows, and select a line in each window to set the scope of the alignment check. Teamcenter considers all contexts for the lines in scope. The check compares specific properties of the logically equivalent occurrences that are under the designated scope.

Caution: If you use Multi-Site Collaboration and want to consume a remote EBOM in the MBOM, you must create the AbsOccID properties for the BOM lines to be consumed at the owning site before you export the EBOM to a remote site. Otherwise, an alignment check does not report differences between the structures.

- Keeps the data of aligned occurrences synchronized.

- Propagates alignment when a new MBOM is created from an existing MBOM. This is achieved by replacing the assembly in context in Manufacturing Process Planner or Multi-Structure Manager. This feature allows you to create new MBOM substructures (kits) from existing ones, while preserving the alignment of items within the kit. Typically, the manufacturing engineer does this to create a new subassembly when it is not possible to revise the existing subassembly. All relationships of the items in the kit are preserved, regardless of GRM rules.

- Generates structure alignment reports that display the results of alignment checks that you can configure in the Report Builder application.

For more information about these procedures, see the Manufacturing Process Planner Guide or the Multi-Structure Manager Guide.

Assigning occurrences between structures

When the MBOM planner assigns an occurrence in an EBOM structure (the source) to the MBOM structure (the target), the system:

- Creates a new occurrence with the same child item as the source occurrence in the target structure.

- Copies the other occurrence properties to the target structure. The properties copied are configured by the Teamcenter administrator with preferences, as listed in the Preferences and Environment Variables Reference.
Chapter 2  Basic concepts

The copying action is not tracked in incremental change and cannot be overridden in absolute occurrence context, as it is an essential part of creating the target occurrence. If the structure is packed, data is copied for each of the packed lines, and only the top line of any assigned assemblies.

- Establishes an equivalence between the source and target occurrences by setting the same absolute occurrence identifier on both occurrences.

You can make assignments regardless of the expansion state of the structure; that is, you can make assignments in fully or partially expanded structures. You can make multiple assignments from the source to a single target structure or to several target structures.

Typically, you start with a skeleton or template of the MBOM structure, then make assignments to develop the MBOM in a controlled and reliable manner.

Both an EBOM and MBOM may change after initial assignments have been made. In this case, assignments must be modified to avoid broken links. It is good practice to not create links until the EBOM has reached a stable maturity. However, it may still be in WIP and can be changed without revisioning.

Checking alignment

The alignment check mechanism compares the properties of aligned occurrences and identifies any pairs whose properties do not match. If required, certain properties can be excluded from the alignment check according to preference settings (see the Preferences and Environment Variables Reference for more details). Each mismatch condition must be analyzed manually and the root cause identified before resolution.

The results of the alignment check may be incorrect in the following situations and an attempt to synchronize aligned occurrences in these situations may fail.

- If packed lines are present. You should unpack the structures before performing the alignment check.

- If the precision of aligned occurrences does not match, that is, one occurrence is precise and the other is imprecise.

Children of aligned assemblies are not checked unless they are themselves aligned.

Teamcenter assumes you use same unit of measure (UOM) value for the same item throughout the system. The comparison does not check if the quantity differs between occurrences because different UOMs are used. In this case, the result may indicate a mismatch, but it would be incorrect to update the quantity without also updating the UOM value.

**Note** The alignment check uses the same procedure as the accountability check.

You can optionally create reports in Microsoft Excel format on the status of the alignment of the EBOM and the MBOM. The report format is similar to that provided for the accountability check.

Synchronizing aligned occurrences

When the MBOM planner makes an initial assignment, Teamcenter copies data from the source occurrence to the new target occurrence. The target occurrence is created
with the same child item as the source child item, so that the target occurrence is initialized with the same item and item revision data. Other data on the source occurrence is copied to the target, according to preference settings.

If the source occurrence is subsequently updated, the alignment check identifies a mismatch between properties. The user initiates a synchronization of the structures and the properties of the associated target occurrence are updated accordingly. If any of the property updates fails, all of them are discarded and the user is warned of the error. As before, preference settings control the properties that are synchronized subsequent to an update.

**Managing kits**

Users frequently create new MBOM substructures (*kits*) from existing substructures. Teamcenter preserves the alignment of items within these kits.

An engineering change may lead to a decision to create a new kit, rather than revise the existing kit. This typically happens when content that was consumed in the kit is replaced in the EBOM and the existing kit is still in use on the shop floor. The new kit must be a clone of the old kit, including positioning and other properties. The higher level assembly that contains the kit must also be revised after completion of the cloning process. All content that is common to the existing kit and the new cloned kit retain alignment links to the EBOM.

**Creating structures from templates**

You can create a product structure from a product template, a plant structure from a plant template, and process structure from a process template. You may choose to define a generic product structure that does not represent an actual manufactured product because it contains generic placeholders in place of variable parts. These placeholders are identified by absolute occurrences that are populated when an actual product structure is created from the template.

If you create a generic manufacturing process structure from the generic product structure, the occurrences in the generic product structure are consumed in the generic process structure.

In a template process, although no real parts are consumed in the operations that comprise the generic process, each consumed item has an identity. This identity of the item in the process structure is the same as it is in the product structure. Consequently, if you use this process structure as a template for the creation of a new process structure for manufacturing a product that maps exactly to the generic product structure, the product and process share the same absolute occurrences. The new process that you create from the template consumes the parts from the real product structure with the same absolute occurrences.

Similarly, if you define a mapping between the plant structure referenced by the template and the actual plant structure, generic work areas are substituted with specific absolute occurrences.

For further information about creating structures from templates, see the *Manufacturing Process Planner Guide*, the *Part Planner Guide*, or the *Multi-Structure Manager Guide*.
Defining cloning rules

To clone structures from templates or other structures, you must define cloning rules that allow Manufacturing Process Management to determine how to map objects between the structures. These cloning rules are defined by preferences and the defined rules are available to users when creating an actual structure. The location of the cloning rule preferences for a specific user is determined by a preference. For details about how to set preferences, see the Preferences and Environment Variables Reference.

You can define three cloning rule preferences, one each for process templates, product templates, and plant templates. You can associate multiple action rules with each preference.

When Manufacturing Process Management encounters an object that is mapped between the template and the actual structure, it determines if a substitution is necessary and takes one of four actions, according to the mapping of the object type:

- **Clone**
  Copies the object referenced by the template to a new object in the structure.

- **Reference**
  The structure references the same object as the template.

- **Ignore**
  The new structure does not reference the object.

- **Map**
  The system maps the template to a replacement structure. For example, this allows you to create a process structure from an existing process or from a process template. Optionally, this rule may be allowed by a second (default) action: Reference or Ignore.

Manufacturing Process Management only takes the specified action if an absolute occurrence or its ID is the same in the template and the structure. If the same absolute occurrence appears more than once, only the first matched instance is mapped.

You can specify a second action after the Map action. The second action is the default action and is taken if the mapping fails. For example:

```
*:*:OccType.MEConsumed:Map.Ignore
*:*:OccType.MEWorkArea:Map.Reference
```

In this example, the MEConsumed occurrence type is mapped to the occurrence in the new structure. Manufacturing Process Management uses the absolute occurrence referenced from the template and finds the corresponding occurrence in the new product. It links the consumed items for that operation to the occurrence in the new product. When there is no mapping for the MEConsumed item, Manufacturing Process Management ignores it. When there is no mapping for the MEWorkArea item, Manufacturing Process Management creates a reference to the item from the cloned element.
By default, the **MEConsumed** occurrence type maps consumed items to the corresponding occurrences. The site administrator can change the **controllingOecsForProcessConfiguration** preference to map another occurrence type. Alternatively, this preference may be left unassigned, in which case the configuration of consumed items does not affect the configuration of operations.

Each site can define a set of cloning rules from which users may select. In addition, individual users may change the cloning action rules by editing their preference file.

These rules determine if objects are cloned, referenced or ignored. You cannot change this behavior with the Business Modeler IDE application.

The general format of the cloning rule preferences is as follows:

```
Process.Template=
Action_Rule_Name1
Action_Rule_Name2
.
.
Product.Template=
Action_Rule_Name3
Action_Rule_Name4
.
.
Plant.Template=
Action_Rule_Name5
Action_Rule_Name6
```

For each set of action rules, you must define a new variable. The following example shows a typical **Action_Rule_Name1** variable for the **Process.Template** variable:

```
Process.Template.Action_Rule_Name1=
*:*:Attribute.items_tag:Clone
*:*:Attribute.structure_revisions:Clone
*:*:Attribute.bom_view:Clone
*:*:Attribute.parent_item:Clone
```

The following example shows two cloning rules (**Mapping_Consumes** and **Ignore_Consumes**) for process templates:

```
Process.Template=
Mapping_Consumes
Ignoring_Consumes
Process.Template.Mapping_Consumes=
*:*:Attribute.items_tag:Clone
*:*:Attribute.structure_revisions:Clone
*:*:Attribute.bom_view:Clone
*:*:Attribute.parent_item:Clone
*:*:Attribute.variant_expression_block:Clone
*:*:OccType.Null:Clone
*:*:OccType.MEConsumed:Map.Ignore
*:*:OccType.MEWorkpiece:Ignore
*:*:OccType.*:Reference
*:*:Relation.TC_Rendering:Clone
*:*:Relation.TC_specification:Clone
*:*:class.MEActivity:Relation.TC_MEFolder:Clone
class.MEActivity:*:Attribute.contents:Clone
class.Folder:*:Attribute.contents:Reference
Process.Template.Ignoring_Consumes=
*:*:Attribute.items_tag:Clone
*:*:Attribute.structure_revisions:Clone
*:*:Attribute.bom_view:Clone
*:*:Attribute.parent_item:Clone
```
Chapter 2  Basic concepts

The syntax for adding an entry to a cloning rule is as follows:

```
parent-object:object:relation:action
```

For example:

```
*:type.MEPSDRevision:OccType.+:Reference
```

In this example, the system searches through any parent (*), and when it finds a child of type `MEPSDRevision` (type.MEPSDRevision), with any occurrence relation (OccType.+), it creates a new reference to that MEPSD element (Reference).

### Carrying over future incremental change when cloning

When cloning, you have the option to carry over future effective incremental changes. If you are using date effectivity, future is anything falling after the date set in cloning dialog box. If you are using unit effectivity, future is anything after the unit number set in the cloning dialog box. Incremental changes that are currently effective are merged with the currently set incremental change. Those that are effective in the future are carried forward to the new cloned structure or re-created under new incremental change revisions. Any incremental changes whose effectivities are in the past are dropped from the cloned incremental change.

**Note**

- Specify the release status type that is assigned to any newly created incremental change revisions during cloning in the Clone_Pending_Release_Status preference.

- Specify the release status type that should be considered as secure in the Clone_Secured_Release_Status preference.

In the following figure, the vertical line represents the point of cloning.
Cloning with incremental changes

1 IC1 Incremental change 1 whose effectivity lies in the past.
2 IC2 Incremental change 2 that is secure and configured.
3 IC3 Incremental change 3 is the current IC (set at the bottom of the pane) that is configured and pending. This is the target IC.
4 IC4 Incremental change 4 is pending and unconfigured.

After cloning, Teamcenter:
• Disregards IC1.
• Clones IC2 to a new IC5 and the change elements in IC2 (that is, add or remove) that are associated with the cloned object are cloned and added to IC5. IC5 then has a pending status.
• IC4 is left alone and the new change elements are added to it. This is not cloned because the status is pending and the out effectivity does not match the currently set IC revision (IC3).

Reusing process templates

In many manufacturing environments, process planners design standard processes that are reused several times for different products. For example, the basic structure of the manufacturing process for a car is similar for a basic model or a premium model.

You can use any process as a template for a new structure. However, if you have many of these processes, it can be more efficient to save such manufacturing processes as templates in a library, and then use the templates as the basis of
new manufacturing processes for similar products. The template may include configuration rules that map the product data to process operations, and may include references to standard resources. You can classify the templates in the Classification application for easy retrieval later.

When you create a new process operation, you can browse the available templates and select one to create the process operation. After it is created, you can attach the geometry of the part to be processed.

Similarly, you can create assembly setup templates containing assembly information for a typical assembly setup. When you initialize a setup from a template, Teamcenter loads the related method, equipment, features, and manufacturing operations.

Configuring processes with rules

A manufacturing process is designed for a specific product, and the process definition is tightly integrated with the product definition. Although the manufacturing process has configuration rules that are independent of the product configuration rules, the configuration rules of the product may automatically configure the associated manufacturing processes.

For example, the product structure for a vehicle may include options for a CD or tape player. The manufacturing process for the dashboard mounting includes the following steps:
1. Make the molding.
2. Make the instrument panel.
3. Assemble the molding and instrument panel together.
4. Buy or make the tape player.
5. Buy or make the CD player.
6. Assemble the tape player into the vehicle.
7. Assemble the CD player into the vehicle.

When the tape player is selected, processes 1, 2, 3, 4, and 6 can be configured automatically in the process definition. When the CD player is selected, processes 1, 2, 3, 5, and 7 can be configured.

You can also use the product revision rules to control the process structure. For example, the product includes occurrence A, which has two revisions, Working and Released. Each revision has a different manufacturing process. If the revision rule for the product configures Working revisions, you can use the rule to automatically configure the appropriate manufacturing process for building the working revision. You define and apply revision rules for the product in Structure Manager, as described in the Structure Manager Guide.

You can configure processes to accommodate variations in product structure in several ways:

- Variant rules
You can define options at the top level, then define rules for each process derived from those options. For example, you can specify a different process for the manufacture of item **End 1** according to the process capabilities of the location.

- **Revision rules**
  You can load appropriate revisions of processes by applying predefined revision rules.

- **Effectivity rules**
  You can set effectivity dates on processes in a similar way to which they can be applied to the product structure. You can also define a range of unit or product serial numbers against which the process is effective.

- **Substitutes**
  You can define substitute processes and select a preferred substitute.

### Configuring processes with variants

Variant configuration allows you to create options (such as manufacturing locations) and allowed values (such as a specific plant within the company) and associate these options with an item revision. You would normally apply this to the top-level structure, but you can apply options anywhere in the process structure.

In the Manufacturing Process Management environment, you can configure variants to the manufacturing process to accommodate different configurations of the product. These differences are typically configured as variants of the product design. If so, you can configure manufacturing variants that correspond to specific variants of the product design.

When you configure variants, you define the following parameters:

- **Option**
  The option is an attribute of the item revision with a set of allowed values; for example, if you manufacture two sizes of engine, there may be two allowed values—1200 and 1600. An option is attached to a specific revision of an item, such as a process or operation. Option names are unique within an item revision, but may not be unique within the entire system.

- **Variant condition**
  A condition that you set on an occurrence to specify what option values are required to configure that occurrence. The variant condition is defined programmatically, for example:

  ```plaintext
  Load IF engine = 1200
  ```

- **Variant rule**
  A collection of option values that define programmatically the variant of the structure to configure, for example:

  ```plaintext
  car type = GLS, engine = 1200, gearbox = manual
  ```

- **Fixed default**
A specific default value that is applied if no other values are specified. For example, if the default product manufacture is the 1200 engine and you rarely build other sizes, you may set a fixed default of:

\[
\text{engine} = 1200
\]

A fixed default is attached to an item revision of a process, operation, or other item.

For more information about variants, see *Getting Started with Product Structure*.

### Creating in-context data (absolute occurrences)

Absolute occurrences allow you to manage data that is unique to a specific application of an item in a process or product structure. Technically, an absolute occurrence represents a specific line in the context of the top line item.

For example, you manufacture a vehicle that is designed with four identical wheel and tire subassemblies. However, the front tires must be inflated to 33 PSI, while the rear tires must be inflated to 30 PSI. To accommodate this difference, you can create absolute occurrences of these subassemblies in the manufacturing environment with the required pressure set on each absolute occurrence.

By default, creation of absolute occurrences is not enabled and the user must select absolute occurrence editing mode to permit this. Any changes affect only the current context, that is, the structure currently displayed.

You can edit BOM line property values in context to create an absolute occurrence. You can also attach forms and datasets to an absolute occurrence, for example, a specification describing the reason for the tire pressure change.

For more information about absolute occurrences, see the *Manufacturing Process Planner Guide*.

### Using compositions

A composition is a special kind of structure context that allows components to be added from one or more structure contexts, each of which may contain a different product structure. A composition typically models a scenario or manufacturing process. For example, you can place a product view of an assembly and the corresponding process view in a composition for review and approval purposes.

A composition may contain any of the following:

- An instance of the representation of any top-level item. In this case, the composition includes the complete representation.

- Instances of occurrences from other representations. A new instance is created and associated with an occurrence in the source representation.

- Instances of occurrence groups. An instance of the representation of the top-level item is created and filtered to show only the occurrences that are members of the occurrence group (with its subgroups and members).

When you create a composition, you can manage and edit the occurrence groups in it in the same way as any other line in the structure, including:
• Attaching data to occurrence groups in the context of the top line composition or any other context. This action creates a persistent absolute occurrence and the occurrence path representing the line in the composition.

• Attaching a 3D markup to them in the viewer.

• Changing their position in any context. If you change the position of a member of the group, the in-context edit must be set to the appropriate context. If no editing context is defined, the relative position is changed.

• Defining item elements (GDEs) and connections as part of the occurrence group.

• Creating connection interfaces on the members of the occurrence group.

• Comparing occurrence groups and their members.

• Creating and retrieving 2D product views (snapshots) in compositions containing instances of occurrence groups.

• Creating, editing, and manipulating graphics.

• Using any member of an occurrence group as the top level context for creating and editing absolute occurrences.

• Configuring members of an occurrence group to match the configuration of the source view. If a member does not exist in the source, it is grayed out in the composition.

If additions, removals or changes are made to the source line, the occurrence line in the composition is updated accordingly.

If an occurrence is a member of several occurrence groups in a single hierarchy, it is treated as a single occurrence for all purposes.

**Associating data in the context of a composition**

When you associate absolute occurrence data to a member of an occurrence group in a composition, Teamcenter applies the following rules to determine the context of the association:

• If the selected line is a line higher than the instance of the occurrence group that brought this member into the composition, the selected line itself is the context of the association.

• If the selected line is the instance of the occurrence group that brought this member into the composition, the root item of the source structure of the group is the context of the association. Consequently, if multiple instances of the association are referenced in the composition, the same data is attached to the same member in every instance of the occurrence group.

• If the selected line is a subgroup in the instance of the occurrence group that brought this member into the composition, the context of the association is the root item of the source structure of the occurrence group. A subgroup cannot be a context if it does not exist in the hierarchy of the source structure.
Chapter 2  

Basic concepts

- If the selected line is a subassembly (not a subgroup) in the instance of the occurrence group that brought this member into the composition, the selected line is the context of the association.

- If no line is selected as the context, the actual parent of this member in the source structure is the context of association.

The line you select as the context must be a parent or ancestor of that member on the composition.

Configuring referenced structures in a composition

You can configure members of a referenced structure in a composition from the configuration of the source structure. A referenced structure may be a subassembly, single component, or occurrence group. This may be done in one of the following ways:

- If a configuration context is stored on the line of the referenced structure (by association with its occurrence or absolute occurrence), the referenced structure is configured by this configuration context. In general, any configuration context associated with a higher level overrides one attached at a lower level.

- If there is no stored configuration context, but the referenced structure is open and active in the source structure, the reference structure takes the same configuration as the one in the source structure.

- If there is no stored configuration context and the source structure is not open, the referenced structure does not appear in the composition.

Note  The final configuration of the referenced structure and its components also depends on the configuration of the composition. A subassembly or component is configured in a composition if it is successfully configured in both the composition and the source structure.

Revising processes and operations

Each manufacturing process and operation is represented in the system by an item. The item contains the information about the process or operation that does not change; for example, a single item may represent the operation in which the crankshaft is assembled onto the engine. Each item has one or more revisions that contain information that changes. For example, the item revision of an operation may specify a particular operation that requires a specific welding machine; if you change the operation to use a different machine, you revise the operation and create a new item revision.

Revisions have a particular effectivity, expressed as a date range or sequence of product serial numbers. When you revise an item, you can specify that the new process or operation takes effect on a particular date or product number. You can define revision rules that determine the revision of the item that is loaded in any particular situation.
Managing incremental changes

You can control changes to the manufacturing process, structure, or operations by revising the affected items. As many changes may be small and not related to each other, Manufacturing Process Management allows you to create incremental changes to add components to and remove them from the structure. Each incremental change consequently comprises several individual and possibly unrelated revisions to a component, occurrence, or attachment in the structure.

You can define an effectivity for an incremental change, allowing (for example) all changes related to the production of a new model to be implemented simultaneously. You normally assign an effectivity expressed as a date range or sequence of serial numbers to the change, ensuring that all grouped additions and removals happen at the same time.

An incremental change may have revisions, and you can configure the various components, occurrences and attachments to a specific revision by applying appropriate revision rules.

You can also create intents to an incremental change. An intent represents an event or alternate solution that is not bound to a date or unit number effectivity. For example, you may define an intent as a new prototype and group all changes required to build the prototype accordingly.

You can attach relevant data to an incremental change, including a form, dataset, or folder. For example, you may want to attach the individual change requests that are satisfied by the incremental change.

You can also use incremental change to view the definition of a structure effective of a certain date or unit number. For example, the structure of part 100 may have been originally designed with two occurrences, one of part 200 and the other of part 300. A design change is made that cancels the occurrence of part 300 and adds an occurrence of part 400 in its place. These removals and additions remain linked to the incremental change that carries the effectivity of the change. If you view the structure of part 100 for an effective date after the change occurred, you see only parts 200 and 400.

You can also baseline changes by rolling up several incremental changes that apply to a certain date or unit number into a new single baseline revision. You can then apply further incremental changes to the baseline.

For more information about incremental changes, see Getting Started with Product Structure.

Tracking changes

When data changes in the context of an incremental change, these changes are recorded against the specific incremental change. These individual change elements, such as the addition of an occurrence or removal of an attachment, are not visible to the user but may be accessed by advanced users.

Changes that are tracked and recorded against an incremental change are:

- Addition of activities to an operation.
- Removal of activities from an operation.
Chapter 2  

Basic concepts

- Addition and removal of attachments to an operation, including forms and datasets.
- Creation of data such as forms and datasets.
- Changes to predecessor occurrences of processes or operations.

Changes are normally tracked dynamically, as the user makes edits with an active incremental change. Changes may also be tracked retrospectively, if the changes are already made but were not originally tracked against an incremental change.

Configuring changes

You can release a revision of an incremental change with an appropriate workflow. You define a release status (typically Released) that applies the defined effectivity.

**Note**  You can also release a revision of the incremental change with a Preliminary status, which allows you to still modify the incremental change. If Released status is applied, you can no longer modify the incremental change.

The user can choose to view all configurations associated with an incremental change or to hide those components and attachments that are not configured by the current revision rule. The configuration rules that are applied to a component or attachment are as follows:

- If an occurrence has an associated change and the incremental change associated with that change is configured, the change is applied.
- If more than one change affects an occurrence according to effectivity, a removal is configured in preference to an addition.

Configuring with intents

You can define intents to represent milestones or events that cannot be defined by a date or unit number effectivity. For example, you may configure several alternative processes for evaluation and discussion and identify them as Alternate Process 1, Alternate Process 2, and so on.

You can optionally apply one or more intents to an incremental change, as follows:

- If you apply several intents to an incremental change, the incremental change is valid for all of those intents.
- An intent is applied to all revisions of the incremental change.
- Intents may optionally be configured by a revision rule. Any incremental change that matches an intent set in the revision rule is selected.
- You can configure an incremental change with both intents and effectivity, when appropriate.
Controlling access

To create incremental changes against a structure, a user must have access to the revision of the incremental change. The Teamcenter administrator controls this access with the Access Manager application, as described in the Access Manager Guide.

Understanding relations versus occurrences

When you associate the top-level product with a process, or top-level plant and process, the linked object are associated as relations. Relations are not part of the assembly structure. They are listed under the attachments tab.

In the following figure, Product Root is linked to Process Root using a Mfg. Targets relation.

Occurrences are part of a BOM structure, meaning they are part of an assembly. In the manufacturing world, these are typically product, plant, or process structures. Each BOM line is represented by a revision of a specific item type (for example, item, MEWorkArea, MELine, MEProcess, MEOP). These BOM lines can be assigned by a specific occurrence type.

<table>
<thead>
<tr>
<th>Occurrence type</th>
<th>Used typically to represent</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEWorkPiece</td>
<td>In-process workpiece (input geometry)</td>
</tr>
<tr>
<td>MEResource</td>
<td>Fixtures</td>
</tr>
<tr>
<td>MEMachineTool</td>
<td>Machine</td>
</tr>
<tr>
<td>METool</td>
<td>Tool assemblies</td>
</tr>
<tr>
<td>MEWorkArea</td>
<td>Assigned work area (subelement of the plant structure)</td>
</tr>
<tr>
<td>MEConsumed</td>
<td>Consumed items for the assembly process (subelement of the product)</td>
</tr>
<tr>
<td>METarget</td>
<td>Part being machined in part planning (assigned automatically by NX CAM using the master model concept)</td>
</tr>
</tbody>
</table>

When you consume products in a process plan, the occurrences are added as a BOM line to the process structure with a specific occurrence type. By default, for instance, when you consume a product in a process, the product gets added to the process structure with an MEConsumed occurrence type.

The following figure shows consumed items in a process structure.
For more information about consuming, see the *Manufacturing Process Planner Guide*.

When you assign a resource to a process plan, it gets assigned with an **MEResource** occurrence type. You can change the occurrence type of an assigned resource in the Classification Search Dialog.

For more information, see the *Manufacturing Process Planner Guide*.

You can see which occurrence type an object has in the **Occurrence Type** column.

**Tip**

You can change the occurrence type in the column cell by clicking the cell and selecting from the **Occurrence Type** list.

When you work with the NX CAM Integration, NX automatically adds the product to the operation using an **METarget** occurrence type. In the following figure, the **camera_clip_core** product was added to the process structure using an **METarget** occurrence type.
Reconciling broken links

Changes to the product or plant structure are not automatically updated in the process tree where the occurrence is referenced. This results in a *broken* link. With this feature, you can identify broken links and search for the possible occurrence that was originally defined, for example, as a consumed part.

When you expand a process to display linked items, each link line includes a symbol indicating the state of the reference:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:image:</td>
<td>Occurrence is linked to a product or plant structure.</td>
</tr>
<tr>
<td>:image:</td>
<td>Link is broken.</td>
</tr>
<tr>
<td>:image:</td>
<td>Occurrence is linked to a product or plant structure that is not currently configuring the process structure.</td>
</tr>
</tbody>
</table>

*Note* Teamcenter retrieves the status of a link from a run-time property, `bl_occ_assigned`, that you can also display as a column, *Assigned Occurrence*.

When performing the search for broken links, Teamcenter searches through process structures that you specify and looks for broken links. It then looks through product or plant structures that you specify for likely *candidates* to repair these links. The search is based on criteria that you can also specify, whereby at least the item ID must always match. Once found, you can have Teamcenter repair these links automatically, or you can choose to select manually from the candidates list.

When there are many broken links and each broken link has many candidates, a large amount of memory is required for high-speed processing. If the total number of broken links and candidates is too large, for example, 32,000 lines, the memory usage may be a concern if you are performing the search on an underpowered computer. To speed up the search, divide the search scope into smaller pieces to reduce memory usage; for example, select subnodes instead of the root node.

Controlling access

Manufacturing processes and operations are represented as *items*, allowing the Teamcenter administrator to define appropriate access control privileges to the manufacturing processes. For example:

- The manufacturing engineer or process planner should have full permissions to access and modify process data. Users in this group can change process operation sequences, and define the configurations of processes.

- Shop floor (assembly) personnel have read-only access to process data. Users in this group can view process data such as work instructions, but not make changes.

- Access may be further refined by subgroup. For example, an NC machinist may only have access to NC machining process operations.
You define access permissions in access control lists (ACLs) with the Access Manager application.

Managing resources

Process operations require resources to execute them. The availability of resources affects the definition of manufacturing processes. For example, in an automated assembly plant, robots may weld assemblies. In plants where weld robots are unavailable, the same manufacturing process may be followed, but a different resource (a manual welder) executes the operation.

A process operation can directly identify that a specific work area in the resource structure is needed to execute it. In the case of a weld process operation, it may specify a weld workstation on the shop floor and a specific weld gun as the resources required for its execution. You can use the assigned resources and the process requirements to estimate the time needed for execution. In locations such as assembly plants where meeting throughput requirements is critical, the process definition influences the quality and layout of the resources.

Process operations may only specify a requirement of the required resources. For example, any arc welding robot on the shop floor can execute an arc welding process. In this case, the weld process operation does not directly reference specific instances of the weld work area and weld guns. Manufacturing execution systems (MESs) marshal the process definitions and the product order size, and assign jobs to specific resources according to their availability. The process planner may receive feedback from the MES system and modify the process definition to improve performance. For example, the planner might decide to buy material from external suppliers instead of manufacturing in-house to avoid resource constraints.

Users of resources fall into three general categories:

- **Machine tools**
  An NC machine that executes the NC program. It must be included in the equipment definitions.

- **Resource authors**
  Users with this role define new resource components, assemble components into standard tool assemblies, and define the attributes and properties of the components and assemblies.

- **Resource consumers**
  Users with this role are manufacturing engineers who define process plans that utilize resources already defined by the authors. They may also be plant workers who use the process plans and update the resource inventory.

For more information, see *Planning resource management*.

Releasing a manufacturing process

After you complete a manufacturing process, you may want to make it available for review and approval before manufacturing commences. Teamcenter does not provide
review, approval and release processes, but you can develop suitable processes with the Workflow application, as described in the Workflow Designer Guide.

You can also provide intermediate releases of work-in-progress process designs for others to view. This intermediate snapshot of data is sometimes referred to as the preliminary data indicator (PDI). Consumers of a PDI can then work with the data, knowing that it will not change until the next PDI is released.

The process and its operations may reference occurrences of product data and plant data. The PDI may not be valid if this related data changes. Consequently, if you create a baseline of the process in this way, you must also baseline any related structures at the same time. Teamcenter allows you to baseline a process structure (or a composition) and create a PDI in a single operation.

Creating product views

You can save the configuration of the assembly viewer, including the current selection of objects, zoom factor, rotation angle, and pan displacement. The saved configuration or product view (sometimes called a snapshot view) is attached to the product, process, or plant structure that is associated with the view. Teamcenter assigns a thumbnail image to each saved configuration, allowing users to browse and retrieve configurations.

For information about managing and using product views, see the Manufacturing Process Planner Guide.

To configure the product views feature, edit the com/teamcenter/rac/cme/snapshot/snapshot.properties file. If you change any of the settings in this file, exit and restart the rich client.

You can change:

- The height in pixels of thumbnails, as defined by the thumbHeight setting.

- The width in pixels of thumbnails, as defined by the thumbWidth setting.

  Note You must set the height and width to the same value.

- The quality of thumbnails, as defined by the thumbQuality setting. Defined as a relative value between 0 (low) and 100 (high).

- Product views are sorted alphabetically in the viewer, according to the name of the dataset. If the snapShotDataset.case.SensitiveSort setting is false, the system ignores the case of the name; if it is set to true, names with uppercase initial letters appear before names with lower case initial letters (for example, Apple is listed before apple).

- When product views are created, the system assigns them a numeric value derived from the date and time they were created. To have new product views appear at the end of the list, set the snapShotDataset.numbersSortLast setting to true; to have new product views appear at the beginning of the list, set the snapShotDataset.numbersSortLast setting to false.
Managing weld points (features)

Teamcenter allows you to include manufacturing features such as weld points and datums as part of the process structure. Features can be attached to one or more physical parts; for example, a weld point can be attached to two different components if they are welded together during the manufacturing process.

You can include weld points in occurrence groups, search for and compare them, and check they are consumed in the process structure.

For more information, see the *Manufacturing Process Planner Guide*.

Managing process design in the thin client

You can manage the design or modification of manufacturing processes with the thin client interface, even if other participants in these activities are working with the rich client applications. A typical scenario is as follows:

1. The project manager defines the task and submits it to a workflow. The workflow assigns the task to a manufacturing engineer.

2. The manufacturing engineer receives the assignment and completes the necessary work in rich client applications such as Manufacturing Process Planner. On completion, the engineer associates the affected items (for example, attachments and reports) to the workflow assignment and approves it.

3. The project manager receives notification in the thin client worklist that the workflow assignment is approved and sent back.

4. The project manager reviews the items attached to the workflow assignment and, if they are satisfactory, approves the assignment as complete. If not, the project manager may return the assignment to the manufacturing engineer for rework.

Setting manufacturing preferences

You can set manufacturing preferences in the same way you set all Teamcenter preferences:

- Set the preferences in the preferences hierarchy.
  1. Choose Edit→Options.
     Teamcenter opens the Options dialog box.
  2. Click Manufacturing in the hierarchy tree.
     Teamcenter displays tabbed pages that provide preferences pertaining to different views.

     Note: Not all manufacturing preferences are available in the preference hierarchy.

- Search for and set individual preferences.
  1. Choose Edit→Options.
Teamcenter opens the **Options** dialog box.

2. Open the **Index** tab.

3. In the **Search on Preference Name** box, type the name of the preference you want to modify, or select **Manufacturing** from the **Category** list.

   **Note** Many manufacturing preferences begin with the ME prefix. This stands for **Manufacturing Environment**.

For more information about modifying preferences, see the *Preferences and Environment Variables Reference*.

**Integrating through Multi-Structure Manager**

Tecnomatix process design software allows you to design manufacturing processes and manage them at the point of execution. As such, it shares many data items with Teamcenter, in which the data is linked to the product design. You can integrate the two systems, so that Tecnomatix and Teamcenter can share common data. To ensure the shared data is always synchronized, you can define a **collaboration context**, in which the shared data resides.

The collaboration context contains one or more **structure contexts**, which are folders, each containing the definition of a shared product, process, or plant structure. Users can view and manipulate the shared data with Manufacturing Process Planner or the Multi-Structure Manager application, while administrators can manage the synchronization process with the Application Interface Viewer in My Teamcenter.

To obtain these capabilities, you must install the comprehensive manufacturing data types when you run Teamcenter Environment Manager (TEM).
Chapter

3 Basic tasks

Basic tasks ................................................................. 3-1
Creating a factory structure ............................................. 3-1
  Importing data with the In Context Editor (ICE) .................. 3-1
Creating a manufacturing process ..................................... 3-2
  Open the product ...................................................... 3-2
  Assign work areas .................................................... 3-2
  Create the process structure ....................................... 3-3
  Creating operations .................................................. 3-4
    Create an operation ................................................. 3-4
  Establish relationships between process, product, and work areas 3-5
  Assign activities to operations .................................... 3-6
  Verify the manufacturing process .................................. 3-6
  Add resources to operations ....................................... 3-7
  Importing structures from Excel ................................... 3-7
    Import structure from an Excel spreadsheet .................... 3-7
Creating a preliminary data indicator (baseline) .................. 3-8
Create an intermediate data capture ................................ 3-8
Performing process simulations ..................................... 3-9
Design a fixture for manufacturing .................................. 3-11
Creating reports ......................................................... 3-11
  Creating reports in the rich client ............................... 3-11
    View or print rich client reports with Web browser ............ 3-12
    Configure reports in the rich client ........................... 3-12
    Troubleshooting reports in the rich client ..................... 3-14
  Creating reports for the thin client .............................. 3-16
    Configure thin client report generation ......................... 3-16
    Set preferences for thin client reports ........................ 3-16
    Update report designs ............................................. 3-18
    Run batch reports ................................................ 3-18
Displaying GD&T symbols and rich text in forms .................. 3-19
  Configure GD&T symbols and rich text in forms ................. 3-19
  Generating PDF reports ............................................. 3-21
Basic tasks

There are many basic usage and administrative tasks that apply to planning the manufacturing of both assemblies and parts. You can create manufacturing objects, such as processes, products, or plant structures. You can create baselines or intermediate data captures to preserve specific states of these structures. You can also configure the manufacturing environment to create reports, use methods-time measurement (MTM) data cards, publish manufacturing data or create standard text.

Creating a factory structure

Manufacturing Process Management allows you to design, modify, import, and export the structure of the factory in which you build the product. You can use the Plant Designer application to provide a view of the plant structure layout and to organize the areas where the product is manufactured and assembled.

Plant Designer allows you to define various Workarea types. These could include everything that is specific to a particular manufacturing facility; for example MESStations, MELines, and MEDepartments.

Importing data with the In Context Editor (ICE)

The In Context Editor (ICE) is an optional Siemens PLM Software addition to the AutoCAD Architectural Desktop product that allows you to design and manage plant structures and layouts. For example, you can create the detailed design of the factory structure in AutoCAD and ICE, import it into Plant Designer, and manage the design data as follows:

1. Store model files in DWG format or as JT files for visualization. Individual drawings may correspond to specific plant types, and the JT files created for the drawings can be attached to them, making them available in the Graphics view.

2. Configure, revise, and view the plant structure.

3. Create and edit the plant structure. You can break the structure into logical groups by station, section, or department.

When you import a drawing, you can specify the work area type to create at that time. You can subsequently add or remove work area items from the factory structure in the Plant Designer application. You can then export the modified plant structure to the In Context Editor and locate added work area items.
Creating a manufacturing process

Follow these general procedures to design a manufacturing process for a part in an existing product. The examples describe a manufacturing process for a right crank arm, which is part of a crank assembly that was designed in NX.

Open the product

When the design engineer releases the design, you should begin the planning process as follows:

1. Set the options in Teamcenter Integration for NX to save images in JT format. This allows you to visualize representations of parts in the assembly viewer in Manufacturing Process Management.

2. Locate the product assembly with My Teamcenter.

3. Send the product to Structure Manager and verify its assembly structure. Structure Manager allows you to manage assemblies, parts, and components before you begin the planning process.

4. Identify the component to manufacture in the assembly, and send the component to Manufacturing Process Planner or Part Planner. The component is visible in the BOM pane and the assembly viewer.

   **Note** Lines with a NoLoad reference designator are not loaded during the manufacturing process.

Assign work areas

After you identify the component to manufacture, you can then assign it to a location where it will be manufactured, as follows:

1. Locate the top-level work area and open it in Manufacturing Process Planner or Part Planner. Click the plant structure tab and click **Open Work Area by name** 🌘.

   Teamcenter displays the **Open Work Area** dialog box, allowing you to search for the applicable area and assign it to the product.

   If the work area does not exist, create it, as follows:

   a. Select the parent work area in the BOM hierarchy and click **Create New Work Area**.

      Teamcenter displays the **New Work Area** dialog box.

   b. Click **Assign** and Teamcenter assigns a unique work area identifier and revision number.

   c. Type a name and description for the new work area.

   d. Select **MEPlant** from the **Workarea Type** list.

   e. Click **OK** to create the work area.
2. Select the facility to fabricate the component. A list of available facilities is shown in the BOM pane when you click the plant structure tab. You can expand the hierarchy of the facility to view departments, work centers, and work areas that are configured within it.

3. Optionally, assign a substitute facility to fabricate the component. A substitute facility has similar capacity and equipment as the primary facility, and can fabricate the component, for example, when there are resource constraints at the primary facility.

   If the primary facility has a substitute, a symbol appears next to it in the hierarchy.

   To designate a substitute facility, right-click the primary facility, choose **Set Preferred Substitute**, and choose the substitute from the list of available facilities.

   You can create an additional substitute facility by clicking the button in the bottom toolbar and selecting a facility from the list.

4. View the facility layout. You can generate plant layouts in FactoryCAD, save the images as JT files, and attach the images to the appropriate work area. You can view the image in the assembly viewer when you select the work area in the hierarchy.

**Create the process structure**

When you have data about the product and the work area where you will fabricate it, you can design the manufacturing processes. The manufacturing process is a separate Teamcenter item that can be revised and controlled separately from, but is closely related to, the product and work areas. For example, you can design a manufacturing process for the fabrication of the right crank arm of the crankshaft assembly, and then define manufacturing processes for each of the other components.

1. Choose **File** → **New** → **Process** or click the **Create a New Process** button. Teamcenter displays the **New Process** dialog box.

2. Create the new process as follows:
   a. Choose a process type from the list, for example, **MFabrication** and click **Next**.
   b. Click **Assign** and Teamcenter assigns a unique process identifier and revision number.
   c. Type a name and description of the new process.

3. When you have completed the new process definition, click **OK** or **Apply** to create it.

   You can also save manufacturing processes as templates, and use the templates to create processes for similar products. You can organize and store such templates with the Classification application.
Creating operations

An operation is one step in the manufacturing process for a product and is executed at a single workstation. There are many different types of operation you can attach to the manufacturing process, including:

- Procurement of raw material
  For example, a blank forging may be procured from a storage area.
- Machining
  For example, the forging may be roughly machined on a CNC machine.
- Quality control
  For example, the forging is inspected prior to finishing.
- Finish machining
  For example, the forging may be finished on a CNC machine.
- Finishing
  For example, the machined part is coated, inspected, and bench finished.

After you establish the basic hierarchy of processes and operations, you can designate alternate operations if necessary. You can also use the PERT view to resequence operations within the process.

Create an operation

1. Select the top-level process and click the New Operation button.
   Teamcenter displays the Create New Operation dialog box.

2. Create the new operation as follows:
   a. Choose an operation type from the list, for example, MEPProcurement for a material procurement operation and click Next.
   b. Click the Assign button.
      Teamcenter assigns a unique operation identifier and revision number.
   c. Type a name and description of the new operation.

3. After you complete the new operation definition, click OK or Apply to create it.

4. With the top-level operation still selected, repeat each of the previous steps to create the other operations for the manufacturing process.

   Designate machining operations as one of the operation types that your administrator specified in the NX_supported_operation_types preference. This ensures that Teamcenter Integration for NX recognizes them when you generate CAM data.

5. If appropriate, create and designate substitute operations and select a preferred substitute, as follows:
Basic tasks

a. Create the substitute operation, following the previous steps. If it is similar to the primary operation, you can use the primary operation as a template for the substitute operation definition.

b. Select and right-click the substitute operation.

c. Choose Set Preferred Substitute, then choose the substitute from the list of available operations.

d. You can create additional substitute operations by clicking the 🧠 button in the bottom toolbar and selecting them from the list.

6. If necessary, adjust the sequence of operations in the process by opening the PERT view and moving the connections between operations.

Establish relationships between process, product, and work areas

After you create work areas and a manufacturing process containing operations, you must link them to the product, as follows:

Note The following example assumes you are working with one process, product, and plant structure. Depending on your data, you may have multiple structures of the same type. Each structure appears in its own view.

1. Assign the process to the product. This ensures any changes to the product configuration are reflected in the manufacturing structure. Similarly, any changes to the revision rules may affect variant and alternate operations in the process. To assign a process to a product:

   a. Load the product and process structures.

   b. Right-click the process root and choose Link/Associate→Associate Product as Target.

   c. In the Associate Product as Target dialog box, select the product to establish the relationship between the product and process. When you associate a product and process in this way, both structures are imported into Manufacturing Process Planner or Part Planner when you open one of them.

      Note You can determine if a relationship exists between a process and a product by selecting the process and clicking the Attachments tab. If the relationship exists, the product item is shown as an attachment to the process revision.

2. Assign the process to a work area, as follows:

   a. Open a work area.

   b. Right-click the process and choose Link/Associate→Associate Workarea to establish the relationship between the work area and process. When you associate a work area and process in this way, both structures are imported into Manufacturing Process Planner or Part Planner when you open one of them.

3. Assign each individual operation to a work area, as follows:
a. Select the operation in the process and click **Copy**.

b. Select the work area and click **Paste**.

Once the product, process, and work areas are related, you can add the structures to a collaboration context. You can easily retrieve and open the data in the collaboration context from Manufacturing Process Planner in future sessions.

**Assign activities to operations**

You can further define a process operation by creating a series of *activities* within it. For example, an NC machining operation may include activities to set up the part in a fixture, download and run a CNC program, and inspect the machined part.

**Note** An activity is not an item in Teamcenter and cannot be revised in the way that processes and operations can. Do not create activities until their definitions are stable and unlikely to change.

You can create activities in several ways:

- In the **Activities** section of the **Time** pane.
  You can either create activities manually, or use a data card to choose from predefined activities.

- In the **Operation Activities** pane.

**Verify the manufacturing process**

Teamcenter includes several analysis tools that allow you to check the completeness of the process, and identify discrepancies between the product structure and the process structure. You can make the following checks:

1. Select a line in the product structure and search to see where this item is consumed in the manufacturing process.

2. Select a searchable item in the process and find it in the original product structure.

3. Interactively compare two structures to identify differences and similarities. You can expand the structures, select an assembly or subassembly, then initiate an interactive comparison. The results of the comparison are color-coded in the structure.

4. Compare the product and the process to ensure all occurrences of product components are consumed in the process structure. You can save the results for later analysis.

For more information about performing these tasks, see the *Manufacturing Process Planner Guide* or the *Part Planner Guide*. 
Add resources to operations

You can associate one or more resources with an operation or activity, which are required for its execution. You can retrieve resources that are stored in the Classification application and attach them to the operation, as follows:

1. Select the top-level operation in the process structure and click the Assign resources to operation/work area button. Teamcenter opens the Classification application.

2. Expand the classification folders until you find the appropriate subclass.

3. Double-click the subclass to expand its structure.

4. Click the Search button to search for the resources by name or characteristic. After the search completes, click the Table tab to display a list of resources matching the search.

5. Select the resources to attach to the operation and click OK.

For more information about managing classified resources, see the Classification Guide and the Classification Administration Guide.

Importing structures from Excel

You can use the Microsoft Excel import utility to populate structures in Teamcenter from definitions in an Excel spreadsheet. Typically, you import process structures, such as build sequences or process plans, but the utility can also import any type of product, process, plant, and resource information. Additionally, it allows you to:

- Link between multiple structures.
- Assign relations (consumed, required, work area, and resources).
- Attach forms and fill in attribute values.
- Modify ownership.
- Define activities (with time information).
- Attach variant information.

For more information about how to use the tcexcel_import utility, see the Manufacturing Process Planner Guide.

Import structure from an Excel spreadsheet

1. Create an Excel file with specific headings. The import utility recognizes specific headings and formats in the Excel file and translates them into Teamcenter structures.

2. Save the Excel file as a tab-delineated text file.
3. Import the Excel file into the Teamcenter database using the `teexcel_import` utility.

Creating a preliminary data indicator (baseline)

You can create a preliminary data indicator (PDI) or baseline of a manufacturing process that is in development. You can then release the PDI to other users and external suppliers, who can then complete associated work on the basis of the data in the PDI. You would normally release PDIs at defined intervals, so that users know the effectiveness of the data provided.

Before creating baselines, you must:

1. Use the Workflow application to define a baseline release procedure (for example, **PDI**) that creates and adds an appropriate status to structures. Do not define any tasks.

2. Use Business Modeler IDE to create a naming rule (for example, **BaselineSuffixRule**) and attach it to the item types (for example, **BaselineSuffix**).

3. Use Business Modeler IDE to create the necessary status (for example, **PDI**).

4. Edit the **Baseline** preference to identify the baseline procedure you want to use.

Before you baseline a process structure, you must baseline the associated product structure and plant structure. This ensures you can reproduce the process structure if necessary, by setting the product and plant configuration rules to match the rules applied when you created the process baseline. (To reproduce the exact baselined process or composition, the product structure and plant structure must have a baseline or release status.)

**Note** You cannot release **product** data in the manufacturing environment. Release product data with Structure Manager.

Typically, you would not release individual processes or operations, although Teamcenter does not constrain you from doing this. The processes and operations are related in the in-process assembly, and you would typically release them as a set.

If you baseline a process, Teamcenter does not attempt to baseline consumed parts or assigned work areas, as it assumes the manufacturing engineer does not own this data.

Create an intermediate data capture

You can capture the state of any structure or part of a structure for subsequent retrieval and viewing. This capture data does not represent the final released state of the structure, so it is referred to as an **intermediate data capture** (IDC). The configuration rules are saved with the structure allowing its exact state at the time of capture to be reproduced each time it is retrieved. Creating an IDC does not affect any subsequent changes to the structure or its release by a Workflow process.

An example scenario in which you use an IDC follows:
1. Initiate a project to validate how a product is manufactured and create a collaboration context to store the data needed.

2. Collect all occurrences of interest from the product and process structures and put them in structure contexts in the collaboration context.

3. Set the configuration rule for the product data and save it with the structure context.

4. Create a composition structure, add it to the new structure context, and instantiate the product occurrences into the composition.

5. Add other nodes needed for the project to the structure context such as resources, parts from other products, and documents.

6. Create a document that describes the conclusions of the project and save it with the composition. The document may be updated if the project data changes.

7. Set closure rules for the composition that specify the objects of interest for capture.

8. Create an IDC from the composition that includes only the objects of interest, and save it in Teamcenter.

9. At a later time, you can open the IDC and compare the captured data with the current configuration of the product. If necessary, the captured data can be updated with any changes.

**Note**  An IDC is a PLM XML file that you can manage, share, and release in the same way as other Teamcenter objects.

For further information about creating and manipulating IDCs, see the *Multi-Structure Manager Guide*.

### Performing process simulations

The Process Simulate application allows you to perform detailed simulations of process structures that are maintained in Teamcenter. It includes the following modules:

- **Assembly Studies**

  Assembly Studies facilitate part assembly and disassembly planning processes. You can conduct a static analysis and detect design errors early in the design phase of the process. You can create static and dynamic analyses. Assembly Studies also enable you to examine service and maintenance procedures even before building the first physical prototype. It allows you to do the following:

  0 Plan a product assembly sequence

  Assembly Studies enables you to define the optimal sequence of operations for product assembly and disassembly. Dynamic Gantt charts and time diagrams enable you to view assembly possibilities and limitations, and define the best sequence of operations accordingly.
Chapter 3  Basic tasks

- Conduct a static collision analysis
  Assembly Studies enables you to conduct static analysis of an assembly. You can calculate distances between parts and focus on problematic zones within the path. The system highlights collisions, violations and near misses in red or yellow throughout the process. You can create a cross-section in wire frame or solid to study assembly possibilities and limitations more closely.

- Create insertion and extraction paths
  Using Assembly Studies, you can create a path to insert and extract a part. This is done by moving the part into its location, while the system automatically records a path. Process Simulate also creates paths automatically in complex assemblies.

- Perform dynamic analysis
  Assembly Studies allows you to analyze the process while the simulation is running so that you can detect collisions during insertion of a part as it would happen in real life—either during manufacture or service. The Stop on Collision feature automatically stops the simulation on detecting collisions and violations, allowing you to note and fix problems throughout the process.

- Review and Analyze
  Review and Analyze enables you to simulate and visualize processes created by Assembly Studies. It provides views of the product hierarchy, review of detected collisions, and communicates intent through on-screen annotations.

- Weld
  Weld addresses the spot-welding design process while taking into account critical factors such as space constrictions, geometric limitations, and collision. It allows you to do the following:

  - Designing a spot welding layout
    Weld enables you to design robotic spot welding and station layouts, using existing parts from CAD systems and weld point data imported from the process database. You can use 3D models of robots and standard peripheral components from the user-defined library, or create new models using system modeling tools.

  - Selecting the best gun
    Weld automatically creates sections by slicing weld point fixtures and workpieces. You can use these cross-sections to investigate specific zones of interest. Weld enables you to select the optimal gun for a particular spot welding job by providing weld-point information and required welding parameters. Alternatively, you can design your own gun or modify an existing one using 3D solid modeling tools provided and the dimensions of cross-sections of the workpiece.

  - Optimizing robot placement and layout
    Weld placement tools verify that the robot reaches each weld point without colliding and interfering with other equipment, fixtures and workpieces.
For more information, see the Process Simulate help provided with the application.

**Design a fixture for manufacturing**

When you design a fixture to use in a manufacturing operation, you must know the state of the product in the manufacturing process in which the fixture is consumed. You also should know the factory components in which the operation is performed, and the relative positions of the in-process workpiece and the factory data.

To design a fixture to hold the product in the factory location, the following general steps are necessary:

1. Identify the product components and datums for the fixture design. Put the occurrences of these objects in the structure into a persistent occurrence group.

2. Locate the in-process model of the product. Extract the components of interest for the fixture design from the in-process model into the persistent occurrence group.

3. Identify the factory area in which the operation that consumes the fixture is performed. Identify the relevant elements and interfaces, and put them into a persistent occurrence group.

4. Put the data obtained in the previous steps into a study for the tool designer. You can roughly set the position of the product in the factory.

5. Transfer the study into the CAD system for design.

6. Once the tool is available, assign the operations using the fixture to the newly designed tool.

7. Continue detailed planning with the newly created fixture.

**Creating reports**

You can create reports in the rich client or the thin client, and attach them to appropriate nodes in your process structures. The report formats differ between the two clients, so you cannot open a rich client report in the thin client, or vice versa.

**Creating reports in the rich client**

Manufacturing Process Management provides a framework on which you can create reports that are relevant to your manufacturing processes and the Report Generator application that produces the predefined reports.

For example, you may want to create a report that details the usage of various machined components in assembly operations. From the definition of the product structure, Teamcenter extracts the following information:

- The name or identification of each part consumed in the manufacturing process. Thumbnail pictures can also be included if required.

- Detailed information about each part, such as the dimensions, quantity, and **Make/Buy** status.
Chapter 3  Basic tasks

- A graphical representation of the manufacturing structures and assemblies. Each part is linked to others to show the hierarchy in the assembly structure.

A summary page may define the total number of components, welds, and the comparative number of bought and manufactured parts.

View or print rich client reports with Web browser

You can view or print these reports with a Web browser, as follows:

1. Select the process or operation on which you want a report and open the Report view.

2. Click the button next to the Type box and select a report from the list of available types.

3. Click the Generate Reports button to create the report. Optionally, you can click the Save button to save the report you generated.

4. Open the Attachments view. The generated report is shown as an attachment to the process or operation item revision. You can double-click the report to open and view it.

For more information about generating reports, see the Report Generator Guide.

Configure reports in the rich client

If you use the Report Generator application to create reports and view JT files, Teamcenter runs a Java applet that allows you to manipulate the view (rotate, pan, and zoom), gather measurement data, and mark up the image. For correct operation of this functionality, you must configure your Web browser and related settings, as follows:

**Note** The following instructions assume you are configuring a Microsoft Windows system. Similar configuration steps are necessary if you are using another operating system.

Some of the steps may vary depending on which operating system, browser, or JRE version you use.

- If you are using the Microsoft Internet Explorer browser, modify the Internet properties, as follows:
  1. In Windows, choose Start→Control_Panel→Internet Options
     Windows displays the Internet Properties dialog box.
  2. Click the Advanced tab.
  3. Select the Allow active content to run in files on My Computer check box in the Security group of entries and the Java console enabled check box in the Microsoft VM group of entries.
  4. Click OK or Apply, and then close and reopen the browser to complete any changes.
Basic tasks

If you are using the Firefox browser, this procedure is not required. However, if you encounter problems viewing reports with Firefox, choose Tools→Options and verify the Enable Java check box on the Content pane is selected.

- Install the embedded viewer and run its setup program, if you did not already do so when you ran Teamcenter Environment Manager (TEM). During installation, avoid spaces in installation directory names, as these may cause incorrect operation in certain circumstances.

- Ensure that the JRE version you have installed is at least version 1.6.0_20 or later.

- Click the Advanced tab of the Java control panel and ensure the Place Java icon in system tray check box in the Miscellaneous group is selected. This ensures the Java console is always present in the Windows system tray, allowing easy access to it for troubleshooting.

  When troubleshooting, you can also select the Enable tracing and Enable logging check boxes in the Debugging group. This displays more detailed information in the Java console window that may help you debug problems.

  Optionally, you can click the Show Console button in the Java Console group to open the Java console for debugging purposes whenever the applet is active.

- Click the Browser tab and ensure that the check box for your browser is selected. If you have more than one browser on your system, check each applicable check box.

  If you changed the Java run-time parameters in the previous step, clear your current browser, click Apply, then recheck the current browser and click Apply again. This loads the new environment variable setting into the browser.

- Modify the Java security settings to allow the browser to open the DLL files for the embedded viewer and JT files in one of the following ways:

  - Edit the java.policy file in your JRE_HOME\lib\security directory with a text editor, not a word processor. Add the following entry to this file:

    ```
    grant {
      permission java.security.AllPermission;
    };
    ```

  - Use the policytool.exe application in your JRE_HOME\bin directory to grant the same permissions. See the Sun documentation for information on using this tool.

If you have more than one version of the JRE on your system, or if you also have the Java Software Development Kit (SDK) installed, there may be more than one copy of the java.policy file. If possible, identify the version that is used by the browser. Identify the policy deployment directory by opening the Java console, typing s to dump all system properties, and noting the displayed value of the application.home property. Place your modified java.policy file in the directory indicated by this property; if the directory does not exist, create it. Alternatively, if you are using Internet Explorer, you can modify the file pointed to by the deployment.system.security.policy value.
Note  Close all browsers before testing changes.

**Troubleshooting reports in the rich client**

If the embedded viewer does not work correctly, an **Applet CMEReportJTVApplet not initiated** error message may be displayed in the browser status line. If so, open the Java console in the system tray and check for Java exception codes that may indicate the cause of the problem, as follows:

- **NoClassDefFoundError**

  Java cannot access the embedded viewer JAR files. This error is identified by an error message similar to the following:

  ```java
  java.lang.NoClassDefFoundError:
  com/ugs/plmvis/components/<AnyClassName>
  ```

  Possible solutions:

  1. Check the Java console and verify the correct classpath is set, as described previously. Ensure the **SingleEmbeddedViewer.jar** file is included in the classpath.

     To view the classpath, open the Java console and type `s` to dump all system properties. Note the displayed value of the `javaplug in.vm.options` property and check that it contains the classpath string you entered when you configured the Java console, as described in **Configure reports in the rich client**.

  2. If the classpath is not correct, there may be more than one Java control panel; you may not have modified the correct one. Choose **Start→Settings→Control Panel** and select the button that matches the version number in the Java console window. Modify the classpath for this Java control panel.

  3. Verify there are no typographical errors or unnecessary spaces in the classpath definition. For example, ensure there is an `=` character between `-classpath` and the path name. Also, if you have multiple classpath entries, ensure they are separated by `;` characters.

  4. If none of the solutions work, copy the **SingleEmdeddedViewer.jar** file manually from `PV_Base_Dir\Program` to `JRE_HOME\lib\ext` folder. Close all Internet Explorer browser instances, restart the browser and try again.

- **NoClassDefFoundError: com/eai/visweb/components/…**

  You are trying to run reports from a previous version of Teamcenter. You should convert your reports to use the current embedded viewer.

- **Could not load library jawt.dll**

  Java does not have permission to execute the `jawt.dll` file. Verify that you have updated the Java security policy file correctly, as described in **Configure reports in the rich client**.

- **VerifyError**
The JRE version is incompatible with the HTML page. This error is identified by an error message similar to the following:

```java
java.lang.VerifyError: (class:CMEReportJTViewerApplet,
method: destroy signature: ()V)
Incompatible object argument for function call
at java.lang.Class.getDeclaredConstructors0(Native Method)
```

Possible solutions:

- Ensure you have installed the supported JRE version, as specified in the installation guide for your operating system, and update it if necessary.
- Verify you have installed the correct version of the CMEReportJTViewer.jar file and update it if necessary.

- **Viewer window has no image**
  The viewer window shows a green background and the icons are visible, but there is no image in the window.
  To resolve this problem, try right-clicking in the JT window and choosing **All On** or **Fit All** from the menu.

- **EXCEPTION_ACCESS_VIOLATION**
  The version of the embedded viewer you have installed is not compatible with the CMEReportJTViewer.jar file in the reports directory. This error is identified by an error message similar to the following:

  ```
  An unexpected exception has been detected
  Unexpected Signal : EXCEPTION_ACCESS_VIOLATION
  (0xc0000005) occurred at PC=0x6FD24D6
  Function=Java_com_teamcenter_visweb_components_PMI3D_nativeHideEntity+0x1D9
  Library=C:\WINDOWS\system32\VisWeb.dll
  Current Java thread:
  at com.teamcenter.visweb.components.PrimaryViewer.nativeDestroyViewerDone
  (Native Method)
  at com.teamcenter.visweb.components.Viewer3D.shutdown(Viewer3D.java:718)
  - locked <0x108c3938> (a java.lang.Object)
  at com.teamcenter.visweb.components.Module.removeNotify(Module.java:285)
  ```

  Possible solutions:

  - Ensure you have the correct version of the embedded viewer. You must also update the **PATH** environment variable and modify the classpath for the new version.
  - Ensure you have the latest version of the CMEReportJTViewer.jar file in the reports directory.

- **java.security.AccessControlException**
  If you see a blank HTML page with the following error in the Java console window, the Java policy file is not edited correctly. Check the file and edit as described in *Configure reports in the rich client*.

  ```java
  java.security.AccessControlException: access denied
  (java.lang.RuntimePermission getenv.JRE_HOME)
  at java.security.AccessControlContext.checkPermission(Unknown Source)
  ```
Chapter 3  Basic tasks

at java.security.AccessController.checkPermission(Unknown Source)
at java.lang.SecurityManager.checkPermission(Unknown Source)
at java.lang.System.getenv(Unknown Source)
at com.ugs.plmvis.components.VisView.loadLibraries(VisView.java:2665)
at com.ugs.plmvis.components.VisView.init(VisView.java:2580)
at com.ugs.plmvis.components.VisView.<init>(VisView.java:241)
at com.ugs.plmvis.components.VisView.init(VisView.java:2580)
at com.ugs.plmvis.components.VisView.loadLibraries(VisView.java:2665)

- **To help protect your security, IE has restricted...** message
  If you see a **To help protect your security, Internet Explorer has restricted this file from showing active content that could access your computer. Click here for options...** message in a yellow box instead of the viewer, you have not allowed Internet Explorer to run active content. Repeat the browser configuration procedure described in *Configure reports in the rich client*.

- **Java Applet Support Required...** message
  If you see a **Java Applet Support Required (If you are seeing this message, you need to turn on applet support in your browser)** message in a box instead of the viewer, you have not configured applet tag support. Repeat the browser configuration procedure described in *Configure reports in the rich client*.

- Nothing is displayed in Firefox
  If only a single box is displayed in Firefox, you have not enabled Java. Choose **Tools→Options** in Firefox, click the **Content** button, and ensure the Enable **Java** check box is selected.

**Creating reports for the thin client**

The manufacturing report generation functionality in the thin client requires the Report Generator and PLM XML features of Teamcenter. Ensure that you have the appropriate Java Runtime Environment (JRE) and Lifecycle Visualization software installed on your system. Lifecycle Visualization allows you to create and view 2D and 3D images in reports.

**Configure thin client report generation**

Before you can generate reports in the thin client, the site administrator must configure it as follows:

1. Ensure the embedded viewer software is installed and configured for visualizing 2D and 3D images.

2. Set the required preferences for generating thin client reports, as described in *Set preferences for thin client reports*.

**Set preferences for thin client reports**

- **TC_suppress_report_designs**
  Hides one or more manufacturing reports in the navigator. Valid values are **Process-Structures**, **Product-Structures**, and **Plant-Structures**.

- **TC_TransferMode_XMLReport**
Contains the name of the transfer mode used when generating XML files containing reports. Set this preference to `web_reports`.

- **TC_Report_Transfer_Area**
  Contains the location of the temporary working directory where reports are generated (`%TC_ROOT%\web\htdocs\web_reports`).

- **Batch_Report_Request_File**
  Contains the location of the flat file that defines how batches of reports are generated (`%TC_ROOT%\web\htdocs\web_reports\data\batch_request.txt`).

- **Web_Report_Jars**
  Contains the location of the report JAR files for the JT viewer (`%TC_ROOT%\web\htdocs\web_reports\jar`).

- **WEB_CCObject_shown_relations=**
  Contains the relation types to show with the `CCObject` object. Set this preference to `TC_CCContext` and `TC_reference`.

- **WEB_MECollaborationContext_shown_relations=**
  Contains the relation types to show with the `MECollaborationContext` object. Set this preference to `TC_CCContext` and `TC_reference`.

- **WEB_StructureContext_shown_relations=**
  **WEB_MEProductContext_shown_relations=**
  **WEB_MEProcessContext_shown_relations=**
  **WEB_MEPlantContext_shown_relations=**
  **WEB_MEResourceContext_shown_relations=**
  **WEB_PrivateContext_shown_relations=**
  Contains the relation types to show for all types that fall under the `StructureContext`, `MEProductContext`, `MEProcessContext`, `MEPlantContext`, `MEResourceContext` and `PrivateContext` objects. Each should be set to `TC_reference`.

- **WEB_ZoneProcess_Reports**
  Defines the report formatters that are listed when the user selects a zone process report from the `Generate ME Report` menu.

- **WEB_StationProcess_Reports**
  Defines the report formatters that are listed when the user selects a station process report from the `Generate ME Report` menu.

- **WEB_MEWeldOp_Reports**
  Defines the report formatters that are listed when the user selects a weld operation report from the `Generate ME Report` menu.
Chapter 3  Basic tasks

- **WEB_MEPrPlantProcess_Reports**
  Defines the report formatters that are listed when the user selects a plant process report from the Generate ME Report menu.

- **WEB_MELocateCompOp_Reports**
  Defines the report formatters that are listed when the user selects a CD lead sheet report from the Generate ME Report menu.

- **Class_Resource_WeldGun**
  Defines the class names of weld guns used in report formatters.

- **Class_Resource_Robot**
  Defines the class names of robots used in report formatters.

- **Show_Msv_Occurrence Types**
  Set to True to apply filters on occurrence types for data included in the reports.

  **Note**  If you are using incremental change to manage data included in the reports, you should also set the Incremental Change Management preference to true.

  If you encounter problems with the operation of this feature, try setting absolute paths in these preferences, rather than relative paths.

For more information about setting preferences, see the Preferences and Environment Variables Reference.

**Update report designs**

If you change report designs, you must update the Manufacturing Process Management report design formatters in the database, as follows:

1. Ensure that all users are logged out of the system.

2. Open a Teamcenter shell and enter the following command to install the default report designs:

   ```
   install_default_report_designs -u=infodba -p=infodba -g=dba -file=%TC\DATA\report_writer\default_report_designs.xml -update_formatter
   ```

   Ensure you include the complete path in the file argument, for example:

   ```
   file=C:\Siemens\Tc9.0\default_report_designs.xml
   ```

**Run batch reports**

To create batches of reports, you create a batch request file and define its location in the Batch Report Request File preference. Ensure the file identified by this preference has global read/write permissions.

To create a batch of reports manually, run the rep_batch_report utility by entering the following command:

```
rep_batch_report -u=username -p=password -g=group
```
You can also create a scheduled task that creates batches of reports automatically. Use the sample script located in the TC_ROOT/web/htdocs/web_reports/data folder as the basis of your script and modify it to set the environment.

## Displaying GD&T symbols and rich text in forms

You can configure any form to contain geometric dimensioning and tolerance symbols or rich text in forms. It is possible to include geometric dimensioning and tolerance symbols in any box of a form but, in practice, you would restrict them to fields intended for engineering data. For example, a box titled **Work Instruction** is a good candidate for geometric dimensioning and tolerance symbols, while a box titled **Quantity** is not. To define fields that may contain geometric dimensioning and tolerance symbols, you add their names to the **GDT_properties** preference. Once a box is added to this preference, a user can add geometric dimensioning and tolerance symbols to the box in any instance of the form.

You can activate geometric dimensioning and tolerance symbols for any type of form, including master forms for a Teamcenter object. You can apply normal Access Manager rules to any form that contains geometric dimensioning and tolerance symbols.

Users may only edit or add geometric dimensioning and tolerance symbols or rich text in a box after clicking the edit button next to the relevant box. They may then enter symbols, control characters or font formatting from the buttons provided or the keyboard. When editing or additions are complete, the user saves changes and leaves edit mode. This feature is available in any application that allows viewing of forms, including Manufacturing Process Planner, Part Planner, and Multi-Structure Manager. Detailed usage information is given in the guides for the applications that support this feature.

You can print and view reports that are generated from forms that contain geometric dimensioning and tolerance symbols or rich text. You can generate forms in two formats:

- **HTML**
  Teamcenter generates a PLM XML file from the process structure, which is parsed, interpreted with XSL style sheets, and converted to HTML files.

- **PDF**
  Teamcenter generates a PLM XML file from the process structure, which is parsed, interpreted with XSL style sheets, and converted to Formatting Object (FO) files. The information must then be converted to PDF format by a third-party formatting tool.

This feature is only supported in the rich client on Microsoft Windows workstations; geometric dimensioning and tolerance symbols and rich text may not display correctly in the thin client.

## Configure GD&T symbols and rich text in forms

Activate GD&T symbols and rich text in forms by implementing XML style sheets.
Chapter 3  Basic tasks

**Note**  This feature is only supported on Microsoft Windows systems.

Ensure you have a supported browser installed on your workstation if you want to view or print HTML reports of information containing GD&T symbols.

1. Prepare the names of GD&T and rich text properties as follows:
   a. In My Teamcenter, choose **Edit→Option→Search** and find the **GDT_properties** preference. If the preference does not exist, create it at the **SITE** level or above.
   b. For this preference, enter the names of the GD&T and rich text form fields you want Teamcenter to recognize in the **Values** box, for example, **work_instruction**.
   c. Similarly, create a new preference at the **SITE** level or above called **GDT_formtypes**, and enter a list of form type names as the value of this preference. Teamcenter recognizes these form types and, if a form of one of these types has a GDT box (that is, the name of the box is defined in the **GDT_properties** preference), it renders this field as a GD&T/rich text box.

   You have now defined names of GD&T/rich text attributes (**fields**) and form types. By default, Teamcenter renders GD&T and rich text fields as specified by the **GDT_properties** and **GDT_formtypes** preference entries. You can also use the XML style sheet mechanism to override these settings, as described in step 4.

2. Create the form types that can display geometric dimensioning and tolerance symbols or rich text, as follows:
   a. Start the Business Modeler IDE and create new classes that have attributes listed in the previous step under **POM_object**. For example, you might create a new class called **DemoGdtForm**. Define the attributes for GD&T and rich text as **String** type with the maximum length possible.
   b. In the Business Modeler IDE, create new form types that use the class you created as the underlying POM class. Similarly, you can also create item master forms or item revision master forms to support GD&T and rich text in those master forms.

3. Copy the **gdtex.ttf** font file from the rich client installation directory to the **FONTS** directory of your Microsoft Windows system.

   **Note**  You must install this font file on every workstation where HTML are generated, viewed or printed.

4. Define an XML style sheet for the form type that contains GD&T or rich text fields. Enter **gdtpanel** as the rendering hint of the intended GD&T and rich text fields. If you do not give a rendering hint, Teamcenter uses the default renderer (it checks if a box contains GD&T or rich text by looking at the preference settings, as described in the previous section).

   The following XML example shows a style sheet for a form. In this example, **comments** and **tolerance** are rendered GD&T or rich text, while **work_instruction** is rendered as a text area. These rendering hints override any conflicting preference settings.
5. Verify you can now display and edit geometric dimensioning and tolerance symbols or rich text on the appropriate forms and can export the data into reports.

**Note** If Teamcenter indicates it cannot find the `corojdk11.dll` library, locate this file on the Teamcenter installation CD, and copy it into your system path.

If the display of GD&T and rich text becomes unstable when you type quickly, increase the value of the `EDITOR_UPDATE_LATENCY` variable in the `com\teamcenter\rac\form\gdt\gdt.properties` file from the default value of 50 and restart Teamcenter. This instability occurs on slower machines.

**Generating PDF reports**

To generate reports from PLM XML data, you must write XSL style sheets that define how GD&T and rich text should be translated into FO format. Examples of the necessary style sheets are provided in the *Client Customization Programmer’s Guide*.

You should also install a suitable third-party formatter to convert FO files into PDF files. Siemens PLM Software does not provide or endorse a particular formatter, but possible options if you want to use standard FO format include **FOP**, **XEP**, and **XSL Formatter**. If you want to use standard FO format mixed with MathML syntax, consider **XSL Formatter** with the **MathML** option. You may want to test each of these approaches with your report data to determine which formatter gives the best results.
Chapter

4 Planning assembly manufacture

Setting manufacturing preferences ............................................ 4-1

Associating the EBOM and MBOM ............................................. 4-2
  Reconciling EBOM and MBOM in a Multi-Site environment .......... 4-3

Designing processes for precise or imprecise structures ................. 4-3

Working with relative occurrences ............................................. 4-4
  Examples of occurrences in manufacturing ............................... 4-4

Managing studies ................................................................. 4-5

Body-in-white integrated workflow ............................................. 4-6

Work with flexible components and NX ....................................... 4-7

Integrating with a manufacturing execution system ....................... 4-8
# Planning assembly manufacture

## Setting manufacturing preferences

The following preferences are useful in specifying manufacturing behavior. Setting these preferences can affect all manufacturing applications.

The following preferences determine how occurrence types are interpreted and what behavior is associated with the occurrence types. Suitable default values are provided for these preferences.

<table>
<thead>
<tr>
<th>Preference</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>MERelationTypeUsedEquipment</td>
<td>Defines occurrences types that are interpreted as tool or equipment assignments.</td>
</tr>
<tr>
<td>MERelationTypePartsAssigned</td>
<td>Defines occurrences types that are interpreted as product assignments.</td>
</tr>
<tr>
<td>MERelationTypePartsConsumed</td>
<td>Defines occurrences types that are interpreted as product consumptions.</td>
</tr>
<tr>
<td>MERelationTypePartsHandled</td>
<td>Defines occurrences types that are interpreted as product handling.</td>
</tr>
<tr>
<td>MERelationTypePartsAssembled</td>
<td>Defines occurrences types that are interpreted as product assembly.</td>
</tr>
<tr>
<td>MERelationTypePartsDisassembled</td>
<td>Defines occurrences types that are interpreted as product disassembly.</td>
</tr>
<tr>
<td>MERelationTypePartsOtherAssigned</td>
<td>Defines occurrences types that are interpreted as other assignment types (those not matching the product assignment types set in the other preferences).</td>
</tr>
<tr>
<td>MERelationTypeFeature</td>
<td>Defines the relation that indicates the consumed item is a manufacturing feature.</td>
</tr>
</tbody>
</table>

The following preferences configure the loading of structure contexts in Manufacturing Process Planner.

<table>
<thead>
<tr>
<th>Preference</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default_StructureContext_Type</td>
<td>Determines which type of structure context is created when saving a loaded object as a context.</td>
</tr>
</tbody>
</table>
### Associations

### Preference

<table>
<thead>
<tr>
<th>CCCompositionTypesPref</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Determines which composition types are displayed to users in Manufacturing Process Planner. A composition is a structure that comprises components from different top lines, in the context of a configured top-level assembly.</td>
<td></td>
</tr>
</tbody>
</table>

### Associating the EBOM and MBOM

The manufacturing engineer creates a manufacturing structure (MBOM or manufacturing bill of materials) from the product structure (EBOM or engineering bill of materials). The MBOM is a different arrangement of the EBOM that can be optimized for manufacturing purposes. Each line in the MBOM must be logically aligned with a line in the EBOM to ensure that, if the EBOM changes, the MBOM can be updated appropriately. Linked occurrences represent individual instances of the same physical part or assembly in the different domains. You can run an alignment check after any changes to identify discrepancies and resynchronize aligned occurrences.

Operation of this feature is controlled by the `MEAlignedPropertiesList` preference. This preference defines the list of BOM line properties considered during alignment, comparison, and synchronization of aligned occurrences. By default, the `AbsOccID`, `Child Item`, `Notes`, `Quantity`, `Unit of Measure`, `Transform`, `Substitute List`, `Source`, and `Occurrence Type` properties are considered. Do not add properties to this preference unless you ensure methods exist to read from and write to them.

When you link an EBOM with an MBOM, their BOM view revisions are linked by an `METTarget` relation.

- **Caution**
  - This feature uses the in-context ID (IDIC represented by the `AbsOccID` property) to establish logical equivalence between aligned occurrences. If you want to use this feature, you cannot use the in-context ID for any other purpose.

  If you use Multi-Site Collaboration and want to consume a remote EBOM in the MBOM, you must create the in-context IDs for the BOM lines to be consumed at the owning site before you export the EBOM to a remote site. Otherwise, an alignment check does not report differences between the structures.

  For more information, see *Reconciling EBOM and MBOM in a Multi-Site environment*.

- To use the alignment check, you must link the EBOM and MBOM structures. You cannot link an EBOM to an MBOM contained in a composition structure context. If you use a structure context to hold the MBOM, you cannot use the alignment check feature.

- Cutting and pasting the same line back into an MBOM creates a new occurrence without an in-context ID. This removes the equivalence between the new occurrence and the original source part in the EBOM.
For more information, see the *Manufacturing Process Planner Guide*.

**Reconciling EBOM and MBOM in a Multi-Site environment**

The determination of equivalence between the engineering bill of materials (EBOM) and the manufacturing bill of materials (MBOM) is based on the in-context ID (IDIC, also referred to as the absolute occurrence ID). When you paste a part from the EBOM to the MBOM, Teamcenter generates an identical, unique absolute occurrence ID for the part on both the EBOM and the MBOM. This mechanism is used when aligning structures to determine whether parts are missing in the source or target structures.

The ID is generated in the context of the top-level item. Even if you do not have write access to the top level of the EBOM, Teamcenter allows this ID to be generated and saved with the top-level item. If you are using Multi-Site and the EBOM is at the remote site, however, this mechanism does not work as Teamcenter must be able to check out the top level to generate the ID. A similar problem arises if you want to assign from an EBOM that exists on a remote site to a local process.

To enable a remote site to use the EBOM-MBOM mechanism in these situations, you can generate in-context IDs on the EBOM that are available when you paste parts to the MBOM. Use the *bom_expand* utility to generate these IDs on the EBOM. For the EBOM shown in the figure, enter this command:

```
bom_expand -u=user-id -p=password -g=group -item=000035 -create_absocc_ids
```

Generate these IDs on a regular basis, for example, nightly, so that they are up-to-date when users assign parts from the EBOM to MBOM.

**Caution**

Attaching an in-context ID to a line in a structure creates several new objects in the database. If you work with large amounts of data, this may have a detrimental affect on performance or scalability. You should only regenerate these IDs if you use EBOM and MBOM in a Multi-Site environment.

**Warning**

Do not use the *bomwriter* utility to generate absolute occurrence IDs. This can lead to data corruption.

**Designing processes for precise or imprecise structures**

Manufacturing Process Management allows you to design manufacturing process for precise and imprecise assemblies.

An *imprecise* assembly contains components with unspecified revisions, rather than specific revisions of those components. This allows engineers to create product
structures (BOMs) for varying configurations of the product, such as a configuration with revisions that are released for production or a product structure with the latest working revision of components. The underlying product structure remains unchanged, so you can design a manufacturing process for an imprecise assembly if the process does not change when the revision of a component changes.

Conversely, a precise assembly contains specified revisions of components and may need to be updated each time a component is revised. The manufacturing process is linked to the specific configuration and may need to be updated more frequently than when working with imprecise assemblies.

**Working with relative occurrences**

An occurrence (relative occurrence) is a persistent object that represents the usage of an item in the context of a top-level structure. It represents the shape, position, or other attribute of the item at a specific point in the manufacturing process. For example, the occurrence of the assembly process for the left front wheel of a vehicle differs from the occurrence of the right front wheel, even though the actual wheels are identical.

You can combine multiple occurrences into occurrence groups in Teamcenter to model assembly and manufacturing requirements. Occurrence groups may contain occurrences of parts and subassemblies that are not necessarily grouped together in the product design. Alternatively, an occurrence group may collect all occurrences in the product that are within a given proximity of a part that you are designing.

An absolute occurrence is a specific instance of an item, in the context of its immediate parent. It stores occurrence data such as usage specific names and descriptions or positional overrides. Absolute occurrence data overrides relative occurrence data. For example, the vehicle may have two occurrences of a left wheel, in the context of the suspension assembly. Users can name one of those occurrences as left rear wheel, in the context of the complete vehicle. Assigning a name to an absolute occurrence of an item does not override or change the attributes of the general item.

**Examples of occurrences in manufacturing**

The following list shows some examples of how occurrences may be applied in the manufacturing environment:

- Variation from the design scenario

  Flexible components may change their shape in the context of a top-level product. For example, a shock absorber in a car may not be compressed when in inventory. However, when the part is assembled into a car, it absorbs the weight of the car and becomes compressed. As a result, the geometric definition of the part changes in the context of the car. To define this condition for the manufacturing process, you must associate a separate NX part and JT image with the occurrence of the compressed part in the car.

- Factory configuration

  A factory structure contains many standard components such as robots, conveyors, and welding guns. Each such standard component may have a structure of its own. When positioned in the factory, the location of a component may change based on its relationship to other factory elements. For example, you
might need to change the installed position of a robot in one area of the factory compared to another. To allow for these differences, you can create occurrences of each location of the robot relative to the standard component.

- Installation cost

The cost of installing a robot may differ from one area of the factory to another. You can associated different cost data with each occurrence of the robot.

- Manufacturing setup

A setup is a *composition* that represents the environment in which an operation or study is performed. The setup contains occurrences groups of various structures in the product and the factory. In the manufacturing setup, one of the occurrence groups represents the in-process assembly as it arrives at the station. If, for example, this in-process assembly includes the doors of a car, the doors may be in the closed position in the design product structure. However, to assemble the seats into the car, the doors must be open. To allow for this requirement, you should change the positions of the doors in the context of the in-process model during manufacturing. The position must be associated with the door only in the context of the product view (occurrence group) and should not affect the product definition.

- Human studies

A study or setup may contain a model of an assembly worker and instances of parts from the car model that interact with the worker, including the steering wheel and pedals. The positions of these parts can be changed only in the context of the study without affecting the original position in the product definition. For example, the steering wheel may be rotated along the steering bar or a pedal pressed to the down position. In this case, the worker is positioned relative to the car in the context of the setup, and you can change the positions of parts of the car to model different assembly scenarios.

- Integration with Tecnomatix process planning software

You may use Teamcenter to manage and maintain process data, but author your actual processes in the Tecnomatix process planning software. Information about the process structure is required during the planning phase of the process plan. During this planning phase, factory components may change their positions and associated data. Tecnomatix uses a data model in which each component in the factory is unique. When you import data from Tecnomatix into Teamcenter, it is associated with the corresponding occurrences in Teamcenter. A robot may change its position as the result of a change to the design of a station layout. However, this change should not affect other instances (occurrences) of the robot.

**Managing studies**

Studies provide an environment where you can scope data to perform trial activities. There are two types of studies:

- **Shared studies**
  Teamcenter copies the objects from the source to the study. Any changes you make in a source structure are automatically reflected in the shared study.
Simulation studies  Teamcenter creates cloned processes and operations, creates references for any products or resource, and places them in the simulated study. A simulation study is a partially isolated structure where you can make changes to the process-related data that are not reflected in the source structures. When you are ready, you can choose to synchronize your changes from the source structure to the study or publish them from the study back to the source structure.

You can send studies to Process Simulate.

For more information, see the *Manufacturing Process Planner Guide.*

**Body-in-white integrated workflow**

1. Design engineers design the product in NX. This is done in the Teamcenter Integration for NX environment, meaning they pull files from the Teamcenter database and save them back interactively. Only design engineers can modify the assembly that includes part and JT files.

2. Body-in-white (BIW) manufacturing engineers run Teamcenter rich client. They open the engineering BOM (EBOM) using a revision rule that lets them see the approved (released) content. They create a body BOM skeleton with precise revisions. They assign only the sheet metal parts from the design BOM. Any multiple lines for the same part in the EBOM (for example, for different variants) are distilled to a single line in the body BOM. The body BOM is modifiable only by the body-in-white engineers.

3. BIW engineers in NX add welds and datum points to the body BOM assembly. Because all the revisions are precise, the body BOM has exact revisions of everything from the EBOM.

4. BIW engineers import a plant structure with JTs to Teamcenter.

5. BIW engineers create a process structure. Generally each process represents a workstation in a factory. The process references the in-process assembly (the state of the product to date), the parts to be welded, and the plant structure’s workstation. All of it can be seen in the assembly viewer. BIW engineers add child operations representing weld guns. Weld guns and the welds to be created by the gun are attached to the operation as resources and as consumed lines, respectively.

6. BIW engineers create a **Tool Design Package** study and copy the process containing the workstation, IPA, consumed objects, guns, and welds into it that they want to send to Process Simulate.

   For more information, see the *Manufacturing Process Planner Guide.*

7. In Process Simulate, BIW engineers analyze the welds and modify the approach angles on them. The results are exported back to Teamcenter.

8. The user suppresses some of the lines under the process in the study that do not need to be sent to the tool supplier, including some of the product and workstation contents. In the assembly viewer, the user sees the location of the parts,
workstation, and weld guns in the process structure. Teamcenter does not display the weld angles graphically.

9. BIW engineers use PLM XML export to export the tool design study to a directory on a local disk. The engineers specify the ToolDesignPackageExport transfer mode specifically designed to export the desired content, including the in context forms that are specially processed to modify the BOM line transforms of the welds. This content includes the XML file along with part files in subdirectories.

10. BIW engineers zip the export directory that contains the XML and associated files and sends the compressed file to the supplier.

11. The supplier unzips the NX assembly and uses the nx_import_tool_design_package utility to load the data into an NX installation running in native mode. The supplier sees an assembly structure that resembles the original structure but is not the same. Manufacturing features are opened as parts and additional coordinate system objects are added to represent the projected position (the orientation) of the locations created in Process Simulate. The supplier designs the tool in the product assembly’s coordinate system producing part files on a local disk.

12. The supplier moves the weld gun to each weld and aligns it to the weld angle to visually ensure that the gun does not collide with the tool in each weld position.

13. The supplier zips up the part files and sends them back to the BIW engineer.

14. BIW engineers, running Teamcenter Integration for NX, import the tool assembly sent by the supplier to the Teamcenter database.

15. In the Teamcenter rich client, BIW engineers load the process structure and attaches the tool assembly to the process. By using the same transform as the IPA, it is located correctly in the assembly viewer relative to the rest of the objects in the study.

16. The process is sent to Process Simulate for validation of the tool design against the robot tool paths.

**Work with flexible components and NX**

The following scenario is an example of how you can use occurrences to manage flexible components created in NX:

1. Create a flexible component (F) in NX.

2. Add F to a subassembly (B), together with other components.

3. Use subassembly B in another subassembly (A).

4. Create a JT file (F1.jt) to represent the flexible component in context A. To define a JT file, choose File→Options→Save Options and select Save JT Data; NX creates the JT file next time you save the design.
5. Use A in a top-level assembly (T), which requires a change in the definition of the flexible component F. To meet this requirement, you create another JT file (F2.jt) and store it in the context of T.

6. Attach the F1.jt file to an occurrence of F with respect to A. Also, attach F2.jt to the occurrence of F with respect to T.

7. Import and save these definitions in Teamcenter.

If another user now opens the top-level assembly T in Teamcenter and views F in the product viewer, the correct representation in the F2.jt file is shown. However, if the user opens A in the assembly viewer, F1.jt is shown.

**Integrating with a manufacturing execution system**

A manufacturing execution system (MES) is designed to help companies more effectively and efficiently execute manufacturing operations from the product order through each step of the manufacturing process to its final point of delivery. Equally important, an MES is a dynamic information system that is key to collaborative manufacturing strategies by providing mission-critical information about production activities to managers across an organization and its supply chain.

The integration between the definition process in product lifecycle management (PLM) and the control and execution process in an MES generates tremendous value for both processes. As part of the PLM process, the bills of process (BOPs) can be fully simulated in digital replications of the factory. These BOPs can be created, tuned, potentially optimized, validated, and then sent as a technical work package to an MES that automatically generates a work plan for the shopfloor. The MES also controls and monitors the execution of this work plan on the factory floor. Therefore, the interoperability between the MES and PLM is critical in closing the gap between simulation in the engineering environment and reality on the factory floor. As production ramps up, information collected by the MES is essential in adjusting the bills of process. This information is fed back to engineering to improve their knowledge of the actual BOPs needed to produce desired results. Continuous improvement initiatives can benefit from the collection of data when the MES is monitoring and reporting what actually occurred on the physical factory floor.

The Manufacturing Execution System Integration collects the bill of process, the bill of materials, and any relevant work instructions into a work package that is released to the MES. The operator can use this information to machine the desired product. If there is a problem with the data, the planner can modify the contents of the work package and release it again.

For more information, see the *Manufacturing Execution System Integration Guide*. 
Chapter

5 Planning part manufacture

Planning part manufacture ........................................... 5-1

Transfer data with the NX CAM Integration ........................ 5-1
  Updating CAM templates in the database ......................... 5-2
  Load CAM data into Teamcenter .................................. 5-3
  Managing multiple versions of NX templates .................... 5-3
    Importing the template files .................................. 5-4
    Managing the PAX files ........................................ 5-4
  Create operation types for the NX CAM Integration ............ 5-4
  Modify the CAM integration to display additional data types 5-5

Storing postprocessor and simulation drivers in Teamcenter .... 5-6
  Using the import_nxcam_post_files utility ...................... 5-8

Copying MENCMachining operations .................................. 5-10
Chapter

5 Planning part manufacture

Planning part manufacture

Teamcenter provides an interface to the NX Manufacturing application, allowing you to:

- Capture and manage NC data in the Teamcenter environment.
- Capture output from the NX Manufacturing application, including CLSF, PTP, shop documents, and IPW into Teamcenter.
- Access resources stored in Resource Manager including templates, machine tools, tooling and fixtures from NX CAM Integration.
- Access the process plan created by the process planner in Teamcenter.

Transfer data with the NX CAM Integration

The following sequence shows how Teamcenter transfers data between the process planner and the NC programmer. It assumes that the process planner has created the overall process plan in Teamcenter.

1. The process planner creates an NC operation in Teamcenter as part of the overall process plan. Typical operations may include milling, grinding, drilling, machining and inspection. Data associated with an operation may include the duration, machine parameters, and target specifications.

2. Optionally, the process planner specifies the tooling and machine tools in the planning environment. The definitions are based on planning data such as data from an MES system.

3. The process planner may assign resources that are required by NC setups. The NC programmer can view these assignments, as Teamcenter updates the CAM environment.

4. The NC programmer can add, modify, or remove tools specified for the operation if additional criteria are identified.

5. The NC programmer generates the necessary NC program information and this is also available in Teamcenter, including the NC setup, CLSF, PTP, and shop floor documents. In addition, tooling and in-process models are available in Teamcenter.
6. The process planner generates work instructions from the activities, tooling, NC operations and other information available in Teamcenter.

Alternatively, the NC programmer may create NC setups within a process and route them from within the CAM environment. At the same time, the NC programmer may create tools in the NC environment and save them in Teamcenter.

The following figure shows how data that is defined in the NX Manufacturing application can be stored and managed in Teamcenter.

Using NX CAM Integration data in Teamcenter

Updating CAM templates in the database

Use the upgrade_nx_cam_templates utility to help you load new CAM templates when installing a new version of NX. With the help of this utility, you can specify whether the existing templates in the database should be overwritten with new ones or left untouched because they are customized.
Load CAM data into Teamcenter

If you create CAM data in the NX CAM Integration, you can generate a CLSF file or postprocess the data to generate a PTP file. You can also create NX shop floor documentation that contains information about the CAM objects.

You then save these files and documentation in Teamcenter and verify the results, as follows:

1. Open Part Planner and select the operation for which you generated CAM data. Refresh the operation if necessary.

   **Note** You can select any operation type supported by NX and specified in the `NX_supported_operation_types` preference.

2. Expand the selected operation. Teamcenter displays the setup, which includes all items included in the NX CAM setup templates and any additional resources. These items may include the workpiece itself, the machine tool, its clamps and fixtures, and the cutting tools.

3. Click the **Operation Activities** in the right-hand pane, and verify that an **MENCProgram** activity was automatically created under the operation with the program name defined in NX CAM. Check the PTP and CLSF files are added as datasets to the activity; double-click these files to open them in a text editor.

4. Click the **Attachments** tab in the right-hand pane and verify the NX shop floor documentation is attached to the operation revision as a dataset.

Managing multiple versions of NX templates

You can manage multiple versions of NX CAM templates in the Teamcenter database. Depending on the version of NX that you use, NX retrieves the appropriate templates. In addition, when you update to a later version of NX, you can import the new templates into the database without overwriting the existing templates.

The templates are available through the **File→New** dialog box and in the CAM setup. Each NX version is associated with a PAX file and these are specified in Teamcenter preferences. If you set the Teamcenter preference, NX retrieves the PAX files from the Teamcenter database. If the preference is not set, NX uses the PAX files from the native installation.

You must set the Teamcenter preferences manually. They do not exist in the database. Each preference refers to a specific NX installation and lists the item IDs of the items containing the appropriate PAX files.

- `TC_NX_FileNewPAXFiles_NX75`
- `TC_NX_FileNewPAXFiles_NX8`
- ...

**Note** These preferences are supported when running NX 6.3 and later. However, you can only manage CAM templates from NX 7.5 and later.

You can set the Teamcenter preferences differently for user, group, and site. This provides a means to configure what templates users see and helps reduce the number of templates presented to a single user.
Chapter 5  Planning part manufacture

Before you use this feature, you must import the appropriate PAX files for each NX installation. The item ID of the item containing these text datasets is used as the value for the corresponding preference. For example, if you import the NX 7.5 PAX file into an item with an ID of NX75_PAX_FILE/A, you must enter NX75_PAX_FILE/A as the value for the TC_NX_FileNewPAXFiles_NX75 preference.

You can find a CAM PAX file, **nxdm_ugs_manufacturing_template_sample.pax**, in the *UGII_BASE_DIR\UGII\templates\sample\* directory.

**Importing the template files**

You import new template files using the **tcin_cam_template_setup.bat** utility in the NX installation. This script imports the CAM templates found in the following directories to the Teamcenter database.

- UGII_BASE_DIR\MACH\resource\template_part\english\T_DIR
- UGII_BASE_DIR\MACH\resource\template_part\metric\T_DIR

The item ID of an imported template consists of the name of the template suffixed with _inch_NX? or _metric_NX?, where ? is the release version of the NX installation. For example, the NX 7.5 metric template **MillTurn_Express.prt** receives the item ID of **MillTurn_Express_metric_NX75** when you import it into Teamcenter. Teamcenter imports the templates into a folder called NX? CAM Setup Templates, where ? indicates the NX release. For example, the NX 7.5 templates are organized under the **NX75 CAM Setup Templates** folder.

You must set the following NX environment variables:

- UGII_BASE_DIR
- UGII_ROOT_DIR
- UGII_TMP_DIR

You find the **tcin_cam_template_setup.bat** utility in the NX installation in the *UGII_BASE_DIR\UGII\templates\sample\* directory. Run the script using the following parameters:

- **UGII_BASE_DIR\UGII\templates\sample\tcin_cam_template_setup**
- _u=**username**  _p=**password**

**Managing the PAX files**

You can store a PAX file in any item type. You must import the PAX files into the revision of this item. Siemens PLM Software recommends that the dataset only contain one PAX file. If you need to revise this PAX file, use the **Revise** menu command. By putting each PAX file dataset into an individual item revision, you can independently author, revise, and initiate Teamcenter workflow processes for each of them. If a site wants to share only a selection of PAX files to other sites (but not all), it can control the sharing more effectively if the files reside in different item revisions.

For more information about PAX files, see the *Teamcenter Integration for NX Help* in the NX Help Library.

**Create operation types for the NX CAM Integration**

The NX CAM Integration allows you to see certain planning operation types that are created in Teamcenter in NX. By default, only an operation of the **MENCAMachining**
type is visible in NX. You can create a new operation type specific to your needs by creating a new business object in Business Modeler IDE. After assigning this new type to a planning operation, the operation is visible in the Setup Selection dialog box in NX.

1. Add the new name of the new type to the `NX_supported_operation_types` preference.
   
   For more information about setting preferences, see the *Preferences and Environment Variables Reference*.

2. On the Business Objects tab, right-click `POM_application_object→WorkspaceObject→Item→MEOP`.

3. Choose *New Business Object* from the shortcut menu.

4. Type the name of the new type in the *Name* box.

5. Click *Next*.

6. Add the required form attributes and click *Next*.

7. Click *Finish*.

Now, when you create a new operation in Part Planner, you can select your new operation type. When you open the part in the NX CAM Integration, the operation appears in the list of visible operations in NX.

**Modify the CAM integration to display additional data types**

If you want to extend the NX CAM Integration to save files in an additional file format with your part file, you must customize Teamcenter to recognize these new data types. Once you do this, Part Planner can display the files with the new data types in the activity or as an attachment. If, for example, NX stores a file with the `.nc` ending, you can display these files in Part Planner.

1. Ensure that NX CAM saves the new file type in the following directory:
   
   `%TEMP%\item ID_xxx`

2. Open the Business Modeler IDE.
   
   For information about setting up your own business modeler project, see the *Business Modeler IDE Guide*.


4. If you want to:
   
   a. Display the new document type in the activity along with the CLS files, select `UGCAMCLSF`.
   
   b. Display the new document type in the activity along with the postprocessor files, select `UGCAMPTP`.
c. Display the new document type as an attachment along with the shop floor documentation files, select **UGCAMShopDoc**.

A tab opens containing the properties of the selected dataset business object.

5. Click the **References** tab.

6. Click **Add** to add a new line to the list of references.

7. In the **Add Dataset Reference** dialog box, type a reference name in the **Reference** box, for example, **NC**.

8. Type the ending of the file type in the **File** box, for example, **.nc**.

9. Select the file type in the **Format** list.

10. Click **Finish**.

11. Click the **Parameters** tab.

12. Click **Add**.

13. In the **Add/Modify Dataset ToolAction** dialog box, click **Browse** to expand the tools hierarchy and choose the tool you need.

14. Select the type of operation you want to perform in the **Operations** list.

15. In the **References** list, click **Add**.

16. In the **Add Reference** dialog box, select the reference name and the **Export** check box.

17. Click **Finish** twice.

Now, when you save an operation in NX CAM, the NX CAM Integration searches in the **TEMP** directory for those file extensions that you have customized using these steps and lists them in the **Import ME Datasets for Part** dialog box. Teamcenter displays the new data type in addition to all CL source files, postprocessor, and shop floor documentation files with the operation in Part Planner.

For more information about extending dataset types, see the **Business Modeler IDE Guide**.

### Storing postprocessor and simulation drivers in Teamcenter

You can store all postprocessor and simulation driver files in the file system or import them into the Teamcenter database. By default, these files are stored in the **MACH** directory of your NX installation. If you import them into the Teamcenter database, NX retrieves them when postprocessing or using Integrated Simulation and Verification (ISV) in NX. This feature is available with NX 7.5 and later.

Before you use this feature, you must run the **import_nxcam_post_files** utility to import the required files into the database.
After import, a classified machine contains a dataset with a MachConfigFile named
reference that stores all configuration files for that machine under the UGMASTER
dataset. The following shows a sample machine, sim010101_001_mm.

The postprocessor configuration file is listed in the PP config filename
classification attribute.

To change the configuration file of a machine, change the name in the PP config
filename box. If the machine post or simulation driver item has more than one
revision, Teamcenter uses the latest revision.

When NX CAM loads a machine in an NX session, the corresponding post files and
simulation driver data are copied from the Teamcenter database to a temporary
directory for further postprocessing or for use with ISV. If you select a new machine
in the CAM session, the post and simulation driver data are also copied to the
temporary directory. To support a wide range of different file types for post and
simulation data, Teamcenter stores and manages the files in a ZIP file. Post and
simulation files are organized in the ZIP file and saved as a named reference in the
dataset. The same directory structure is maintained after all the necessary data files
are exported to the operating system's file system.

Three types of machine-independent data are stored in the Teamcenter database
to support all machine postprocessing and simulation activities. These are global
files that are shared by all machines.

- The data for postprocessing from the postprocessor directory is stored in an
  item with an ID of MEPPostProcessor.

- The data for user-defined events from the user_def_event directory is stored in
  an item with an ID of MEUserDefinedEvent.

- The data for shop floor documentation from the shop_doc directory is stored in
  an item with an ID of MEShopDocument.
The generic machine in CAM is supported if the template_post.dat file is defined in the zip file attached to the MEPostProcessor item.

**Note** You must set the Use Postprocessors and ISV Drivers from Teamcenter customer default in the NX session by choosing File→Utilities→Customer Defaults before you can use supporting files in Teamcenter.

### Using the import_nxcam_post_files utility

Use the import_nxcam_post_files utility to import all or partial postprocessing and simulation driver data into the Teamcenter database. You must provide the directory that contains all post and driver data. Siemens PLM Software recommends that a system administrator execute the import utility.

During the import steps, all newly created items are placed in a folder named CAM Post Files. If the folder does not exist, Teamcenter creates it. The utility builds the necessary links between machines and simulation data. Encrypted binary post files or posts that are linked posts must reside within the imported directory so that all necessary files are imported into the database for further uses.

The import_nxcam_post_files utility does not import the graphics for a machine. Typically, they already exist in the database if you work with machine simulations. If not, you must import the NX assembly using the ug_import utility.

For more information about the ug_import utility, see the Teamcenter Integration for NX section of the NX Help Library.

You must follow folder and file structure naming conventions for the utility to work correctly.

- The import directory can contain folders named library, postprocessor, user_def_event, and shop_doc. The library folder must contain a machine folder. The machine folder must contain an installed_machines folder. All other names are ignored.

- All machine-specific data must be defined inside the installed_machines folder. The classified machine item ID must have the same name as the directory name under the installed_machines directory so the utility can find the correct machine and perform the import task. For example, when importing the sim010101_001_mm machine, an item with an ID of sim010101_001_mm must already exist in the database.
• Each machine folder must contain at least two folders. One folder name must have the keyword `cse`, for example, `cse_driver` or `fanuc_cse_files`. The folder contains all simulation files for the machine. The second folder name must have the keyword `postprocessor`, for example, `fanuc_postprocessor`. The folder contains all postprocessor files for the machine. At least one configuration file (`.dat`) for the machine must exist in the machine folder. The utility imports all `.dat` files and the contents of the `postprocessor` and `cse` directories only. All other folders and files are ignored.

**Note** When you add new posts, the file structures must be consistent with these structures. For new posts and configuration files, you must use the NX CAM `UGII_CAM_POST_DIR` and `UGII_CAM_LIBRARY_INSTALLED_MACHINES_DIR` environment variables.

**Note** The classified machine must exist in the database before you execute the utility. If a machine cannot be found in the database, the utility fails on that machine only.
For examples of calling the `import_nxcam_post_files` utility, see the Utilities Reference.

**Copying MENCMachining operations**

In Part Planner, CAM PTP files reside in a UGCAMPTP dataset which are attached at a very low level in a process structure, for example:

```
MENCMachining Revision→MeActivity→MENCprogram→UGCAMPTP
```

If you revise an MENCMachining operation, or try to save it as a new item, by default the attached NC programs are referenced, not copied. This can lead to problems when you modify the NC program as you actually modify the original program.

To resolve this you can set the `MEAActivityReferenceNonActivityChildren` to `FALSE` to copy the NC program dataset as a new object. When you modify the new dataset, the original remains intact.
Chapter

6 Planning resource management

Planning resource management ........................................ 6-1
Setting access rights .................................................... 6-1
Using the Manufacturing Resource Library ....................... 6-1
Install NX templates .................................................... 6-2
Enable graphics creation .............................................. 6-3
Enable tool retrieval from Teamcenter to NX .................... 6-3
Exporting the classification hierarchy ............................... 6-4
Create mapping views for NX ......................................... 6-4
Manually extend the Classification hierarchy with a new class .... 6-5
Building tool assemblies ................................................. 6-6
Build a tool assembly .................................................. 6-7
Understanding propagated attributes ............................... 6-10
Adding components using the guided component search ........ 6-11
Map vendor catalog tools to customer hierarchy ................ 6-11
Retrieve a tool from Teamcenter in NX ............................ 6-12
Chapter 6  Planning resource management

Planning resource management

In Teamcenter, you can manage resource assemblies then use them in your NX CAM session. You can assign resources to your process plan in Teamcenter or assign them to NX operations and they are saved with that operation in the process plan.

Users of resources fall into four general categories:

- Classification administrators
  Users with this role define the class hierarchy to classify resources. This task includes defining the attributes and properties.

- Resource authors
  Users with this role define new resource components, assemble components into standard tool assemblies, and enter the attributes properties of the classified resources.

- Planning engineer
  Users with this role are manufacturing engineers who define process plans that use resources already defined by the authors. They assign resources such as the machine and tools to the process steps (planning operations).

- Resource consumers
  Users with this role are NC programmers using resources for the CAM system. They assign the machine to the CAM setup and use tool assemblies to generate tool paths and use the resources for simulation and tool path verification.

Setting access rights

When setting up resource management, Siemens PLM Software recommends you create one group, for example, a manufacturing group, for all users involved with resource management. Using this group, you can set up custom read and write privileges for the group, as well as defining group-specific workflows and business rules. You should import the Manufacturing Resource Library (with tools and machines) using this group.

Using the Manufacturing Resource Library

To assist you in filling the classification hierarchy with data, you can use the Manufacturing Resource Library. When you install this library, the data is organized
in a detailed classification structure of manufacturing data such as tools, machines, fixtures, assemblies, and components, including tool graphics.

The tooling classification tree contains two hierarchies—a catalog hierarchy (Vendor Catalog) that can contain tool components found in major tool vendor catalogs and a customer hierarchy that contains only those components and assemblies used at your site. You can copy relevant components from the catalog into the customer hierarchy using the mapping definitions that are delivered with the vendor catalog tooling library.

For more information, see the Resource Manager Guide.

**Note** Vendor catalogs are provided separately and are not part of the Manufacturing Resource Library installation kit. Contact your Siemens PLM Software representative for more details.

**Warning** Do not modify the catalog data as these changes are lost when the Manufacturing Resource vendor catalog libraries are updated. Save all your customer-specific components and assemblies in the customer hierarchy.

A sample of part family templates that you can use to quickly create graphics are included in some component classes. You can use these for evaluation purposes only. To find the classes to which part family templates are assigned, search the class hierarchy for the DIN4003 alias name using the quick search feature.

For more information, see the Resource Manager Guide.

For more information about additional part family templates for component classes, contact your Siemens PLM Software representative.

The Manufacturing Resource Library is provided with the Teamcenter installation image. It is updated regularly.

For more information about installing the Manufacturing Resource Library, see the Installation on Windows Servers Guide.

**Install NX templates**

If you have an empty database with no NX templates installed, you must install these templates before you can work in the manufacturing environment.

**Warning** Do not perform this procedure if you already have templates stored in the database. They will be overwritten.

Before you perform this step, ensure that the UGSTRUCTURES feature is installed in your NX installation. When you complete this procedure, the NX templates are available the first time you enter NX.

1. Open a Teamcenter command prompt window.

2. Type:

   ```
   cd /d "%UGII_ROOT_DIR%\templates\sample"
   tcin_template_setup -u=user -p=password
   ```
Enable graphics creation

An administrator typically performs this step.

1. In the Teamcenter Environment Manager (TEM), install the Teamcenter Integration for NX.

2. Restart TEM and install the graphics builder by selecting the **NX Graphics Builder** option in the **NX Integration** section.

3. If you use the Manufacturing Resource Library:
   a. Run setup program in the
      `advanced_installations\resource_management\server` directory of the installation image.
      
      For more information about installing the Manufacturing Resource Library with this option, see the *Installation on Windows Servers Guide*.
   
   b. Select the **Manufacturing Resources — Database Population** option and provide the appropriate input to import tools until you reach the **Support GRAPHICS BUILDER** step.
   
   c. Select **Tools**.
      
      This step installs the part family templates required to create graphics for the tool components found in the Manufacturing Resource Library.
   
   d. Complete the installation.

   **Note**  When you create graphics using the graphics builder, all part family members are created with the same item type as the part family template. Therefore, ensure that your part family templates are of the same item type as the one you specified during the Manufacturing Resource Library import.

Enable tool retrieval from Teamcenter to NX

An administrator typically performs this step.

1. In the Teamcenter Environment Manager, install the Teamcenter Integration for NX.

2. Run the Manufacturing Resource Library setup program in the `advanced_installations\resource_management\server` directory of the installation image, selecting the **Configure NX Library** option.

   This step copies the `cam_part_planner_mrl.dat` configuration file to the `MACH\resource\configuration` directory in your NX installation.

   For more information, see the *Installation on Windows Servers Guide*
Exporting the classification hierarchy

You can export the classification hierarchy using the `mrm_export_resources` utility. This utility exports the classification hierarchy, including class and hierarchy images, to a DEF file. This file can be read by NX CAM.

By default when you export the classification hierarchy and use the DEF file in NX, all Teamcenter attributes are shown in the Search Criteria and Search Results dialog boxes. You can limit the number of attributes shown in these dialogs by creating mapping views to your tool assembly classes in Classification Administration that only contain the attributes that you need in the NX dialog boxes.

If you create a new attribute that must be passed to an NX CAM tool, you must modify the appropriate TCL file as explained in Customizing NX Library in the Manufacturing General documentation in NX.

Tip Schedule a task to run the `mrm_export_resources` utility regularly to keep your NX tooling data up-to-date.

Create mapping views for NX

You can limit and organize the number of attributes you show in the Search Criteria and Search Results dialog boxes in NX CAM by creating mapping views for classes.

1. In Teamcenter Classification Administration, select the class whose attributes you want to limit.

2. In the Class Details pane, click Add View.

3. In the Add View for Class dialog box, select a view type.

<table>
<thead>
<tr>
<th>View type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NXLIB_SearchCriteria</td>
<td>Specifies which attributes are displayed in the</td>
</tr>
<tr>
<td></td>
<td>Search Criteria dialog box in NX.</td>
</tr>
<tr>
<td>NXLIB_SearchResult</td>
<td>Specifies which attributes are displayed in the</td>
</tr>
<tr>
<td></td>
<td>Search Result dialog box in NX.</td>
</tr>
</tbody>
</table>

4. Type a name for the view in the View ID box and click OK.

5. Click the View Attributes tab.

   The View Attributes pane displays all the attributes available in the selected class.

6. Select the attributes you want to display in the NX dialog boxes in the Class Attributes list and move them to the View Attributes list using the left and right arrows.

7. Arrange the attributes in the View Attributes list in the desired order by selecting them and clicking the up and down arrows.

8. Save the view.

9. Export the classification hierarchy using the `mrm_export_resources` utility.
Teamcenter lists the attributes that you specify in the NXLIB_SearchCriteria view as parameters of the DIALOG statement in the DEF file. It writes the attributes that you specify in the NXLIB_SearchResult view as parameters of the RSET statement. If you do not create an NXLIB_SearchResult view, Teamcenter automatically uses the same attributes for the Search Result dialog box as it does for the Search Criteria dialog box. If you do not specify either of these views, all Teamcenter attributes are displayed in both dialog boxes.

Tip Use the mrm_export_resources utility to export hierarchy images for use in the NX dialog boxes.

For more information about creating classification views, see the Classification Administration Guide.

**Manually extend the Classification hierarchy with a new class**

If you extend your Teamcenter Classification hierarchy with new tool assembly classes, these must be reflected in the NX Library dialogs so that you can search in the new classes from within NX. Because NX supports only a limited number of tool types, you must map every new tool assembly in Teamcenter to a tool type in NX.

For more information about NX Library, see the Manufacturing General Help in the NX Help collection.

You can extend the hierarchy manually, or you can use the mrm_export_resources utility to perform these steps automatically.

For more information, see Exporting the classification hierarchy.

The following steps show how to manually create a new class, Custom_Mill_Tool, to hold 5-parameter milling tools. The new class ID is TA_MILL_10_75.

1. Create a new class in the Classification Administration application. Keep in mind that the NX Library mechanism only reflects tool assemblies found in the Resource Management→Tools→Assemblies branch of the classification hierarchy. For the example, you must create the Custom_Mill_Tool class within Resource Management→Tools→Assemblies→Milling.

   For more information about how to create a class in Classification, see the Classification Administration Guide.

2. Add all necessary class attributes. Every tool type within NX requires its own minimum set of attributes. These are based on the attributes that appear in the tool dialogs within NX CAM.

   For more information about the mandatory parameters necessary for a five parameter milling tool, see Customizing NX Library→Creating New Classes & Subclasses→Mandatory Parameters for Classes in the Manufacturing General Help in the NX Help collection.

   Note Negative attribute numbers are reserved for Siemens PLM Software attributes. If you add custom attributes, use positive attribute IDs.

3. Open the MACH\resource\library\tool\inclass\dbc_mrl_tooling_library_tlas_en.def file in an editor. You can find this file in the directory where NX is installed.
4. Add the new class to the class hierarchy section of this file as explained in *Entering the DB Alias in the Class Hierarchy* in the NX Library Help.

5. If you assigned customer-specific attributes that are required by the NX CAM tools, adjust the event handler file to map to these attributes in the DBC retrieve procedure in the `dbc_mrl_tooling_library_tlas_en.tcl` file.

6. Restart NX and Teamcenter.

   The new class now appears in the NX Library dialog boxes when you search for a tool in NX.

**Building tool assemblies**

When building new tool assemblies in Resource Manager, search first in your customer hierarchy to find existing components that you can reuse. If you do not find any, continue to search in the vendor catalog, and then map that component to the customer hierarchy.

The following chart shows the approach that enables the maximum amount of reuse of existing material.
**Build a tool assembly**

1. Create a new item that represents the root of the new assembly structure by choosing **File→New→Resource** or clicking 📄.

2. Specify the item type for the new resource.
   The following four types are generally used to create a new resource.
   - **MENCTool** — for tools
   - **MENCMachine** — for machines (do not confuse with MENCMachining)
• **MEResource** — for resources

• **MEEquipment** — for use with Tecnomatix applications

Teamcenter stores the default value of this entry in the **MRMIItemTypes** preference.

3. Search for components to build the tool structure. You must start with the machine-side component—that is, a machine adapter or tool holder—and work toward the cutter or workpiece side.

   a. Select the parent component below which you want to add the next component.

   b. Search for the next component using the **Classification Search Dialog** and click **OK**.

   **Tip** Turn on the graphical browser for a more visual search experience.

   For more information, see the **Classification Administration Guide**.

   Teamcenter displays the new component as a child of the selected component.

   For more information about searching the classification hierarchy, see the **Resource Manager Guide**.

4. (Optional) If you put a component in the wrong position in the hierarchy, or prefer to search in a different order from the assembly structure (for example, search for the cutter first), cut and paste components to the correct position in the tool assembly.

5. Attach the propagation start point (PSP) to the cutter by selecting the cutter and clicking 🗑️.

6. Click **Save** 🗑️.

Teamcenter saves the assembly and propagates the defined attribute values from all components, starting at the propagation start point (in this case, the cutter), to the assembly root node. The following figure shows the propagated attributes in italic font.
### Attribute values

<table>
<thead>
<tr>
<th>Tool Description</th>
<th>DES</th>
<th>00 Turning Left 80°</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comments</td>
<td>COM</td>
<td></td>
</tr>
<tr>
<td>Company Code</td>
<td>FEN</td>
<td>PT</td>
</tr>
<tr>
<td>Supplier</td>
<td>SUP</td>
<td></td>
</tr>
<tr>
<td>Plant</td>
<td></td>
<td>3 Cologne</td>
</tr>
<tr>
<td>Machine Group</td>
<td></td>
<td>120 Maschine Group 120</td>
</tr>
<tr>
<td>Status</td>
<td></td>
<td>23 Test</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NX Tool Type</th>
<th>NX 03:01 NX Turning Tool Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orientation Angle</td>
<td>OA</td>
</tr>
<tr>
<td>Insert Position</td>
<td>IP</td>
</tr>
</tbody>
</table>

### Tracking Points

<table>
<thead>
<tr>
<th>Rad, ID</th>
<th>Tracking Point TR</th>
<th>Nomin. Setup X</th>
<th>Nomin. Setup Y</th>
<th>Adjust A</th>
<th>Cutcom C</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Holder Orientation</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Machine Adapter</th>
<th>MA0 120 Parallel ZV40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert Shape</td>
<td>IS C Rhombic '80d.</td>
</tr>
<tr>
<td>Inscribed Circle Diameter</td>
<td>IC 12.700</td>
</tr>
<tr>
<td>Corner Radius</td>
<td>IR 0.800</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cutting Edge Length</th>
<th>IL 12.900</th>
</tr>
</thead>
<tbody>
<tr>
<td>Insert Included Angle A1</td>
<td>80.000</td>
</tr>
<tr>
<td>Insert Thickness ISO IT</td>
<td>4.76 (4)</td>
</tr>
<tr>
<td>Clearance Angle Major A3</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Warning Time</th>
<th>W0 Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Tool Life</td>
<td>T0</td>
</tr>
<tr>
<td>Tool Material ID</td>
<td>M004</td>
</tr>
</tbody>
</table>

### Planning resource management

Getting Started with Manufacturing 6-9
Note  This procedure works correctly for all existing tool assembly classes in the Manufacturing Resource Library. If, however, your administrator adds new classes to the classification hierarchy, the class in which you classify the resource assembly must be specified as an assembly class in the Classification Administration application. In addition, the attributes that you want to propagate to the root node must have the **Propagated Property** option selected in the resource assembly class in Classification Administration.

For more information, see the *Classification Administration Guide*.

### Understanding propagated attributes

The following table demonstrates attribute propagation. In this example, the **Test Assembly 1** class is specified to be an assembly class in Classification Administration. Furthermore, in the assembly class, **Attributes 1–5** have the **Propagated Property** option selected.

<table>
<thead>
<tr>
<th>Attributes</th>
<th>Propagated from</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute 1</td>
<td>Component 3</td>
</tr>
<tr>
<td>Attribute 2</td>
<td>Component 1</td>
</tr>
<tr>
<td>Attribute 3</td>
<td>Component 3</td>
</tr>
<tr>
<td>Attribute 4</td>
<td>Component 4</td>
</tr>
</tbody>
</table>
Attribute 5 contains no value. Although there is a value assigned to this attribute in Component 2, Component 2 does not lie within the propagation path as it is not a parent or grandparent of Component 4, the propagation start point. Similarly, Components 6 and 7 do not lie within the propagation path.

A gray PSP indicates that this PSP is defined in the context of a subassembly. It is not taken into consideration in the current tool assembly. Subassemblies are highlighted with a colored background in Resource Manager.

Adding components using the guided component search

Using a guided component search (GCS) accelerates the search for matching components within an assembly. This search allows you to choose from a list of only those classified workspace objects that fit into the selected components from which you are initiating the search. Whether a component fits into something else could be based on physical criteria or on other criteria that you decide (for example, Material A can only be used with Material B). To use this search, you must first configure it in the Classification Administration application.

For more information about configuring the guided component search, see the Classification Administration Guide.

The guided component search uses the following criteria to determine if a component matches another:

- If the connection type of each component is the same.

- If components fit into each other. All components in the GCS are assigned a physical shape—plug, socket, or neutral. For the search to find a match, the matching component must always have the opposite shape to the initial component. In other words, a plug component requires a socket component and vice versa, or a neutral component requires another neutral component.

- The attribute values of the components must fulfil the matching criteria set in the Classification Administration application.

For more information about running a guided component search, see the Resource Manager Guide.

Map vendor catalog tools to customer hierarchy

1. Select the class in the vendor catalog hierarchy that you want to map to a customer class.

2. Search for and select the appropriate ICO from the vendor class.

3. Click 

4. If there are multiple possible target classes specified in the mapping definition, the Target Class Selection dialog box is displayed:
   a. Choose the desired target class from the list.
   b. Click OK.
Chapter 6  Planning resource management

The New Item dialog box is displayed.

5. Type a unique identifier for the new item in the Item ID box adhering to your company's naming conventions or let Teamcenter automatically create a name for you by clicking Assign.

6. Type a name in the Name box.

7. Select the correct item type from the list, for example, MENCTool.

8. Click OK.

Teamcenter creates a new ICO and item in the customer class and maps attributes from the vendor class to attributes in the customer class as specified by the mapping definition in Classification Administration. The new ICO is displayed in the customer class and its attributes are shown in the Properties pane.

Retrieve a tool from Teamcenter in NX

An NC programmer typically performs this step.

1. If you have not opened a part in NX before, initialize the manufacturing environment.
   a. In NX, create a new part and choose Start→Manufacturing. NX displays the CAM Session Configuration dialog box.
   b. Select cam_part_planner_mrl from the list of possible configurations. NX displays the Library Class Selection dialog box.
   c. Select a CAM operation setup template.

2. Open the machine tool view.

3. Click Create Tool or double-click the root machine.

4. Click Retrieve Tool from Library.

   The Library Class Selection dialog box is displayed. This lists the tool assembly classes found in the Manufacturing Resource Library hierarchy that is stored in the Teamcenter database.

5. Select the class in which you want to search.

   NX displays the Search Criteria dialog box containing a list of attributes belonging to the class that you selected in the previous step.

6. Narrow your search by entering parameters in the dialog box and click OK. NX displays a list of results.

7. Select the desired tool assembly and click OK.
Chapter

7 Managing the enterprise BOP

Managing the enterprise BOP .................................................. 7-1
Overview .............................................................................. 7-2
Prerequisites for managing the enterprise BOP ....................... 7-5
Understanding logical designators ......................................... 7-5
  Defining logical designators for types ................................. 7-6
  Understanding logical assignments .................................... 7-6
Understanding the generic BOP .............................................. 7-7
Understanding the product BOP ............................................. 7-8
Understanding the plant BOP ................................................. 7-8
Understanding partitions ...................................................... 7-9
Understanding origins ........................................................ 7-9
Creating operations, processes, and partitions ....................... 7-10
Consuming parts .................................................................. 7-11
Allocating processes and operations to a plant BOP ............... 7-11
Assigning tools .................................................................... 7-12
Visualizing a process or operation ....................................... 7-12
Understanding time analysis ............................................... 7-13
Comparing and propagating differences in enterprise BOP structures ........................................... 7-13
Importing and exporting EBOP structures using PLM XML .... 7-14
Managing the enterprise BOP in a Multi-Site Collaboration environment ..................................... 7-16
Configuring the enterprise BOP ............................................. 7-16
Chapter

7 Managing the enterprise BOP

Managing the enterprise BOP

The enterprise bill of process (EBOP) capability allows you to capture standards and best practices across the business as generic processes and operations. You can reuse these generic definitions multiple times when planning the manufacture of any product or specific models.

To implement EBOP, you should create the following data:

• A generic BOP or process plan that represents the processes for building a generic product for a type of product. For example, you can define a generic process plan for building any model of pickup truck. You build this process plan from standard tools, processes, and operations, excluding elements that are unique to a particular model.

• One or more product BOPs. This process plan represents the processes for building a particular model. It is derived from the generic BOP for the corresponding type of product, allowing for commonality and reuse between different products of the same type.

• One or more plant BOPs. This process plan represents all processes or operations that are performed at a specific plant or work station. Processes and operations are allocated from a product BOP to a plant BOP that is then used for analysis and validation of the process plan. This can include consideration of nonvalue-added operations that are required in the specific plant, time analysis, line balancing of a given product mix, simulation, and derivation of precise requirements for tools and worker skills.

• Partitions to organize groups of processes within the generic BOP and product BOP into a hierarchical structure, for example, a partition may contain seat system processes or chassis processes.

• Logical designators to allow the same partition, process or operation to be mapped to different generic or product BOPs.

You can adapt the product BOP for different plants that will manufacture the same product, while maintaining maximum commonality between the manufacturing processes at each plant.
Overview

An enterprise BOP structure is built from processes, partitions, and operations. It can be created from these components, modified, and managed in Manufacturing Process Planner using similar techniques to other BOPs.

- **A partition** is a group of processes in a generic BOP or a product BOP, but is not a process itself. It provides a means to organize processes with a similar purpose for easy access, for example, all seat system processes may be placed in one partition and all chassis processes in another. A partition may have a logical designator to allow the mapping of partitions across different product BOPs. It breaks a BOP into a navigable structure.

  The content of a partition is not tracked over time, so there is no revisioning of partitions. Similarly, configuration information (variant expressions or effectivity) should not be applied to the partition itself.

A partition may contain other partitions.

- **A process** is an ordered sequence of operations with a single purpose. It represents the complete content of the assembly or installation procedure for a subsystem, for example, **install the brake dust shield to knuckle**. Each process has an output, typically the built or installed subsystem. In final assembly, a process normally contains 5 to 10 operations.

  The order of operations under the process represents the logical order in which the shop floor operators execute them. Operations should also be arranged in a logical sequence in the generic BOP and product BOP.

- **An operation** is the smallest manageable unit of work that can be allocated to the plant, for example, **obtain a tail light** or **tightly a screw**. It can be broken down into activities. It can also consume or handle parts, and you can assign resources to it.
The logical designator of an operation is typically derived from the logical designator of the owning process.

- A partition, process, or operation has a *logical designator* that allows identification of partitions, processes, and operations with the same purpose in different BOPs. For example, the process to assemble a headlight may have the same logical designator in all BOPs. The process logical designator is derived from the part consumption and the purpose of the process. For example, the assembly process for a subsystem has a different logical designator to the installation process for the same subsystem. Conversely, two processes that achieve the same goal share the same logical designator, for example, manual and automated installation processes for the same subsystem.
Prerequisites for managing the enterprise BOP

Before you start to develop the enterprise BOP, logical designators must be defined in the product structures that interact with the generic BOP and product BOP structures. A logical designator comprises the two following fields on absolute occurrences of the product structure:

- **Usage address**
  The run-time property name is `bl_usage_address`.

- **Logical position**
  The run-time property name is `bl_position_designator`.

You define these fields using standard Teamcenter product definition tools such as Structure Manager. An administrator can assign lists of values (LOVs) to each of these fields to limit possible selections to predefined values.

Understanding logical designators

A logical designator comprises a combination of the usage address and position designator (logical position) properties on a structure line. Enterprise BOP uses the values of these properties to resolve logical assignments.

For example, all partitions of processes that assemble the air conditioning system in any car may have the same logical designator, indicating their commonality. Logical designators are represented by the `MELDBaseForm` class.

Processes consume products in an operation according to their logical designators. For example, the process may consume the left mirror of the car. This logical consumption can be materialized into a specific mirror later.

**Note** Logical designators may not be unique in the `product` structure, and more than one product line may have the same logical designator.

The logical designator of an operation or process is a combination of the usage address and logical position properties of an absolute occurrence in the structure. The logical designator is a form. The administrator uses Business Modeler IDE to define a standard pattern in the form property fields that all logical designators must follow. These fields can be implemented as LOVs (list of values). For example, the format `DSSNNNNN` specifies:

- The manufacturing domain (`D`), which may be `B` (body in white), `P` (paint), or `F` (final).

- The subsystem (`SS`). For example, all brakes subsystems may be prefixed with `A`, so that `AA` specifies a front drum brake, `AB` specifies a rear drum brake, `AC` specifies a front disc brake, `AD` specifies a rear disc brake, and so on.

- A unique sequential number that identifies a single logical designator (`NNNNN`).

A logical designator has a separate lifecycle to the process, operation, or partition to which it is attached. You must have write access to the process, operation, or partition but, once the form is attached, you can update the logical designator form without write access to the related object.
Chapter 7  Managing the enterprise BOP

If you clone a process structure or create one from a template, the logical designator is duplicated to avoid the sharing of logical designators.

You can define access control rules that restrict access to the logical designator relation, allowing only certain users or groups to attach or detach a logical designator to or from a process, operation, or partition.

For more information about using logical designators, see the Manufacturing Process Planner Guide.

Defining logical designators for types

You can use Business Modeler IDE to extend the basic logical designator form and define different form types for processes, operations, and partitions. You specify three different constant values to define a specific subclass for process, operation, and partition objects. Teamcenter can then identify the logical designator that belongs to each class during creation, searches, and other procedures. If you define different form classes for processes, operations, and partitions in this way, Teamcenter automatically creates the appropriate form when you create a process, operation, or partition in the context of a generic BOP or product BOP.

Caution

Once you define form classes in this way and instances of them are attached to processes, operations, or partitions in the database, Teamcenter does not prevent you from changing their content. The existing instances are not updated or replaced. This is not good practice because it results in inconsistent logical designator data and should be avoided. The logical designator form is hidden in the Manufacturing Process Planner and My Teamcenter user interfaces to avoid its inappropriate manipulation.

The new form classes and any derived classes are visible in any application that allows the creation of forms, not only in Manufacturing Process Planner. However, they have no special significance elsewhere.

The logical designator form contains 32 characters, and its default value is logical_designator. If this schema is not sufficient for your business, the administrator can use Business Modeler IDE to add other fields to the default form or to any customized forms you create.

Understanding logical assignments

A logical assignment models the reference between an operation revision or process revision and the product logical designator that is assigned to the operation or process. When you assign a product logical designator to an operation revision or process revision, an instance of a logical assignment is created to capture the assignment. The logical assignment is represented by the MELogicalAssignment class, and the logical assignment object is attached to the operation revision or process revision by a Logical Assignment relation.

A logical assignment can be resolved to (regular) assignments of the operation or process. When this occurs, the resolved assignments are appended to the operation or process in addition to the logical assignment and do not replace it; the logical assignment remains even after it is resolved. If the logical assignment is already resolved, the action of resolving it again updates the resolved assignments list; missing resolved assignments are added and redundant resolved assignments are
removed. (Resolved assignments that are created by resolving another logical assignments and direct assignments are not updated.)

The logical assignment holds the following information:

- A fully specified or partially specified product logical designator. When logical assignments are created, it is not required for the product lines that match the logical designator to already exist. You can define the logical designator to suit the allowed values of the product logical designator. Teamcenter only attempts to match the definition with actual product lines when the logical assignment is resolved.

- An assignment type that defines the nature of the assignment. This is used when the logical assignment is resolved as the type of resolved assignment to be created.

- A reference to one or more resolved assignments that are created when the logical assignment is resolved, if they exist. This reference is provided by a resolved assignment relation called `MEResolvedAssignmentRel`.

The logical assignment is owned by the operation revision or process revision. When you add a logical assignment to or remove a logical assignment from an operation revision or a process revision, you must have the appropriate permissions to the revision. Also, you require the same permissions if you modify the information on the logical designator, for example, change the assignment type.

The logical assignment is checked in or out whenever the associated process or operation is checked in or out. It acquires the same access rights as the process or operation. It is deleted from the database if the process or operation is deleted.

You can view the logical assignment of a process or operation with the process assignments viewer. Logical assignments are not visible in other viewers, including`Referencers` panes.

**Understanding the generic BOP**

A generic BOP is a process structure that contains the generic processes for manufacturing a product family, for example, assembling any truck. It contains processes and is organized by partitions. It is represented by the `MEGenericBOP` class.

A generic BOP is a context. It defines a space in which processes, operations, and partitions of the process plan reside. The context contains the processes and operations, on which partitions impose a structural hierarchy. Contexts cannot be nested inside each other.

Use the generic BOP to:

- Create and partition generic processes and operations.

- Create logical product assignments on generic operations.

- Assign tools to generic operations.

- Provide a template for product BOP creation.
You can track the content of a generic BOP root item to identify what processes are added or removed. Partitions are not tracked in this way.

You can associate a generic BOP with a generic BOM (a generic product definition), if you model the generic BOM as a skeleton product structure that represents the breakdown of the product.

You can manually partition the generic BOP and put processes in these partitions, to create hierarchical groups to allow easier data management. Each process can reside in only one partition. There is always a single root partition that cannot be removed.

For more information about creating and managing a generic BOP, see the Manufacturing Process Planner Guide.

Understanding the product BOP

A product BOP is a process structure that contains the processes for manufacturing a certain product family, for example, a specific model of truck. It is represented by the MEProductBOP class. If appropriate, you can use the generic BOP as a template for the product BOP. You can also create a product BOP from another product BOP.

Use a product BOP to:

- Create and partition product-specific processes and operations.
- Consume products directly or by resolving logical assignments.
- Assign tools to product-specific operations.
- Perform accountability checks against the product structure.
- Make comparisons with the generic BOP or other product BOPs.
- Perform time and feasibility analysis.

You can define access permissions that are required to add objects to or remove them from the product BOP. You can also track the content of a product BOP root item to identify what processes are added or removed. Partitions are not tracked in this way.

You can create multiple product BOPs from a single generic BOP. Each product BOP has as its origin the same generic BOP.

You can manually partition the product BOP and put processes in these partitions to create hierarchical groups to allow easier data management. Each process can reside in only one partition. There is always a single root partition that cannot be removed.

A product BOP is a context, as configured in Business Modeler IDE. It defines a space in which processes, operations, and partitions of the process plan reside. The context contains the processes, operations, and partitions, but does not impose a structural hierarchy on them. Contexts cannot be nested inside each other.

Understanding the plant BOP

A plant BOP is a process structure that contains all processes and operations required to manufacture one or more products at a particular plant. It is arranged in
a hierarchy of process areas, each representing an individual work station. A plant BOP is represented by the MEPlantBOP class.

You allocate processes and operations from a product BOP to a plant BOP. This allocation creates a clone of the process or operation in the plant BOP and retains a reference (origin relation) back to the product BOP. You can run an accountability check between a product BOP and plant BOP to see what has changed. You can then propagate the changes from the product BOP to the plant BOP. Teamcenter uses the origin relation to perform this propagation.

Processes and operations that you allocate are considered to be standardized, allowing your company to take reuse existing procedures. You can, however, override any data in the context of the plant BOP. This includes part assignment, tool assignment, attachments, time analysis, child processes, flows, properties, logical designators, and logical assignments. Additionally, you can create work station-specific processes and operations. Such processes and operations are not considered in the propagation mechanism.

Understanding partitions

Process partitions allow you to define a hierarchical breakdown of objects in a context, which may be a generic BOP or a product BOP. The partition provides a container for process objects in the BOP, while the partition structure captures their hierarchy. The process partition is represented by the MEPProcessPartition class.

Partitions cannot be revised. They can be extended in Business Modeler IDE and controlled with Access Manager in the same way as any other occurrence group. Adding an object to or removing an object from a partition requires write access to the partition.

A partition may be owned by a user or groups of users. If the ownership of a partition changes, the ownership of the substructure does not automatically change. You must manually reassign ownership of the subpartitions and processes in the partition.

A user must have permission to modify the generic BOP or product BOP context before creating a process inside a partition. Typically, you give all contributors write access to the generic BOP or product BOP context. You can then restrict users to specific areas of responsibility by defining permissions on the partitions.

If you remove an object from a partition, you also remove it from the generic BOP or product BOP context. If you remove an entire partition, all its contents are recursively removed, including all subpartitions and processes.

Note Teamcenter automatically creates generic BOPs and product BOPs with a root partition.

Understanding origins

When you use a generic BOP as a template for the creation of a product BOP or allocate processes or operations from a product BOP to a plant BOP, Teamcenter creates an Origin relation between the structures. It also creates Origin relations between each individual process, operation, and partition in the product BOP and the corresponding origin in the generic BOP. Similarly, it creates Origin relations when you clone a substructure of a generic BOP to a product BOP using the Paste
Chapter 7  
*Managing the enterprise BOP*

**Duplicate** command. Teamcenter uses these relations to trace objects in the product BOP to their counterparts in the generic BOP when you compare the structures using accountability checking.

**Note**  When using the **Paste Duplicate** command to clone processes or operations from the product BOP to the plant BOP, Teamcenter does *not* create origin relations.

You cannot delete a partition if it is the origin of another partition. Consequently, you cannot delete a generic BOP if it is the origin of any product BOP. You can, however, delete processes or operations that are origins of other processes or operations, but the cloned processes or operations are still marked as having an origin (for example, they still have an allocation indication symbol).

For more information, see the *Manufacturing Process Planner Guide*.

### Creating operations, processes, and partitions

When you create a new product BOP, it inherits all partitions, operations, and processes that are defined on the generic BOP or template. All properties, attributes, and relationships are carried over, including logical designators, tool assignments, variant expressions, and sequence numbers. The system automatically associates parts in the selected product definition with operations according to logical designator and part function.

All partitions, operations, and processes in the new product BOP are linked to the corresponding objects in the generic BOP. If you create a product BOP from another product BOP, the objects are also linked to the original generic BOP.

**Note**  If you create a product BOP from a template, no links are maintained. Consequently, any changes to the original template do not affect the new product BOP.

If variants are defined on the original BOP, they are carried over to the new product BOP by default. You can optionally choose to not carry over variants. This selection applies to process variants and part consumption by logical designator and variant condition.

If some logical designators are missing from the product definition, the operations continue to reference them, but without explicit parts. You can resolve the consumed logical designators and part functions later against the product structure.

You can manually override values copied from the template BOP later, if appropriate, as follows:

- Adding operations to or removing operations from a process
- Adding processes to or removing them from a partition
- Changing part consumption or tool assignments
- Changing operation or process names or descriptions
- Changing variant expressions or operation sequences

You cannot override the logical designator of a process, operation, or partition.
You can also clone operations or processes (including their properties) from the library to manually add to the new product BOP.

### Consuming parts

You can consume parts in several ways:

- Assign a part function. For example, you can consume a wheel, regardless of where it is used on the vehicle.
- Assign a logical designator. For example, you can consume the left front wheel, regardless of any variants.
- Assign a combination of logical designator and variant expression. For example, you can consume the left front wheel, but only if it is the magnesium variant.
- Assign a physical part from the product definition, consuming a specific part occurrence.

The general procedure is as follows:

1. Select the parts and assign them to the process or operation.
2. Define the part assignment type. The available assignment types are **Consume**, **Assemble**, **Disassemble**, **Assign**, and **Handle**.
3. Assign a logical designator or part function to the process or operation.
4. Manually resolve part function or logical designator assignments to physical parts, where the system did not update assignments automatically.
5. View the assigned parts in the BOP to verify correct resolution.
6. Propagate any product definition changes to refresh logical designator or part function to physical consumed part assignments.
7. Run an accountability check between the BOP and associated product definitions.

Processes are automatically configured to suit the product configuration whether you assign parts by logical designator or explicit selection of parts.

### Allocating processes and operations to a plant BOP

When you allocate processes and operations from a product BOP to a plant BOP, Teamcenter creates clones of the objects and adds them to the plant BOP. You can allocate both processes and operations. If you allocate processes, you allocate:

- All child operations of the allocated process.
- All parts consumed in the product BOP. These are also consumed in the plant BOP.
- The flows between operations.
• The deeper hierarchy levels (processes below allocated processes).

• In-context data (in the context of the allocated process or operation or lower contexts) that is configured in the product BOP.

Teamcenter maintains an origin link between allocated processes or operations. If there is a change to the process or operation in the product BOP, you can find these by performing an accountability check and propagating the change to the plant BOP.

For more information, see the Manufacturing Process Planner Guide.

Assigning tools

You can search for a classified tool in a library or an unclassified tool in a directory folder. Once found, you can add the tool to an operation, where it appears in the process tree. You can also view the tool in the 3D viewer.

Values can be attached to an occurrence of a product, work area or occurrence group that is assigned to a process or operation. The value indicates the purpose of the assignment. You can use Business Modeler IDE to customize the list of values available at your site. The following default values are provided for enterprise BOP systems:

• MEConsumed
• MEAssemble
• MEDisassemble
• MEAssign
• MEHandle

Teamcenter must be able to distinguish product assignments from tool assignments. Because occurrence types can be customized, you must set several preferences to define certain assignment types, as described later.

Visualizing a process or operation

You can load a process or operation from any BOP into the 3D viewer. All tools and parts with a resolved CAD solution that are associated with an operation are also loaded. If the operation relates to a part function or logical designator that resolves to multiple parts, all such parts are loaded.

The product’s initial position is derived from the product structure, that is, relative to the product’s zero coordinates. If you do not override this position, it is updated from the product definition next time it is loaded.

You can manipulate the positions of tools and parts, and store the new transforms in the context of the operation or process. You can clear any such in-context overrides if you do not want to save them.
Understanding time analysis

You can perform time analysis on any operation or process in the enterprise BOP, including:

- Breaking down an operation into activities.
- Defining the type of each activity. The available types are VA, NVA, and NVABN.
- Adding predetermined time system information on each activity, including a code from the time system (for example, an MTM code), description, frequency, and unit time.
- Defining an estimated time for an operation.
- Obtaining a rollup of the total times of all activities in an operation.
- Defining an estimated time for each process.

You can also define a target time for each control model in a partition or product BOP and configure the structure and obtain a time rollup.

Comparing and propagating differences in enterprise BOP structures

You can compare enterprise BOP structures to identify changes to equivalent partitions, processes, and operations. Equivalence is established by the following criteria:

- Traceability between a partition, process, or operation in the product BOP and the original partition, process, or operation in the generic BOP.
- Traceability between a process or operation allocated from the product BOP to a plant BOP.
- Logical designator. If a BOP has several operations with the same logical designator due to variance or operation splitting, the variant expression is used as a second criterion for judging equivalence.

The definition of a change is user-configurable but may include any of the following:

- Changes in hierarchy, for example, a process is moved into a different partition.
- Differences in part assignment, as identified by comparing consumed part functions, logical designators, and explicit part assignments. (Explicit parts assignments are not considered if the generic BOP is included in the comparison.)
- Differences in tool assignment.
- Changes in operation properties, for example, in activities and their sequence.

You can also compare two versions of the same enterprise BOP structure, for example, to identify differences between the process plans for two different model years.
Once you establish differences using the accountability check, you can propagate these from the generic BOP to the product BOP or from the product BOP to the plant BOP.

**Importing and exporting EBOP structures using PLM XML**

Teamcenter now supports importing and exporting enterprise bill of process structures using PLM XML. This enables downstream applications to leverage the process definition provided in the enterprise bill of process structures.

The following restrictions apply when importing/exporting these structures.

When importing/exporting origins:

- Origins are exported on the primary object only—processes or operations in the plant BOP reference the related processes or operations in the product BOP. Processes and operations in the product BOP reference those in the generic BOP.

- Association between contexts is exported and imported as a relation on the top level items. Exporting this relation is controlled by a transfer mode.

- When importing structures containing origins, Siemens PLM Software recommends:
  - Import all relevant structures (generic BOP, product BOP, and plant BOP) using a single collaboration context object. You can also use this to create new structures.
  - Import only one structure with the origin links. This requires that the origin structure, including all origin operations/processes/partitions already exists in the database. Use external references for the origin so the import does not attempt to modify the origin structure. You can export data using external references using the `MFGEBOPExportExternalRef` transfer mode.

When importing/exporting logical assignments and tool requirements:

- Tool requirements and logical assignments are imported/exported through their owning operation. You cannot import or export them individually.

- The logical assignment query string can be exported in the PLM XML file for use in the downstream application.

When importing/exporting constraints:

- Exporting constraints is controlled by the transfer mode. If only one side of the constraint is exported (for example, the predecessor), the other side (for example, the successor) is not exported by default—only if requested by the transfer mode.

- You can mark constraints as implicit. Implicit constraints occur, for example, when you have **constraint 1** with **operation A** as predecessor and **operation B** as successor and **constraint 2** with **operation B** as predecessor and **operation C** as successor. You have, therefore, the following chain of constraints:
  
  Operation A → constraint 1 → operation B → constraint 2 → operation C

If **operation B** is filtered out, **constraint 1** and **constraint 2** are not shown. Instead, there is an implicit constraint created between **operation A** and
operation C. This is only a calculated constraint and not a new object. It is meant to indicate that there is still some sequence needed between operation A and operation C.

Implicit constraints are not imported.

- You can only remove constraints from the process or operation by using the application interface.

- Creating a new revision for the operation in import adds the constraint between the new revisions automatically.

The following limitations exist:

- The import does not block origin creation if the contexts are not associated.

- The import does not ensure that both sides of the origin have matching types.

- The import does not enforce one origin per operation/process/partition if multiple origins are added in the same PLM XML file.

- The import does not block the ability to disconnect the association between contexts.

- The import does not block adding logical assignments or tool requirements to processes/operations that are not EBOP structures.

- The import does not prevent linking an operation/process as predecessor of itself or creating cycles.

- The import does not prevent creation of dangling constraints (connected only on one side). These are ignored by the application.

- After removal of processes/operations, the import does not delete their constraints (this is the logic that is applied when processes/operations are removed from the user interface). This means that the removed processes/operations still have the constraints, but they are never displayed as they are not accessed from the structure. Also, if the constraint is still attached to other operations that are still part of the structure, this constraint is also filtered out because the constraints API always considers the run-time information, not the persistent information. These constraints are then treated as dangling constraints.

- The import does not block creation of operations/processes/partitions without a logical designator in EBOP structures.

- The import does not prevent changing the top partition to one of the subpartitions.

- The import does not recursively remove all objects below a removed process partition/process.

- The import does not prevent sharing processes/operations/partitions/process areas.

- The import does not prevent adding objects to the generic BOP or product BOP base view only and not to the partition structure.
• The import does not enforce allocation-level logic.
• The import does not enforce structure rules such as no line under station or no process areas under processes.

Managing the enterprise BOP in a Multi-Site Collaboration environment

You can share enterprise BOP data using Multi-Site Collaboration. Generic BOPs and product BOPs are shared using the same behavior as process structures, partitions are shared as occurrence groups in process structures, and process logical designators are shared as forms.

Enterprise BOP structures with origin links to other structures (for example, a product BOP that links to a generic BOP) share the origin structure as remote objects. This eliminates the need to replicate the origin structure at each site.

Configuring the enterprise BOP

You must set a variety of preferences in your manufacturing environment.

For more information, see Setting manufacturing preferences.

In addition, you can use the following preferences, global constants, and cloning rules to configure the operation of the enterprise BOP for your site.

The following global constants and preferences configure the logical designator form.

<table>
<thead>
<tr>
<th>Preference/Constant</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mfg0OperationLD</strong></td>
<td>Specifies the type of form that represents the logical designator of an operation. If empty, logical designators are not created for operations.</td>
</tr>
<tr>
<td><strong>Mfg0ProcessLD</strong></td>
<td>Specifies the type of form that represents the logical designator of a process. If empty, logical designators are not created for processes.</td>
</tr>
<tr>
<td><strong>Mfg0ProcessPartitionLD</strong></td>
<td>Specifies the type of form that represents the logical designator of a process partition. If empty, logical designators are not created for process partitions.</td>
</tr>
<tr>
<td><strong>MELogicalDesignatorDelimiter</strong></td>
<td>Specifies the delimiter in the display string of the logical designator.</td>
</tr>
<tr>
<td><strong>MELogicalDesignatorForm NameSuffix</strong></td>
<td>Specifies the suffix to the logical designator form name.</td>
</tr>
</tbody>
</table>

The following preferences configure cloning of generic BOPs and product BOPs.
<table>
<thead>
<tr>
<th>Preference</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>GenericBOPTemplate</strong></td>
<td>Contains the default cloning rule when using a generic BOP as the template for creating a product BOP.</td>
</tr>
<tr>
<td><strong>ProductBOPTemplate</strong></td>
<td>Contains the default cloning rule when using a product BOP as the template for creating another product BOP.</td>
</tr>
<tr>
<td><strong>PlantBOPTemplate</strong></td>
<td>Contains the default cloning rule when using a plant BOP as the template for creating another plant BOP.</td>
</tr>
<tr>
<td><strong>GenericBOPAllowedTemplates</strong></td>
<td>Contains a list of the cloning rules that are available when using a generic BOP as a template for creating a product BOP.</td>
</tr>
<tr>
<td><strong>ProductBOPAllowedTemplates</strong></td>
<td>Contains a list of the cloning rules that are available when using a product BOP as a template for creating another product BOP.</td>
</tr>
<tr>
<td><strong>PlantBOPAllowedTemplates</strong></td>
<td>Contains a list of the cloning rules that are available when using a plant BOP as a template for creating another plant BOP.</td>
</tr>
<tr>
<td><strong>MEGenericBOP,MEGenericBOP, PasteDuplicateTemplate</strong></td>
<td>Contains the default cloning rule used when executing the Paste Duplicate command from a generic BOP context to another generic BOP context.</td>
</tr>
<tr>
<td><strong>MEGenericBOP,MEProductBOP, PasteDuplicateTemplate</strong></td>
<td>Contains the default cloning rule used when executing the Paste Duplicate command from a generic BOP context to a product BOP context.</td>
</tr>
<tr>
<td><strong>MEProductBOP,MEProductBOP, PasteDuplicateTemplate</strong></td>
<td>Contains the default cloning rule used when executing the Paste Duplicate command from a product BOP context to a product BOP context.</td>
</tr>
<tr>
<td><strong>MEProductBOP,MEGenericBOP, PasteDuplicateTemplate</strong></td>
<td>Contains the default cloning rule used when executing the Paste Duplicate command from a product BOP context to a generic BOP context.</td>
</tr>
<tr>
<td><strong>PasteDuplicateTemplates</strong></td>
<td>Contains a list of cloning rules that are considered by the Paste Duplicate command.</td>
</tr>
<tr>
<td><strong>PasteDuplicateAllowedPlantBOP Templates</strong></td>
<td>Contains a list of cloning rules names to be used when pasting duplicate objects (clones) between different types of plant bills of process.</td>
</tr>
<tr>
<td>Preference</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>MEGenericBOP.Mfg0MEPlantBOP. PasteDuplicateTemplate</td>
<td>Contains the default cloning rule used when the Paste Duplicate command is executed between a generic BOP context and a plant BOP context.</td>
</tr>
<tr>
<td>MEPProductBOP.Mfg0MEPlantBOP. PasteDuplicateTemplate</td>
<td>Contains the default cloning rule used when the Paste Duplicate command is executed between a product BOP context and a plant BOP context.</td>
</tr>
<tr>
<td>Mfg0MEPlantBOP.Mfg0MEPlantBOP. PasteDuplicateTemplate</td>
<td>Contains the default cloning rule used when the Paste Duplicate command is executed between a plant BOP context and a plant BOP context.</td>
</tr>
<tr>
<td>ProductBOPAllocationTemplate</td>
<td>Contains the default cloning rule used when allocating from a product BOP to a plant BOP.</td>
</tr>
<tr>
<td>ProductBOPAllocationCloningRule</td>
<td>Contains the name of the cloning rule used for allocating from a product BOP to a plant BOP.</td>
</tr>
</tbody>
</table>

The following preferences configure the allocation mechanism between product BOP and plant BOP.

<table>
<thead>
<tr>
<th>Preference</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEAllocationLevel</td>
<td>Contains the type of process area that can have an operation or process allocation. If no type is defined, there is no limitation on the operation allocation.</td>
</tr>
<tr>
<td>MEPProcessIconPropertyNames</td>
<td>Specifies whether plant BOP structures display an allocation indication symbol for processes if they were allocated from a product BOP.</td>
</tr>
<tr>
<td>MEOperationIconPropertyNames</td>
<td>Specifies whether plant BOP structures display an allocation indication symbol for operations if they were allocated from a product BOP.</td>
</tr>
</tbody>
</table>

For more information about these preferences, see the Preferences and Environment Variables Reference.

Two cloning actions are provided for enterprise BOP structures:

- **EBOPCreate**
  
  Indicates the origin link relation. For example, this is used when cloning a product BOP from a generic BOP.
• **EBOPReference**
  Indicates that a product BOP origin link is used as a reference to construct the generic BOP link.

For these actions, configure the following copy action rule entries to use when cloning enterprise BOP structures:

• **MapLD.Ignore**
  Map according to logical designator equivalence. If unsuccessful, do not add the occurrence to the clone.

• **MAPLDAO.Reference**
  (Default action) Map according to logical designator equivalence. If unsuccessful, map according to the absolute occurrence ID. If one is not found, the clone references the source.

• **MAPLDAO.Ignore**
  Map according to logical designator equivalence. If unsuccessful, map according to the absolute occurrence ID. If one is not found, do not add the occurrence to the source.

For more information about cloning rules, see *Defining cloning rules.*
Chapter

8 Configuring the manufacturing environment

Configuring the manufacturing environment ............................................. 8-1
Configure the update of multiple product views ........................................ 8-1
Configuring the accountability check ...................................................... 8-2
   Configuring the inclusion rules ....................................................... 8-2
   Configure the accountability check to use net effectivity .................... 8-3
   Customizing the net effectivity user exit ........................................... 8-3
   Running an accountability check or propagation at a specified time ......... 8-4
Configure new categories in the Time view ............................................. 8-4
Configuring an MTM data card .............................................................. 8-6
   Install MTM data cards .................................................................... 8-7
   Create your own data card .............................................................. 8-7
Configuring the TiCon integration ............................................................ 8-8
   Map attributes of time elements to activity properties ......................... 8-8
Configuring publishing of manufacturing data ......................................... 8-9
   Understanding roles ........................................................................ 8-9
      Creating a customized publish role — example ................................ 8-10
   Set preferences and environment variables ....................................... 8-11
   Installing the transfer mode ............................................................. 8-13
   Defining alias files ........................................................................ 8-13
      Building a sample alias file .......................................................... 8-15
         Specifying end objects .............................................................. 8-21
         Navigating through a PLM XML file ........................................ 8-21
      Strategies for working with alias files ......................................... 8-22
      Create a new publishing alias dataset ......................................... 8-23
      Update a publishing alias ........................................................... 8-23
      Load or reload the alias file ......................................................... 8-24
      Create or edit publishing pages ................................................... 8-24
      Creating manufacturing documentation in batch mode .................... 8-25
         Enabling batch processing of publishing pages ......................... 8-25
      Troubleshooting PublishBatch translator .................................... 8-26
Configure the user interface ................................................................. 8-27
   Configure incremental changes ....................................................... 8-27
   Customizing the BOM line quantity ................................................ 8-27
Customizing line balancing reports ................................. 8-28
  Customize the contents of the reports ............................ 8-28
  Line balancing report parameters ............................... 8-29

Configuring standard text ........................................... 8-32
  Set up standard text ............................................. 8-33
  Standard text template ......................................... 8-36
    Using the standard text template ............................ 8-37
    Edit standard text templates ................................. 8-37
  Editing options ................................................... 8-41
    Add BOM line properties ...................................... 8-41
    Add or modify template styles ............................... 8-45
    Remove symbols section ...................................... 8-45
    Add symbol section .......................................... 8-46

Configuring 3D PDF generation .................................... 8-47
  Importing customized templates ................................. 8-48
  Configure 3D PDF report generation ........................... 8-49
  3D PDF Report prerequisites ................................... 8-50

Configuring the comparison of design to manufacturing features ........ 8-50
  Mapping property and attribute groups between design elements and BOM lines .................................. 8-51

Configuring 4D planning ............................................. 8-52
Chapter

8 Configuring the manufacturing environment

Configuring the manufacturing environment

Many features within the manufacturing environment must be configured before you can use them meaningfully in your environment.

Configure the update of multiple product views

Teamcenter uses the Dispatcher infrastructure to update multiple product views. In addition to installing this software using the Teamcenter Environment Manager, you must also install the translator specific to creating multiple product views—the Update Mass Product Views translator:

1. Check that VisAutomationApp is installed on the same machine on which Dispatcher is installed.

2. Set FMS_HOME as a system environment variable with the appropriate value and restart the Dispatcher machine.

3. Install the Update Mass Product Views translator using the TEM installer.

4. Verify that the ACL rule for Dispatcher is set in Teamcenter.
   For more information, see Getting Started with Dispatcher (Translation Management).

5. Open Dispatcher_Root_Directory\Module\conf\translator.xml in a text editor and ensure the isActive attribute in the TcUpdateProductViews extension is set to true.

6. Open Windows services and restart the following two services in the following order:
   a. Dispatcher Scheduler
   b. Dispatcher Module

7. Run Dispatcher client manually by starting Dispatcher_Root_Dir/DispatcherClient/bin/runDispatcherClient.bat.
   Teamcenter opens a console.
Chapter 8  Configuring the manufacturing environment

Configuring the accountability check

Using the accountability check, you can compare two structures in various ways. You can, for example, ensure all lines in the EBOM structure are assigned to the MBOM structure, or you can compare the product structure and the process structure to ensure that all occurrences of product components and features are used in the process structure as consumed items. You can also compare a set of properties that you specify to see if they match. When running an accountability check, you can compare the entire assembly or only a subassembly to the selected process.

For more information about using the accountability check, see the Manufacturing Process Planner Guide.

You can control selection synchronization between the results tables and the BOM tree tables using the MEE-expandToSelection preference. Set it to True if you want a selection on the results pane to select the equivalent line in the BOM. This can be set in the Options dialog box in the Manufacturing pane.

Configuring the inclusion rules

You can filter the source and target structures before you compare them to limit the number of comparisons required in an accountability check, therefore limiting the scope of the search. The filter mechanism is based on closure rules that traverse the structures and take only those objects specified in the closure rules into consideration for the check.

You may need to create new or modify existing closure rules for your own business use case. The closure rules you create for the accountability check are referred to as inclusion rules and are not available for importing and exporting. You can create and modify closure rules in the PLM XML/TC XML Export Import Administration application.

For more information about creating closure rules, see the PLM XML/TC XML Export Import Administration Guide.

Note  Expanding a structure based on closure rules also plays a role in which objects are taken into consideration in the accountability check. If you expand using an appropriate closure rule, you may not have to set an inclusion rule when running an accountability check.

For more information, see the Manufacturing Process Planner Guide.

All the existing closure rules created for the accountability check begin with Accountability or AC. When creating new closure rules, note the following:

- The scope of traversal is Export.

- The output schema format is TXML.

- If you want a line in the structure to be considered in the accountability check, set the Action Type entry to PROCESS.

If the Show Unconfigured view options are set to show the unconfigured structure and you want to only consider the configured lines for the accountability check (using closure rules mode), adding the following clause to the existing list of closure rule clauses causes the accountability check to skip the unconfigured lines.
Configuring the manufacturing environment

<table>
<thead>
<tr>
<th>Primary Object</th>
<th>Class Type</th>
<th>CLASS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Object</td>
<td>Class Type</td>
<td>BOMLine</td>
</tr>
<tr>
<td>Secondary Object</td>
<td>Class Type</td>
<td>CLASS</td>
</tr>
<tr>
<td>Secondary Object</td>
<td>BOMLine</td>
<td></td>
</tr>
<tr>
<td>Relation Type</td>
<td>PROPERTY</td>
<td></td>
</tr>
<tr>
<td>Related Property</td>
<td>or Object</td>
<td>bl_child_lines</td>
</tr>
<tr>
<td>Action Type</td>
<td>SKIP</td>
<td></td>
</tr>
<tr>
<td>Conditional Clause</td>
<td>SECONDARY.bl_is_occ_configured=&quot;false&quot;</td>
<td></td>
</tr>
</tbody>
</table>

If you create your own inclusion rules for the accountability check, you must add them to the following preferences to have them displayed in the Accountability Check dialog box.

- **MEAccountabilityCheckSourceStructureExpansionFilterRules** (Site)
  List of closure rules that are displayed in the Accountability Check dialog box.

- **MEAccountabilityCheckTargetStructureExpansionFilterRules** (Site)
  List of closure rules that are displayed in the Accountability Check dialog box.

**Configure the accountability check to use net effectivity**

1. Configure unit effectivity to be valid for multiple effectivities as explained in the *Manufacturing Process Planner Guide*.

2. Set the **MEAccountabilityCheckEnableNetEffectivity** preference.
   This controls whether the accountability check mechanism checks for net effectivity.

3. Implement a user exit and register a method for it in the Business Modeler IDE.
   For more information, see *Customizing the net effectivity user exit*.

**Customizing the net effectivity user exit**

Teamcenter provides you with customized code to help you implement net effectivity. The method returns the net effectivity that results when you intersect the effectivity defined on an occurrence (or if there is none, the effectivity of its parent) with the effectivity configured in the revision rule.

You must configure this user exit in the Business Modeler IDE application. When doing so, note that the extension point is **Fnd0BOMLINE_ask_effectivity_by_parent** and the method is **BMF_BOMLINE_compute_net_effectivity**.

For more information about implementing user exits, see the *Business Modeler IDE Guide*. 
Running an accountability check or propagation at a specified time

You can schedule an accountability check or structure propagation to run at a later date or time or at a specific recurring time. This is helpful if you are comparing large structures where the action is time-consuming or if you require comparison results on a regular basis. Teamcenter uses the Dispatcher application to run the comparison and propagation at a time you specify.

To use this feature, your administrator must configure the asynchronous services. For more information, see *Dispatcher Server Translators Reference Guide*.

The asynchronous services require the installation of Teamcenter Dispatcher. For more information, see *Getting Started with Dispatcher (Translation Management)*.

**Note** Because asynchronous tasks run in a separate session, the current rich client session is not updated to reflect any changes that are made. If you choose to execute the task immediately, you do not see all of the results in the current session. Refreshing the window or reloading the structure after the task is complete updates your session to see the changes made by the asynchronous process. This is not an issue for sessions that start after asynchronous processing is complete.

For more information about running the accountability check or propagation at a specified time, see the *Manufacturing Process Planner Guide*.

Configure new categories in the Time view

1. In the Business Modeler IDE, open the Advanced perspective by choosing Window → Open Perspective → Other → Advanced.

2. Open the Activity Category in the LOV folder in the Extensions view.

3. Add a new category, giving it a name and optional description.
4. Save and deploy the project.
   For more information, see the Business Modeler IDE Guide.
   Teamcenter adds a new category to the Time view.

5. (Optional) Change the color of the category in the Options pane.
Configuring an MTM data card

You can add predefined activities based on information available in the Methods Time Measurement (MTM) system to operations in the **Time** view in Manufacturing Process Planner. This system sets industry-standard times required to perform standard manual tasks. The standards include a list of time elements where each element represents a specific and short step a worker can perform as part of the execution of an operation. Typically, the time elements are organized into groups according to types, distance, and difficulty. The standards are based on repeated measurements done on real production lines.

In the **Time** view, you can open a data card and select individual tasks containing a set time. Each of these tasks is added as a separate activity. You can then use these standard times to run time calculations and analyses.

The standard MTM data cards hold times in TMU (time measurement units). Teamcenter stores times in seconds. The **METimeDataCardConversionFactor** preference specifies the factor to convert from data card time units to seconds.

You can also replace the MTM data card with a customized one that includes times required for tasks that are specific to your company. If your customized data card contains times in units other than TMU, you must define the conversion rate between
your data card units to seconds in the `METimeDataCardConversionFactor` preference.

**Note** Please be aware that the proper use of the time data card functionality does require thorough training in MTM methodology. Without such training the user activities may lead to erroneous results and will not represent a true MTM analysis.

For MTM training please contact your local MTM association.

### Install MTM data cards

You must install the data cards using Teamcenter Environment Manager before you can use them in Manufacturing Process Planner. The default data cards are not available until you install them.

For more information about installing the MTM data cards, see the *Installation on Windows Servers Guide*.

### Create your own data card

1. Create a Microsoft Excel spreadsheet containing the data card information.

   Each cell holding the time data must contain a hyperlink with the following format:

   `usercode://code1|time_as_double|description`

   **Where**
   
   | code1 | The time code that appears in the **Code** column in the **Activities** table. |
   | time_as_double | The time required for the activity expressed as a double format. |
   | description | The description of the activity. Teamcenter copies the description and uses it as the name of the activity. |

   **Note** You must use the pipe character (|) as a separator. To prevent illegal characters, you can use the ASCII hexadecimal representation `%7C`. You can use other hexadecimal symbols to replace special symbols in your URL text if you get incorrect results.

2. In Excel, choose **File→Save As...** to save the spreadsheet as a Web page (HTM or HTML extension).

3. Type the path to the HTM or HTML file in the `METimeDataCardPath` preference.

4. Make sure that the `METimeDataCardConversionFactor` preference is correctly set.

   If you enter seconds as the unit in the Excel sheet, set this factor to 1.
5. Restart Teamcenter.

When you open a data card in the Time view, Teamcenter displays a tab for each sheet in the Excel file. When you click an activity cell in the data card, the activity time is displayed in the Activities table of the Time view.

Configuring the TiCon integration

You can access time data from TiCon, an external time management system, and assign it to activities in Manufacturing Process Planner using a Web service. You can search for time elements in TiCon, create new activities, or overwrite existing activities with the timing data of time elements, and update activities assigned to time elements that were changed in TiCon. You can use the existing time management features to calculate time analysis based on the TiCon data.

By default, Teamcenter uses the following internal mapping system to map Teamcenter activity properties to TiCon time element attributes.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<xsi:Configuration xmlns:xsi="http://www.siemens.com/plm/Teamcenter/MSE/TimeFieldMap">
  <Field TeamcenterProperty="al_activity_object_name" TimeSystemAttribute="description"/>
  <Field TeamcenterProperty="al_activity_time_system_category" TimeSystemAttribute="indicator1id">
    <Value TeamcenterPropertyValue="NA" TimeSystemAttributeValue="NA"/>
    <Value TeamcenterPropertyValue="VA" TimeSystemAttributeValue="10000"/>
    <Value TeamcenterPropertyValue="NVA" TimeSystemAttributeValue="10001"/>
  </Field>
</xsi:Configuration>
```

You can use the default mapping as a starting point to create your own mapping file. You can modify the default mapping to map your company’s activity properties to the TiCon time element attributes.

Map attributes of time elements to activity properties

1. Create a mapping file to map activity properties used by your company to TiCon time elements.

   For an example of the mapping syntax, see Configuring the TiCon integration.

2. Create a dataset in Teamcenter and import the mapping file to it.

   Siemens PLM Software recommends you create a **Text** dataset as it is easiest to edit.

   **Note** You must create this dataset within a folder.

3. Add the name of this dataset, its type, and reference type to the **METimeSystemFieldMappingDataset** preference.
Configuring publishing of manufacturing data

Manufacturing Process Management allows you to create publishing pages that can be viewed or printed by nontechnical users with limited computer skills, such as assembly personnel. Publishing pages is a general term for any document used by manufacturing personnel, including work instructions, maintenance instructions, assembly instructions or any other documentation needed for your processes.

To create or view manufacturing documentation, install Microsoft Visio 2003 SP3 or Visio 2007 SP1 on each workstation where you author publishing pages. Both Visio Standard and Visio Professional are supported. You can install Visio before or after the Teamcenter installation.

**Note** If you have installed Teamcenter systems engineering and requirements management, you must use Visio 2007 SP1.

Publishing pages can be published to PDF or HTML so that they can be viewed independently of Teamcenter or Visio.

You create a template for each of the publishing page types needed in your organization from assets (Visio shapes) that are bound to Teamcenter data. The process planners use these templates to create publishing pages that may include text, tables, graphics, 2D or 3D snapshot (product views), and link assets. When all the publishing pages are complete, the process planner can collect them into a technical portfolio that may contain all the activity, operation, and process documentation for a particular assembly line. Users manipulate publishing pages and portfolios in the Manufacturing Process Planner, Part Planner, Resource Manager, Plant Designer, and Multi-Structure Manager applications. They may also add 3D annotations where appropriate. For detailed procedures, see the procedures for viewing, creating, and editing manufacturing documentation in the guides for those applications.

In addition to creating manufacturing documentation in Microsoft Visio, you can create links that point to manufacturing documentation in Microsoft Word and text files.

You can optionally publish manufacturing documentation in PDF format. If you publish a PDF of a portfolio, Teamcenter attaches the PDF dataset to the portfolio. If you create a PDF of a publishing page, Teamcenter attaches the PDF dataset to the page dataset. You must install PDF generation software to use this feature, for example, one of the Adobe products or the freeware GhostScript utility.

**Understanding roles**

Publishing pages have content that reflects a specific purpose or role. By default, Teamcenter supports two roles, **Work Instructions** and **Product Manual**. You can customize Teamcenter to support any desired roles, for example, **Repair Manual**. The same dataset types are used for all roles. When publishing pages are attached to BOM lines or item revisions, the role is determined by the type of relation that attaches the dataset to the parent item revision. Multiple datasets can be attached to an item revision and the datasets can fulfill different roles according to the relation type. When a publishing page is attached to an activity, its role is determined by a preference setting. All publishing pages attached to an activity fulfill the same role. When a portfolio is generated from a structure, the pages that are added to it are determined by the role that the user specifies for that portfolio. Internally, each role is typically associated with:
Chapter 8  Configuring the manufacturing environment

- A relation type used to attach a publishing page dataset to a revision.
- A transfer mode for generating a portfolio for this role.
- One or more alias files for populating assets.

A set of preferences associates the role with the relation type, the transfer mode, and the default alias files described above, as well as the following:
- The list of roles in the portfolio generation dialog box.
- A set of useful alias files.
- The user’s favorite list of templates.

You can create a custom roles to reflect your company’s needs. There are two steps necessary to do this:
- Add a new relation object for the new role using the Business Modeler IDE. For more information, see the Business Modeler IDE Guide.
- Add a new rich client plug-in that extends the com.teamcenter.rac.cme.tcpublishwrapper.publishingPageViewRoles extension point. For more information, see the Client Customization Programmer’s Guide.

For more information, see Creating a customized publish role — example.

Creating a customized publish role — example

The following is an example plugin.xml file for adding a new Teamcenter publish role named Fnd0MyPersonalRole. The Teamcenter publish wrapper extension point requires both a role and a view. The plug-in must define both. The view must be added to a Teamcenter menu. This example shows adding the view as an entry to the Open With menu command where the current Work Instruction menu command exists.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<plugin>
  <extension point="com.teamcenter.rac.vns.viewDef">
    <view alwaysUseSetRootInputObject="true" 
     id="com.myCompany.viewId" 
     selectionInPrimaryEffectsInvisibleSecondary="false" 
     selectionInSecondaryEffectsPrimary="false"> 
      <viewType type="secondary"/>
    </view>
  </extension>
  <extension point="org.eclipse.ui.views"> 
    <view allowMultiple="true" category="com.teamcenter.rac.aifrcp.manufacturingCategory" class="com.teamcenter.rac.cme.tcpublishwrapper.PublishingPageView" 
     icon="icons/workinstructions_view_16.png" 
     id="com.myCompany.viewId" 
     name="%viewId.VIEW" 
    </view>
  </extension>
  <extension point="com.teamcenter.rac.cme.tcpublishwrapper.publishingPageViewRoles"> 
    <viewRoleType tcPublishRole="Fnd0MyPersonalRole" 
     viewId="com.myCompany.viewId"/>
  </extension>
</plugin>
```
Set preferences and environment variables

- **IMAN_MEWorkInstruction_relation_primary**
  Lists the object classes that can be pasted into a work instruction using this relation.

- **AppInterface_TCPublishingTechnicalPortfolio_default_relation**
  Defines the default relation used when a `TCPublishingTechnicalPortfolio` dataset is pasted under an intermediate data capture (IDC) while creating technical portfolios.

- **AppInterface_TCPublishingTechnicalPortfolio_shown_relations**
  Defines the relations available when pasting a `TCPublishingTechnicalPortfolio` dataset under an intermediate data capture (IDC) while creating technical portfolios.

- **IMAN_PublishingPageAssetContext_relation_primary**
  Defines which Teamcenter objects can be used with a `IMAN_PublishingPageAssetContext` relation.

- **IMAN_TCPublishedPortfolio_relation_primary**
  Defines which Teamcenter objects are used as primary objects with a `IMAN_TCPublishedPortfolio` relation.

- **ItemRevision_TCPublishingPage_default_relation**
  Defines the default relation created when a publishing page is pasted within an item revision.

- **TC Publishing Technical Portfolio type_DefaultChildProperties**
  Determines which properties are children of the indicated technical portfolio type. Each type requires its own preference.

- **TC Publishing Page type_DefaultChildProperties**
Chapter 8  Configuring the manufacturing environment

Determines which properties are children of the indicated publishing page type. Each type requires its own preference.

- **IMAN_PorfolioConfiguredPage_relation_primary**
  Defines which Teamcenter objects are used as primary objects with a IMAN_PorfolioConfiguredPage relation.

- **IMAN_PorfolioPage_relation_primary**
  Defines which Teamcenter objects are used as primary objects with a IMAN_PorfolioPage relation.

- **TechnicalPortfolioRoles**
  Use this preference to define the roles of technical portfolios.
  For more information about roles, see *Understanding roles*.

- **TechnicalPortfolioTransferMode_IMAN_MEWorkInstruction**
  Defines the transfer mode of the specified technical portfolio. TransferMode objects control the related data exported into a PLM XML document.

- **TCPublishing_Default_Alias_File**
  The name of the installed alias file, as described later. Defines the default alias file loaded when a new publishing page is viewed.

- **TCPublishing_Aliases.IMAN_MEWorkInstruction**
  The names of the available alias files for work instruction pages. If this preference is not set, the user can choose from any alias file in the database.

- **TCPublishing_Activities_Role**
  Defines the role of all publishing page datasets that are attached to activities in Manufacturing Process Planner, for example, IMAN_MEWorkInstructions (the default setting). The role determines the tab name shown in Manufacturing Process Planner and also how portfolios are generated. If you change the relation name defined in this preference, you must define a new closure rule and transfer mode to generate portfolios. You must also add the new role to the TechnicalPortfolioRoles preference.

- **TCPublishing_Aliases.Activities**
  Contains the names of the alias file datasets that appear when the user wants to load alias files for activity pages.

You should also ensure that the Dataset_DefaultChildProperties preference for each user contains the following settings:

- **TC_PublishingPageAssetContext**
- **TC_PortfolioPage**
- **TC_PortfolioConfiguredPage**

These settings allow users to cut and paste publishing pages and portfolios.

For more information about setting preferences, see the *Preferences and Environment Variables Reference*. 
The following environment variables are set during installation and you should not normally change their values.

For more information about environment variables, see the *Preferences and Environment Variables Reference.*

- **PUBLISH_PDF_TIMEOUT**
  
  This environment variable sets a time-out period for stalled PDF conversions. Enter a number of milliseconds after which the conversion is terminated, enter **120000** for 2 minutes. If you require an infinite time-out, set this environment variable to **INFINITE**.

- **PUBLISH_PDF_MAX_JOBS**
  
  Defines the maximum number of outstanding PDF conversion jobs that are allowed to run simultaneously. This environment variable is effective only when you publish a portfolio to PDF and select multiple PDFs.

- **PUBLISH_PDF_INTERVAL**
  
  This environment variable defines the number of milliseconds Teamcenter waits before issuing each subsequent PDF conversion job. Set a value that ensures there are no conflicts between PDF jobs starting simultaneously. Typically, you do not need to change the default value of **1000** (one second). This environment variable is effective only when you publish a portfolio to PDF and select multiple PDFs.

### Installing the transfer mode

When you install Teamcenter, the necessary default PLM XML transfer modes are provided. The transfer mode for publishing is **defaultTransferModes.xml**, which is located in the **TC_DATA** data directory of the base installation (**TC_BASE**).

If necessary, you can reinstall this transfer mode by running the following command from a Teamcenter shell:

```
$TC_BIN\plmxml import -u=infodba -p=infodba -g=dba -xml_file=
$TC_BIN\defaultTransfermodes.xml
```

Ensure you enter the complete path for the **xml_file** argument, for example:

```
file=C:\Tc2007\tcdata\mfg\defaultTransfermodes.xml
```

You would only reinstall the file if you make system-specific changes, for example, you customize the mode for your site environment.

### Defining alias files

To be able to see Teamcenter-managed data in a publishing page, you have to create an asset, bind it, and populate it. When you bind the asset, the **Bind** dialog box contains a list of transfer modes and a number of choices, called aliases, from which you can select. The choice you make for each asset defines how Publish finds the data that is displayed in the asset when you populate it.

When you select an object in Teamcenter, and then populate an asset, a number of things happen.
Chapter 8  Configuring the manufacturing environment

1. If the selected object is a BOM line, Publish creates a structure context object in the database to persist the selection.

2. Publish creates a PLM XML file that contains information about the selected persistent object and about related objects. This additional content about the related objects is determined by the definition of the transfer mode that was assigned to the asset in the Bind dialog box.

3. Publish searches through the PLM XML file’s content to find the exact data to expose in the asset. To do this, it retrieves the alias definition for the alias that was bound to the asset. The definition contains a list of paths. Publish attempts to follow each path from the root object in the PLM XML file until it finds some valid data. This data can consist of anything from a string, such as a form attribute value, to a file containing an image.

4. Teamcenter creates a snapshot of the data in the asset on the Visio pane.

5. Teamcenter stores the ID of the selected persistent object in the Visio file, associating it with the asset, for later retrieval during an update operation.

The transfer modes and aliases that are available in the Bind dialog box are all defined in the alias file. Each transfer mode functions as a recipe for what is included in the PLM XML file. The definition of each alias functions as a recipe that tells Teamcenter which parts of the exported PLM XML information to expose in an asset.

The snapshot of Teamcenter’s data is visible whenever you open the page file for viewing. You can also update an asset so that it displays changes that were made in Teamcenter since the asset was first populated. When the asset is updated, new PLM XML is generated from the same selected object using the same transfer mode, and the same alias is then used to find the exact data from that PLM XML to be displayed in the asset.

A sample set of alias definitions is provided with the default installation, but you should create alias definitions and transfer modes to support other situations. To do this, you should:

- Ensure that the information appears in the PLM XML file. If it does not, create or modify a transfer mode so that the information is visible.

- Ensure that an alias definition correctly extracts this information from the PLM XML file. If not, modify existing aliases or create new aliases. The default alias definition file contains instructions and examples of aliases.

Teamcenter includes a default alias file (TcPublishSystemAlias.xml) that you can use as the basis for your own mapping files and its location is defined in the TCPublishing_Default_Alias_File preference. The comments within this file contain detailed notes on the structure and usage of the alias file. Read these before beginning with customization.

Caution  Do not edit this file directly. Create your own alias dataset as explained in Create a new publishing alias dataset and work in this file.
Building a sample alias file

The following figure shows a publishing page with a single table asset that has been populated with the **Name** property of child items and the **Description** property of an attached form from each child item.

Example result – table asset populated with child item names and form descriptions

This example starts with a simple product structure containing a parent item with two direct child items. Each child item has an attached form. This scenario is shown in the following figure.

Sample product structure

To populate the table asset with data, you must assign a binding to the table asset. The binding used here is **ProductPartList & Name** and **ProductPartList & FormDescription**, as shown in the following figure.
Publish Bind dialog box

You can find the sample alias file that was used to create this example (ProjectTePublishSystemAlias.xml), as well as the PLM XML code generated by the selected transfer mode (tcn_export.xml) in the Sample directory of your installation. This must be installed separately. If you do not have a Sample directory, contact your Teamcenter administrator.

In the example, the second binding involves the Description property of a form attached to the child item. The default alias file does not provide a definition for attaching the Description field of a form to a child item so you must modify this alias file.

In the following figure, notice the new FormDescription alias entry.
Configuring the manufacturing environment

Expanded ProductPartList alias showing FormDescription entry

This new entry is created by adding a new top level alias definition into the alias file. The following code shows excerpts from the **DefaultTcPublishSystemAlias.xml** file. The line shown in bold print results in the new entry in the Bind dialog box shown in the figure. The statements in italics refer to other alias definitions.

```
<!---- Product part list information addresses the first level parts list of an occurrence it will return only the child occurrences of the first level occurrence selected. ---->
<Alias name="ProductPartList" assetType="8" toolTip="product data"
     qualifier="2" accumulate="Quantity">
 &OCC-ATTR-DEF;  
</Alias>
</!

<!-- properties that you want to see on an occurrence. Entity definition is used to avoid lot of typing-->
<!ENTITY OCC-ATTR-DEF "
  <Alias name='ID' default='No ID'/>
  <Alias name='Description' default='No description'/>
  <Alias name='Name' default='no name'/>
  <Alias name='Rev' default='A'/>
  <Alias name='Transform' default='Identity'/>
  <Alias name='LogicalID' default='Logical Identity'/>
  <Alias name='SequenceNumber' default='No find#'/>
  <Alias name='ParentName' default='No parent'/>
  <Alias name='Type' default='No type'/>
  &FORM-MASTER-ATTR-DEF;
  &ICO-ATTR-DEF;
">
```

```
<!ENTITY FORM-MASTER-ATTR-DEF "
  <Alias name='project_id' default='cannot get project_id'/>
  <Alias name='previous_item_id' default='cannot get previous itemid'/>
  <Alias name='item_comment' default='cannot get comment'/>
  <Alias name='serial_number' default='cannot get serial_number'/>
  <Alias name='UserData1' default='cannot get user_data_1'/>
  <Alias name='UserData2' default='cannot get user_data_2'/>
  <Alias name='UserData3' default='cannot get user_data_3'/>
  <Alias name='rev_project_id' default='cannot get project_id'/>
  <Alias name='rev_previous_item_id' default='cannot get previous itemid'/>
  <Alias name='rev_item_comment' default='cannot get comment'/>
">
```
Chapter 8 Configuring the manufacturing environment

```xml
<Alias name='rev_serial_number' default='cannot get serial number'/>
<Alias name='revUserData1' default='cannot get user_data_1'/>
<Alias name='revUserData2' default='cannot get user_data_2'/>
<Alias name='revUserData3' default='cannot get user_data_3'/>
<Alias name='FormDescription' default='No description'/>
```

For the new alias definition to traverse the PLM XML file, another edit is necessary. The following entry shows the traversal performed by the new alias definition. This entry is added to the `<!ENTITY FORM-MASTER-ATTR >` section of the alias file.

```xml
<Alias name='FormDescription'>
  <Path>:AssociatedAttachment[@role='IMAN_specification']:@associatedAttachmentRefs</Path>
  <Path>:Form:attachmentRef</Path>
  <Xpath>description</Xpath>
</Alias>
```

To determine what to add and how the traversal works, reference the PLM XML file generated by the example and found in the Samples directory.

In the binding, ProductPartsList & FormDescription was selected. Find the Alias and AliasAlternate sections for ProductPartsList in the alias file. The following code section shows the relevant lines from the generated PLM XML file. All PLM XML files have a traverseRootRef entry like this:

```xml
<Header id='id1' traverseRootRefs='#id3' transferContext='tcm_export'></Header>
```

`#id3` is the start node for the traversal. The transferContext transfer mode is used to generate this PLM XML code.

The start node is AssociatedAttachment.

Look through the alias file in the ProductPartsList section for all alias definitions that have a primary object specifying an AssociatedAttachment node. The following code contains the first one:

```xml
< AliasAlternate name='ProductPartList'>
  <Path>AssociatedAttachment:AssociatedAttachment[@role='contents']:@childRefs</Path>
  <Path>AssociatedAttachment::*:attachmentRef</Path>
  <Path depth='1..1'>*:occurrenceRefs</Path>
  &OCC-ATTR;
  &FORM-MASTER-ATTR;
  &ICO-ATTR;
</AliasAlternate>
```

The traversal tries each alias rule that has a primary object matching the start node and traverses through each path statement until it gets to the Xpath statements. If any Path statement fails during the traversal, the traversal tries the next matching alias statement. The first Path statement is:

```xml
<Path>AssociatedAttachment:AssociatedAttachment[@role='contents']:@childRefs</Path>
```

This statement causes the traversal to proceed to AssociatedAttachment nodes with role=contents via the childRefs attribute in the PLM XML file. The PLM XML statement childRef attribute is:

```xml
childRefs='#id4 #id72'
```

Therefore, the traversal moves to PLM XML nodes with id4 and id72, if they match the role=contents condition. The PLM XML node containing id4 is:

```xml
<AssociatedAttachment id='id4' attachmentRef='#id9' role='contents'>
```

The next Path statement in the alias definition is:

```xml
<Path>AssociatedAttachment::attachmentRef</Path>
```

This Path statement indicates that the system expected AssociatedAttachment nodes as results from the last statement. If it did not, this alias definition would fail at this point.
The current **Path** statement tells the system to use the **attachmentRef** and return whatever is there. The * symbol is a wildcard indicating that any node is sufficient. Consequently, the traversal proceeds to the node containing **#id9**.

```xml
<Occurrence id="id9" instanceRef="#id43" associatedAttachmentRefs="#id45 #id46 #id49 #id67" occurrenceRefs="#id12 #id32">
  <ApplicationRef application="Teamcenter" label="SIEhslreBrSpAA/ScFh8BEQBrSpAA"></ApplicationRef>
  <UserData id="id10" type="AttributesInContext">
    <UserValue value="" title="Quantity"></UserValue>
    <UserValue value="" title="OccurrenceName"></UserValue>
    <UserValue value="" title="SequenceNumber"></UserValue>
    <UserValue value="" title="ID"></UserValue>
  </UserData>
  <Transform id="id8">1 0 0 0 0 1 0 0 0 1 0 0 0 0 0 1</Transform></Occurrence>
```

Node 9 is an occurrence.

The next **Path** statement is:

```xml
<Path depth="1..1">*:0@occurrenceRefs</Path>
```

This **Path** statement tells the system to go to the nodes referenced by the **occurrenceRefs** attribute. Again, the * wildcard is used, so the traversal picks up whatever node is contained in the PLM XML. The first node ID is **#12**.

```xml
<Occurrence id="id12" instanceRef="#id14" associatedAttachmentRefs="#id19 #id22 #id25 #id30" parentRef="#id9"></Occurrence>
```

This node is also an occurrence. The first line contains all the relevant reference information.

The next statement in the alias definition is:

```
&OCC-ATTR;
```

This is a macro statement that implies that the traversal expects to be on an occurrence node at this point and is getting ready to traverse its attributes.

At this point, the traversal again encounters alias names as shown for **OCC-ATTR**, **ID**, **Description**, **Name**, and so on.

Remember that the binding specified was **ProductPartsList & FormDescription**. The traversal started at the **ProductPartsList** node and arrived at this point. Now, it must find an alias name matching **FormDescription** to find any additional XML traversal rules to use. Going through the **OCC-ATTR** statement, the traversal cannot find any alias names that match **FormDescription** because the edit was not added to that entry. Therefore, the traversal continues on to the next entry in the original alias definition.

```
<ENTITY OCC-ATTR "">
  <Alias name="/ID/">
    <Path>*:0@instanceRef</Path>
    <Path>*:0@masterRef</Path>
    <Xpath>@catalogueId(@productId)/Xpath>
  </Alias>
  <Alias name="/Description/">
    <Path>*:0@instanceRef</Path>
    <Xpath>@Description/Path>
  </Alias>
  <Alias name="/Name/">
    <Path>*:0@instanceRef</Path>
    <Xpath>@name/Path>
  </Alias>
</ENTITY>
```

The next alias rule is:

```
&FORM-MASTER-ATTR;
```

The traversal looks in the alias file for this macro and starts traversing through the macro looking for an alias named **FormDescription**. The traversal finds the newly added alias as follows:

```xml
<Alias name="FormDescription"/>
```
Chapter 8  
Configuring the manufacturing environment

Remember that the last node collected was:

```xml
<Occurrence id="id12" instancedRef="#id14" associatedAttachmentRefs="#id19 #id22 #id25 #id30" parentRef="#id9"/>
```

Now the first Path statement in the current alias rule is this:

```xml
<Path>*:AssociatedAttachment[@role='IMAN_specification']:@associatedAttachmentRefs</Path>
```

This means the traversal is looking for AssociatedAttachment nodes with an IMAN_specification role. This should be available through the AssociatedAttachmentRefs statement. In this case, the only node that matches these criteria is the one with #id22 shown here.

```xml
<AssociatedAttachment id="id22" attachmentRef="#id23" role="IMAN_specification"/>
```

The next alias rule is:

```xml
<Path>*:Form@attachmentRef</Path>
```

This means that using the attachmentRef attribute, the traversal expects to move to a form. #id23 indicates the node for which the traversal needs to search.

```xml
<Form id="id23" name="ATestForm" accessRefs="#id15" subtype="OfficeDocForm" subClass="OfficeDocForm">
 <Description>ChildItem 1 Description</Description>
 <ApplicationRef version="SYNhstU9BrSpAA" application="Teamcenter">
  <label>SYNhstU9BrSpAA</label>
 </ApplicationRef>
 <UserData id="id24" type="FormAttributes">
 <UserValue id="" title="Title"/>
 <UserValue id="" title="Subject"/>
 <UserValue id="" title="Author"/>
 <UserValue id="" title="Manager"/>
 <UserValue id="" title="Company"/>
 <UserValue id="" title="Category"/>
 <UserValue id="" title="Keywords"/>
 <UserValue id="" title="Comments"/>
 <UserValue id="" title="Client"/>
 <UserValue id="" title="Project"/>
 <UserValue id="" title="Purpose"/>
 </UserData></Form>
```

This shows that #id23 is a form.

At this point, the traversal has successfully executed all the Path statements and ends with a final destination node. The next statement in the alias rule is:

```xml
<Xpath>Description</Xpath>
```

The Xpath statements describe what information to take from the resulting node to use in the publishing page. In the example, the property originally sought was the Description property of the form attached to the ChildItem item. This Xpath statement says to return the data for the Description entry for the current node. Looking at the PLM XML, this is:

```xml
<Description>ChildItem 1 Description</Description>
```

Looking at the populated publishing page in the figure entitled Example result – table asset populated with child item names and form descriptions, you see the same description in the FormDescription column for ChildItem 1. This accomplishes the goal of the example.

The Path statements are used to traverse from the start node to the final node. The Xpath statements are used to extract information from the final node. There are potentially several valid paths from the start node to the final node in the PLM XML. You need to determine any valid set of path statements to reach the final node.
Specifying end objects

To specify which object is the end object that is exposed in an asset, you must mark it in a Path element in the alias definition. The following example shows an existing alias alternate for populating a 2D asset:

```xml
<AliasAlternate name="2DGraphic">
  <Path>AssociatedAttachment:AssociatedAttachment[@role='contents']:@childRefs</Path>
  <Path>ConfiguredOccurrenceGroup:*:@childRefs</Path>
  <Path>AssociatedAttachment:ConfiguredOccurrenceGroup:AttachmentRef</Path>
  <Path>DataSet:ExternalFile[@format='jpg' or @format='jpeg' or @format='gif' or @format='bmp']:@memberRefs</Path>
</AliasAlternate>
```

To identify the end object, use the createRelation attribute on the last Path element:

```xml
<Path createRelation="VisAssetPopulatedByEndobj"/>
```

This causes the end object UID to be sent back through Lifecycle Visualization to the Teamcenter rich client, along with the createRelation identifier. The rich client then creates the VisAssetPopulatedByEndobj relation from the publishing page dataset to the end object.

To create two different relationships to the same end object, use this syntax:

```xml
<Path createRelation='VisAssetPopulatedByEndobj,AnotherVisRelationship'/>
```

When customizing alias definitions, you can identify the end object for any asset and Teamcenter creates the relation specified. You can define your own custom relations and have them created too by adding appropriate createRelation clauses or by adding additional relation names to existing createRelation clauses.

Navigating through a PLM XML file

One approach to determine the path statements needed for your alias definition is to find the final node in the PLM XML file. Search the file for references to its ID. From this current node, find the node or nodes that reference it. Continue this backward traversal making sure you can get to the start node. Remember the start node is the node referenced by the traverseRootRef attribute. After you find a path using the backward traversal from the final node to the start node, go back and record the type of each node in the path. Use this information to build the path statements needed for an alias to your final node. Additionally, look at existing alias path definitions for examples of how to traverse from one type of node to another. A backward traversal using our example in pseudo XML would look something like this:

1. The ChildItem 1 Description node, the data you want to expose in the asset, is in the Form id=id23 node.
2. The Id23 node is referenced by AssociatedAttachment id=id22 as an attachmentRef type.
3. The Id22 node is referenced by the Occurrence id=id12 statement as an associatedAttachmentRef type.
4. The Id12 node is referenced by the Occurrence id=id9 statement as an occurrenceRef type.
5. The Id9 node is referenced by the AssociatedAttachment id=id4 statement as an AttachmentRef type.
6. The Id4 node is referenced by the AssociatedAttachment id=id3 statement as a childRef type.

7. The Id3 node is the traverseRootRef node from which traversal always starts.

There is another reference to id22, but if you follow it back you see a pseudo folder. Pseudo folders are generally not useful in a path traversal and can be ignored. After a while, it is useful to recognize what entities in the rich client relate to which nodes in the PLM XML file. Through some experimentation, these relationships become clearer. If you compare this pseudo XML code with the actual Path XML statements traversed in the example, you can see how they relate to each other. Therefore, when you need to update an alias file, if you can first determine a set of pseudo XML statements, it is easier to write the Path and XPath statements needed to edit the alias file.

**Strategies for working with alias files**

The following are a few high-level tips and strategies about working with the alias file.

- Define a use case to work from.
  1. Determine what will be selected and what you want to see in what type of asset.
  2. When creating a template that will be used repeatedly, it is easier if the user can select a single object with which to populate all of the data in the work instruction.
  3. Find an example of the data in the database that you can select, for example, a BOM. If possible, make it unique so you can find it easily in the PLM XML file.
  4. Determine whether to use an existing alias or to add a new one to the Bind dialog box.
  5. Determine what alias file to use. You can develop an alias file for this user or this use case only that is simple to use and contains only one transfer mode, or you can add to an existing alias file.

- Identify a suitable transfer mode.
  1. Generate a PLM XML file by populating an asset with the transfer mode that you think will work.
  2. Examine the resulting file and make sure it has the data that you want to expose in the asset. If it does not, you must find a better transfer mode, or you must modify the transfer mode’s closure rules to include your data.
  3. In some cases, it makes sense to create a custom transfer mode whose closure rules are optimized to efficiently generate XML that supports only the assets in a single template.

- Examine the PLM XML file and identify the nodes that provide a path from the root ref to the data in which you are interested.
For more information, see *Navigating through a PLM XML file.*

- Define an alias alternate that follows the node path. To save time, test the alias definition using the stand-alone test utility on the generated PLM XML file. The utility is described in the alias file header.

- Decide what other use cases need to be supported by your alias definition.
  1. The data may be attached to the object in more than one way.
  2. The user may start with a different populating object.

- Create a new use case and repeat this process.

**Create a new publishing alias dataset**

To create a new publishing alias from an existing alias file, do the following in My Teamcenter:

1. Use the Search button to find the Installed Alias dataset.

2. Export the TcPublishSystemAlias.xml file to your local disk using the Named References dialog box.

3. Rename the local XML file to something meaningful, such as AliasTutorial.xml.


   Teamcenter displays the New Dataset dialog box.

5. In the list of dataset types, select TCPublishingAliases, and then type a name and description for the new dataset.

6. Click the Import button, browse to the renamed alias file and click OK.

   Teamcenter creates the new publishing alias.

7. Load the XML file into a text editor and read the notes at the top of the file.

   **Note** Each time you create an alias dataset, you must load the alias file, as described in *Load or reload the alias file.*

   You can optionally include packing of tables in the alias definition. If you choose to pack tables, lines that are identical except for quantity are collapsed into a single line. Two aliases with collapsed tables are provided as templates, CollapsedProductPartList and OperationConsumedParts.

**Update a publishing alias**

To update an existing publishing alias from a modified alias file, do the following in My Teamcenter:

1. Right-click the dataset and choose Named References.

2. Choose Edit→Cut to remove the existing named reference for this alias.

3. Click the Import button, browse to the modified alias file and click OK or Apply.

   Teamcenter updates the publishing alias.
Chapter 8  Configuring the manufacturing environment

Note Each time you modify an alias dataset, you must reload the alias file, as described in Load or reload the alias file.

Load or reload the alias file

1. In My Teamcenter, create a new TCPublishingPage or load one of the publishing page templates.

2. Select the viewer pane and verify that the publishing page is loaded.

3. In your embedded Visio application, choose File→Load Alias.

   Teamcenter displays the Load Alias File dialog box, with available aliases listed in the left-hand pane and loaded aliases listed in the right-hand pane.

   To load a newly created alias dataset that is not already listed in the Available Aliases pane, click the Add File by Name button and browse to the newly created alias dataset.

4. To load a new alias file, select it in the Available Aliases pane and click the + button. To remove (unload) an existing alias, select it in the Loaded Aliases pane and click the – button.

   When the aliases are listed correctly, click OK or Apply to load and unload alias files as necessary.

   If you update an existing alias dataset by replacing its named reference, reload it by selecting the Reload aliases check box and clicking OK or Apply.

Create or edit publishing pages

1. Open a configured structure in the Manufacturing Process Planner, Part Planner, Plant Designer, or Multi-Structure Manager application and select an occurrence.

2. Show the data tabs and select the tab for the required type of publishing page. You can configure Teamcenter to show each publishing context in a separate tab. For example, a user clicks the Work Instructions tab and the tab shows all the publishing pages associated with the occurrence by an M EWorkInstructions relationship. If none exist, an empty page is displayed.

   You can create publishing pages for activities only in Manufacturing Process Planner.

3. Browse though the existing publishing pages or create a new publishing page from a template defined as described previously.

   When editing an existing publishing page, the user can add, remove or edit Visio shapes and associate them with Teamcenter assets. An asset is typically 3D geometry, a 2D or 3D snapshot, a 2D image, or a link to a Microsoft Word file or a text file.

4. Save the new or changed publishing page to associate it with the selected occurrence.
Creating manufacturing documentation in batch mode

You can create and update Teamcenter work instructions and portfolios in batch mode, processing multiple documents using the Teamcenter Publish Batch dialog box. You do not need to process individual documents manually. You can schedule these tasks to take place immediately or at a later date.

Batch processing uses the Dispatcher (formerly known as Translation Management) infrastructure for scheduling create and update features. Additionally, Dispatcher has built-in status, logging, and error reporting features. You can use the Request Administrator Console dialog box to obtain status information about the submitted task. Find this in the Translation menu of My Teamcenter after you install Dispatcher.

You must set up the batch processing in the Manufacturing Process Planner or Multi-Structure Manager applications. You can use My Teamcenter to modify the scheduling of the batch process.

For more information about setting up the batch process, see the Manufacturing Process Planner Guide.

For more information about modifying the scheduling, see the My Teamcenter Guide.

Enabling batch processing of publishing pages

Teamcenter batch processing of publishing pages uses Dispatcher (formerly known as Translation Management) infrastructure for scheduling create and update features. You must install the PublishBatch translator to take advantage of this feature. This enables the PublishBatch service.

There are several prerequisites for running the batch feature.

- The translator is installed on a Windows machine.

- VisAutomationApp, Microsoft Visio, FMS file client cache (FCC), and the Teamcenter service-oriented architecture (SOA) client are installed on the translator machine. VisAutomationApp is installed automatically when you install Lifecycle Visualization.

  For more information about installing Lifecycle Visualization, see the Teamcenter lifecycle visualization Install Guide.

  For more information about installing Microsoft Visio, see the Microsoft Visio Online Help.

  For more information about installing FCC, see the Installation on Windows Clients Guide.

  For more information about installing a Teamcenter SOA client, see the Services Guide.

- The startPublishBatch.bat file is installed in a directory that is specified during installation of the PublishBatch translator. Each setting in this file must be modified as specified in the file.

- The transmodule.properties file in the module/conf directory has a MaximumTasks setting that defaults to 3. This defines the maximum number of instances of publish batch tasks that can run simultaneously. Do not increase this value above 3.
• The clients and Dispatcher Server should point to the common staging directory, and both should have read/write permissions to that directory.

For more information, see the *Dispatcher Server Installation Guide*.

**Troubleshooting PublishBatch translator**

The `startPublishBatch.bat` file is installed in a directory that is specified during installation of the *PublishBatch* translator. As described in the file, if you add `-log` to the `%TRANS_PATH%\tcpublish_batch` line, Teamcenter adds detailed progress and error entries to the task’s log file. Teamcenter stores the log files in the directory you specify during installation.
Configuring the user interface

You can customize the BOM line quantity and configure incremental change in the rich client user interface.

Configure incremental changes

If you plan to use incremental changes to control the manufacturing process, you must complete the following steps:

1. By default, incremental changes are not enabled. To enable incremental changes, set the Incremental_Change_Management preference to true.

   For more information, see the Preferences and Environment Variables Reference.

2. Configure the change types to use with incremental change with the Change Viewer application, as described in the Change Manager Guide.

3. By default, all incremental changes (configured and unconfigured) are visible to the user. To hide unconfigured changes, set the ShowUnconfiguredByChangeEffectivity preference to true.

   For more information, see the Preferences and Environment Variables Reference.

4. Define a release status that is attached to an incremental change when the user first creates it. For example, the following setting creates each incremental change with a Pending status.

   Incremental_Change_ReleaseStatus=Pending

   The status must already exist and have an Access Manager rule that allows write access to objects with this status.

5. Ensure an In IC Context rule is available in Access Manager, allowing a user write access to a released structure.

Customizing the BOM line quantity

Certain sites add dimensions at the end of each part name, resulting in a long BOM line name. These dimensions at the end of the BOM Line column and the value in the Quantity column may run into each other and confuse the user.

To avoid this, you can use the BOM Line Title column instead of the BOM Line column. Add the BOM Line Title column as described in the Structure Manager Guide, move it to the far left, and then removing the BOM Line column from the display.

You should then choose Tools—Options in Structure Manager and enter a custom title in the BOM Line Title format for Items of type: Item box. For example, the following entry shows only the item ID and name separated by two dashes, but no quantity value:

   $bl_item_item_id -- $bl_item_object_name
Chapter 8  Configuring the manufacturing environment

Note  As the Options dialog box lists only actual item types in Structure Manager, you cannot customize process or manufacturing item types directly in this window. To do this, you must first choose Options→Preferences and create a new preference for each applicable item type, for example, BOMLine_Title_Format_MEOP or BOMLine_Title_Format_MEProcess. You can then define the current value of the preference name in the format shown in the example.

Customizing line balancing reports

Line balancing is a method to improve the throughput of assembly lines while reducing manpower requirements and costs. Line balancing addresses the problem of assigning operations to work stations along an assembly line in such a way that the assignment be optimized in some sense. You perform line balancing in Manufacturing Process Planner.

During the line balancing planning activities, you can create a variety of reports.

For more information about creating the reports, see the Manufacturing Process Planner Guide.

Line balancing uses the Teamcenter report framework to generate reports. The reports are created in the Microsoft Excel XML format and saved in datasets. By default, the dataset is attached to the scope of the view from which the report is launched. There is a defined default report definition and style sheet associated with each report type. These are defined in preferences.

<table>
<thead>
<tr>
<th>Report</th>
<th>Preference for report definition ID</th>
<th>Preference for style sheet name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints Consistency Check</td>
<td>MELBConstraintsConsistencyReportID</td>
<td>MELBConstraintsConsistencyStylesheet</td>
</tr>
<tr>
<td>Constraints Violation Check</td>
<td>MELBConstraintsViolationReportID</td>
<td>MELBConstraintsViolationStylesheet</td>
</tr>
<tr>
<td>Balancing Overview</td>
<td>MELBBalancingOverviewReportID</td>
<td>MELBBalancingOverviewStylesheet</td>
</tr>
<tr>
<td>Line Detail</td>
<td>MELBLLineDetailReportID</td>
<td>MELBLLineDetailStylesheet</td>
</tr>
</tbody>
</table>

You can create a new report definition in the Report Builder application and modify the report preference to point to that new definition. You can modify the report definition to change parameters or add style sheets.

Customize the contents of the reports

You can customize the following in a report definition:

- Hide a section of the report.
- Hide a column of the report.
• Add a column to show a BOMLine property.

1. Open the relevant report definition in Report Builder.

2. In the Parameters section, modify the parameters using the following parameters.

<table>
<thead>
<tr>
<th>Name</th>
<th>Meaning</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIDE</td>
<td>Specifies a section of the default report that is not displayed.</td>
<td>HIDE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CONSTRAINTS_DEFINITION</td>
</tr>
<tr>
<td>HIDE_COLUMN</td>
<td>Specifies a column of the default report that is not displayed.</td>
<td>HIDE PROCESS_AREA_INFO</td>
</tr>
<tr>
<td>SHOW_COLUMN</td>
<td>Specifies a BOM line property that is added as an additional column.</td>
<td>SHOW_COLUMN</td>
</tr>
<tr>
<td></td>
<td></td>
<td>bl_resequence_no</td>
</tr>
</tbody>
</table>

For more information about what you can hide in a report, see Line balancing report parameters.

**Line balancing report parameters**

**Constraints Consistency Check Report**

<table>
<thead>
<tr>
<th>Constraints Consistency Check Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date: 13.01.2011</td>
</tr>
<tr>
<td>Time: 12:35</td>
</tr>
<tr>
<td>Top Level Item: 000001/A:1-product BOP Root</td>
</tr>
<tr>
<td>Revision Rule: Latest Working</td>
</tr>
<tr>
<td>Variant Rule: ProductOption= Small;</td>
</tr>
<tr>
<td>Check Result: failed</td>
</tr>
</tbody>
</table>

**Check Details**

<table>
<thead>
<tr>
<th>The objects are in a loop</th>
<th>Involved Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Op1, Op2, Op3</td>
<td></td>
</tr>
</tbody>
</table>

**Constraints Definition**

<table>
<thead>
<tr>
<th>Object</th>
<th>Predecessor</th>
<th>Group</th>
<th>Simultaneous</th>
<th>Incompatible</th>
</tr>
</thead>
<tbody>
<tr>
<td>Op2</td>
<td>Op1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Op3</td>
<td>Op2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Enter this**

HIDE

**CONTRAINTS_DEFINITION**

To

Hide the **Constraints Definition** section of the report
## Constraints Violation Check Report

<table>
<thead>
<tr>
<th>Constraints Violation Check Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
</tr>
<tr>
<td>Time</td>
</tr>
<tr>
<td>Product BOP Top Level Item</td>
</tr>
<tr>
<td>Revision Rule</td>
</tr>
<tr>
<td>Variant Rule</td>
</tr>
<tr>
<td>Plant BOP Top Level Item</td>
</tr>
<tr>
<td>Revision Rule</td>
</tr>
<tr>
<td>Variant Rule</td>
</tr>
</tbody>
</table>

### Check Details

<table>
<thead>
<tr>
<th>Description</th>
<th>Involved Objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>The assignment causes a precedence violation</td>
<td>Op1, Op2, Station1, Station2</td>
</tr>
<tr>
<td>The assignment causes a precedence violation</td>
<td>Op22, Op37, Station10, Station12</td>
</tr>
</tbody>
</table>

### Constraints Definition

<table>
<thead>
<tr>
<th>Object</th>
<th>Predecessor</th>
<th>Group</th>
<th>Simultaneous</th>
<th>Incompatible</th>
</tr>
</thead>
<tbody>
<tr>
<td>Op1</td>
<td>Op5</td>
<td>Op8</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Op9</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Op2</td>
<td>Op1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Op3</td>
<td>Op2</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Process Area Information

<table>
<thead>
<tr>
<th>Process Area</th>
<th>Predecessor</th>
<th>Allocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>station 1</td>
<td>station 2</td>
<td>op 2</td>
</tr>
<tr>
<td>station 2</td>
<td>station 10</td>
<td>Op1</td>
</tr>
<tr>
<td>station 10</td>
<td>station 12</td>
<td>Op 9</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>station 12</td>
<td>station 41</td>
<td>Op8</td>
</tr>
</tbody>
</table>

### Enter this

**HIDE CONSTRAINTS_DEFINITION**

Hide the **Constraints Definition** section of the report.

**HIDE PROCESS_AREA_INFO**

Hide the **Process Area** section of the report.
### Balancing Overview Report and Balancing Line Detail Report

#### Balancing Overview Report

<table>
<thead>
<tr>
<th>Date</th>
<th>13.01.2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>12:15</td>
</tr>
<tr>
<td>Product BOP Top Level</td>
<td>000001/A1-product BOP Root</td>
</tr>
<tr>
<td>Scope</td>
<td>000030/A1-Line 2</td>
</tr>
<tr>
<td>Revision Rule</td>
<td>Latest Working</td>
</tr>
<tr>
<td>Variant Rule</td>
<td>ProductOption= Small;</td>
</tr>
<tr>
<td>Plant BOP Top Level</td>
<td>000002/A1-plant BOP Root</td>
</tr>
<tr>
<td>Revision Rule</td>
<td>Latest Working</td>
</tr>
<tr>
<td>Variant Rule</td>
<td>ProductOption= Small;</td>
</tr>
</tbody>
</table>

**Summary**

<table>
<thead>
<tr>
<th>Cycle Time</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of stations</td>
<td>10</td>
</tr>
<tr>
<td>Time Unit</td>
<td>Second</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Capacity</th>
<th>Work Content</th>
<th>Allocated Time</th>
<th>Remaining Time</th>
<th>Number of Workers</th>
<th>Utilization</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Station 1</td>
<td>300</td>
<td>285</td>
<td>15</td>
<td>3</td>
<td>216</td>
<td>76%</td>
<td>NVABR</td>
</tr>
<tr>
<td>Station 2</td>
<td>200</td>
<td>200</td>
<td>12</td>
<td>2</td>
<td>120</td>
<td>64%</td>
<td>NVABR</td>
</tr>
</tbody>
</table>

#### Balancing Line Detail Report

<table>
<thead>
<tr>
<th>Date</th>
<th>13.01.2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>12:15</td>
</tr>
<tr>
<td>Product BOP Top Level</td>
<td>000001/A1-product BOP Root</td>
</tr>
<tr>
<td>Scope</td>
<td>000030/A1-Line 2</td>
</tr>
<tr>
<td>Revision Rule</td>
<td>Latest Working</td>
</tr>
<tr>
<td>Variant Rule</td>
<td>ProductOption= Small;</td>
</tr>
<tr>
<td>Plant BOP Top Level</td>
<td>000002/A1-plant BOP Root</td>
</tr>
<tr>
<td>Revision Rule</td>
<td>Latest Working</td>
</tr>
<tr>
<td>Variant Rule</td>
<td>ProductOption= Small;</td>
</tr>
</tbody>
</table>

**Summary**

<table>
<thead>
<tr>
<th>Cycle Time</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of stations</td>
<td>10</td>
</tr>
<tr>
<td>Time Unit</td>
<td>Second</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Capacity</th>
<th>Work Content</th>
<th>Allocated Time</th>
<th>Remaining Time</th>
<th>Number of Workers</th>
<th>Utilization</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Station 1</td>
<td>300</td>
<td>285</td>
<td>15</td>
<td>3</td>
<td>216</td>
<td>76%</td>
<td>NVABR</td>
</tr>
<tr>
<td>Process Area 1</td>
<td>200</td>
<td>187</td>
<td>13</td>
<td>2</td>
<td>131</td>
<td>82%</td>
<td>36</td>
</tr>
<tr>
<td>op1</td>
<td>98</td>
<td>98</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>op2</td>
<td>89</td>
<td>89</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Process Area 2</td>
<td>100</td>
<td>58</td>
<td>2</td>
<td>1</td>
<td>65</td>
<td>66%</td>
<td>33</td>
</tr>
<tr>
<td>op3</td>
<td>65</td>
<td>65</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>op4</td>
<td>33</td>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Station 2</td>
<td>200</td>
<td>206</td>
<td>12</td>
<td>2</td>
<td>126</td>
<td>61%</td>
<td>NVABR</td>
</tr>
<tr>
<td>op11</td>
<td>97</td>
<td>97</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>op12</td>
<td>33</td>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>op13</td>
<td>45</td>
<td>45</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>op14</td>
<td>31</td>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Enter this To**

**HIDE_COLUMN** CAPACITY

**HIDE_COLUMN** WORK_CONTENT

**HIDE_COLUMN** ALLOCATED_TIME

**HIDE_COLUMN** REMAINING_TIME

**HIDE_COLUMN** NUMBER_OF_WORKERS

Hide the **Capacity** column.

Hide the **Work Content** column.

Hide the **Allocated Time** column.

Hide the **Remaining Time** column.

Hide the **Number of Workers** column.
Enter this                               To

HIDE_COLUMN CATEGORY_VA                Hide the VA column.
HIDE_COLUMN CATEGORY_NVA                Hide the NVA column.
HIDE_COLUMN CATEGORY_NVABR             Hide the NVABR column.
SHOW_COLUMN BOM-line-property-name     Display a column for the specified BOM line property.

Configuring standard text

Teamcenter enables creating work instructions using standard text as building blocks. These work instructions are associated with operations or processes in the process structure.

The standard text library is a Teamcenter structure comprised of standard text folders and standard text elements nested under the folders. Standard text is associated with standard text elements and is accessible in the Teamcenter Attachment pane.

The benefits of standard text include:

• Standardization and consistency of work instructions throughout the entire organization.

• Managing standard text content. Standard text enables you to write once only and reuse the same text many times.

• Easy maintenance of up-to-date work instructions when products are upgraded. Make a single change in a procedure and propagate the change through all the relevant documents.

• Improved accuracy. Work instructions are intrinsically connected with the process structure; they are a textual expression of the manufacturing process.

• Work instructions are composed by the planners or engineers who designed the product, increasing accuracy.

• Standard text is stored in the Teamcenter database, enabling you to search for text by its ID, name, or the text itself.

• Easy localization.

• Easy transfer of documents between different virtual engineering systems.

Typically, there are three user roles involved in creating work instructions:

• Administrator

   Responsible for creating and maintaining standard text and work instruction templates. If required, the administrator can configure different system templates for standard text and work instructions. The administrator can change
these templates or create new ones according to organizational requirements. The templates are in Microsoft Word DOCX format and are associated with a dataset object.

- **Librarian**
  Responsible for creating standard text libraries, managing standard text folders and elements, writing standard text, and creating and managing standard text symbols. Standard text is written in a generic style so that it can be reused in varying contexts. An example of standard text is a procedure describing how to mount a wheel on a car and this can be reused in multiple projects. The librarian uses the standard text editor to write standard text elements and the standard text library interface to manage them. Additionally, the librarian may use the **Teamcenter Search** view and the **Results** window as aids.

- **Planner or engineer**
  Uses standard text as building blocks to compose work instructions. Planners cannot edit standard text in the standard text library; however, they may copy and edit standard text and save it in a new document associated with an operation or process in the process structure. Additionally, the planner may use process structure, the textual work instructions editor, **Teamcenter Search** view, and the **Results** window as aids.

For more information about using standard text to create work instructions, see the *Manufacturing Process Planner Guide* or the *Part Planner Guide*.

Administrators must configure Teamcenter to enable the standard text feature.

**Prerequisites:**

- The Teamcenter client must be installed on a Microsoft Windows platform (this limitation does not apply to the Teamcenter server).

- Requires Microsoft Word 2007 SP2 (or later).

- Requires a Teamcenter Manufacturing Documentation license.

You can use Access Manager to configure various permissions for different roles. Siemens PLM Software recommends using the scenario of administrator, librarian, and planner, as described.

For more information about how to configure role privileges, see the *Access Manager Guide*.

**Set up standard text**

1. Using the Teamcenter Environment Manager, install the standard text templates. By default, the same template is used by both the **Standard Text** and **Textual Work Instructions** views; however, you can set up the system to use different templates for each view.
During installation, Teamcenter imports the following work instruction templates to the Teamcenter Environment Manager:

- Default_WIObjectTemplate.docx
- STX_TC91OOTB_spec_template
- WI_TC91OOTB_spec_template
- TWI_TC10OOTB_spec_template
- TWIfor3DPDF_TC10OOTB_spec_template

For more information about how to use Teamcenter Environment Manager, see the Installation on Windows Servers Guide.

During installation, Teamcenter updates the MEWiStandardTextTemplate preference that defines the template to be used by the Standard Text view and the MEWiWorkInstructionTemplate preference that defines the template to be used by the Textual Work Instruction view. These preferences enable you to use different templates for the Standard Text and Textual Work Instructions views. Both preferences are updated to the TWI_TC10OOTB_spec_template value.

2. If Microsoft security bulletin MS09-073 is installed on your machine, you may encounter the following error messages when loading work instructions. Microsoft has provided complete information on this issue at:

http://support.microsoft.com/kb/973904/

- Converter mswrd632

Open Regedit and do one of the following:

- For a 32-bit machine, delete the following registry entry:
  
  HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Shared Tools\Text Converters\Import\MSWord6.wpc

- For a 64-bit machine, delete the following registry entry:
  
  HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Shared Tools\Text Converters\Import\MSWord6.wpc

- Windows 6.0 files

  Open Regedit and do one of the following:
Configuring the manufacturing environment

- For a 32-bit machine, add the following registry entry:
  HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Applets\Wordpad

- For a 64-bit machine, add the following registry entry:
  HKEY_LOCAL_MACHINE\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Applets\Wordpad

- For both 32-bit and 64-bit machines, do the following:
  a. Select the new entry in Regedit, choose Edit→New, and select DWORD Value.
  b. Type AllowConversion for the name of the DWORD, and press the Enter key.
  c. Right-click AllowConversion and choose Modify.
  d. In the Value data box, type 1 and click OK.
  e. Close Regedit.

3. Customize List of Values Data Collection type:
   a. In the Business Modeler IDE:
      A. Create an LOV business object of any basic type: String, Integer, and so on (for example, ProjectStatusLOV: Green/Yellow/Red).
      B. Create a subclass of Mes0MELOVDCDForm (for example, ProjectStatusDataCollection).
      C. For ProjectStatusDataCollection:
         i. Create a property of the same type as the LOV basic type (for example, Value).
         ii. Connect the property to the ProjectStatusLOV LOV.
   b. Deploy the changes to the database.

4. Customize the List of Values Planner Input Field type:
   a. In the Business Modeler IDE:
      A. Create an LOV business object of any basic type: String (for example, ProjectStatusLOV: Green/Yellow/Red).
      B. Create a subclass of Mes0MELOVPFBaseForm (for example, ProjectStatusPlannerInputField).
      C. For ProjectStatusPlannerInputField:
         i. Connect the property mes0Value to the ProjectStatusLOV LOV.
         ii. Connect the property mes0DefaultValue to the ProjectStatusLOV LOV.
b. Deploy the changes to the database.

**Standard text template**

By default, a single template is defined for both the standard text and work instructions editors. The template contains the following:

- **Symbols section**
  
  Symbols associated with the standard text or the work instructions are displayed in a dedicated symbols section at the top of the document. When editing templates, leave this section at the top of the document (first line) or remove it if symbols should not be displayed. Do not insert the symbols section in the header or footer.

- **Character styles**
  
  Use these to create the *look-and-feel* of the document. You can edit these styles (see *Edit standard text templates*), but you must retain the same style names.

  o **WIFreeTextStyle**
    
    Applied to free text fragments in work instructions. The formatting is: Font: Calibri, Size: 13 points, Color: Blue.

  o **WIDataCollectionStyle**
    
    Applied to the placeholder string of data collection definitions. The formatting is: Font: Calibri, Size: 13 points, Color: Gray-50%, Bold.

  o **WIPlannerFieldStyle**
    
    Applied to the value of planner input fields. The formatting is: Font: Calibri, Size: 13 points, Italic.

  o **WIIInvalidPlannerFieldStyle**
    
    Applied to invalid planner input fields identified while running the *Validate PIF* command.

    For more information, see the *Manufacturing Process Planner Guide*. The formatting is: Font: Calibri, Size: 13 points, Italic, Color: Red.

    **Note** If you add new styles, ensure that the *Style type* is **Character**. If you add new styles whose *Style type* is **Paragraph**, errors may occur.

- **Header and footer**
  
  By default this is empty. You can add your company logo or any other pertinent information. You can also add BOM line properties, see *Add BOM line properties*. 
Using the standard text template

When using the standard text template to create standard text, the following Microsoft Word commands are available on the General and Font Style tabs.

Edit standard text templates

Standard text templates are stored by default in the %Root%\install\cmtmes directory and are also imported to Teamcenter as part of the installation. They are the responsibility of the system administrator. The administrator can format the default standard text template and create more templates by copying existing ones and editing them for different uses. Some editing functions can be performed directly on standard text templates from Teamcenter. Other editing functions require that you access the template from the %Root%\install\cmtmes directory and perform the following three-stage procedure on the template:

- Remove the user interface customization that limits the available Microsoft Word commands.
  For more information, see Remove customization.

- Edit the template.

- Restore user interface customization.
  For more information, see Restore customization.

For more information about editing options, including a detailed list of all the template editing function with information about whether it can be performed directly from Teamcenter or whether you must access the template from the %Root%\install\cmtmes directory and perform the three-stage procedure, see Editing options.
When creating new templates, always use one of the document templates supplied with the Teamcenter installation as your starting point. Either copy one of these templates, rename it, and edit it; or open a template, edit it, and perform a Save As operation. You can clone a template in one of the following ways:

- Use a template from the installation directory:
  1. Select a template from the %Root%\install\cmtmes directory and copy it to an accessible location.
  2. Clear the read-only attribute of the file.
  3. Modify the file name so that you can import it later; the import fails if there is already a template with this name.
  4. Perform the following modifications:
     a. Remove the user interface customization that limits the available Microsoft Word commands.
        For more information, see Remove customization.
     b. Edit the template.
     c. Restore user interface customization.
        For more information, see Restore customization.
  5. In My Teamcenter, choose Tools→Import→Templates→Specification Template to import the new template into Teamcenter.

- Clone a template within Teamcenter:
  2. In the Summary view, click Save As.
  3. Modify the file name so that you can import it later, the import fails if there is already a template with this name.
  4. Click Finish.
     The specification template is created under the Newstuff folder. You can move it to a dedicated folder.

User interface customization ensures document standardization throughout your organization.

After importing the template, define which templates are to be used by the Standard Text and Textual Work Instructions views:
1. Edit the **MEWiStandardTextTemplate** preference so that it contains the name of the new standard text template (without the `.docx` suffix). This template becomes the default template for creating standard text.

2. Edit the **MEWiWorkInstructionTemplate** preference so that it contains the name of the new work instructions template (without the `.docx` suffix). This template becomes the default template for creating work instructions.

Remove customization

1. Change the template file name to include the `.zip` suffix. For example, rename `TWI_TC100OTB_spec_template.docx` to `TWI_TC100OTB_spec_template.docx.zip`. The system recognizes the template file as an archive file.

2. Open the archive file in WinZip, extract `customUI.xml`, `customUI14.xml`, and `.rels`, and store these files for later use.

3. Delete `customUI.xml` and `customUI14.xml` from the archive and close WinZip.

**Note** User interface customization ensures document standardization throughout your organization. Siemens PLM Software recommends retaining the default customization. However, you can edit the `customUI.xml` file to change the customization, or delete the `customUI.xml` file to remove customization.
4. Remove the .zip suffix from the template file name.

Deleting customUI.xml removes all customization from the template file. You can now edit it as any other Microsoft Word file.

Restore customization

1. Save the modifications and close the document.

2. Change the template file name to include the .zip suffix. For example, rename WI_TC9OOTB_spec_template.docx as WI_TC9OOTB_spec_template.docx.zip.

The system recognizes the template file as an archive file.

3. Locate the customUI.xml, customUI14.xml, and .rels files that you saved when you removed the customization, right-click, and choose Add to archive....

4. Type the template ZIP file name in the Archive name box and click OK.

For more information, see Remove customization.

Note Siemens PLM Software recommends restoring the customUI.xml and customUI14.xml files. This restores the default customization. However, you can edit the customUI.xml and customUI14.xml files to change the customization or not restore the customUI.xml and customUI14.xml files to remove customization.

5. Remove the .zip suffix from the template file name.
Editing options

This topic details all the template editing functions. For each one, it notes whether the procedure can be performed directly from Teamcenter or whether you must access the template from the `%Root%\install\cmtmes` directory. To modify the template directly in Teamcenter, access My Teamcenter, expand under the specification template up to the **FullText** element, and double-click this item.

![Editing options](image)

Add BOM line properties

Administrators can add operation or process BOM line properties to the template header or footer. This can be useful, for example, to facilitate quick identification of different documents. This can be done directly from Teamcenter.

1. Launch Microsoft Word.

   **Note** This instance of Word is independent of Teamcenter.

2. Open a blank Word document, and click the **Developer** tab.
If the Developer tab is not available, do the following:

- In Microsoft Word 2010:
  a. Choose File→Options→Customize Ribbon.
  b. From the Customize Ribbon list on the right, select Main Tabs, and select Developer.
  c. Click OK to accept the change and exit the Word Options dialog box.

- In Microsoft Word 2007:
  a. From the Word main menu, open the Options dialog box.
  b. Click the Popular list on the left and select Show Developer tab in the Ribbon.

3. Double-click the top of the document.

   The document header becomes active.

4. Click the Developer tab.

5. Click the Aa Plain Text Content control button.

   Word places an empty plain text content control in the header.
If the property you want to add requires more than a single paragraph, use a Rich Text Content control. For example, the description property may require multiple paragraphs.

6. Select the new text control and click Properties. The Content Control Properties dialog box appears.

7. In the Title box, type a name for the property title. The Title text is used for the content control’s name.
8. In the Tag box, type the name of the BOM line property you want to insert in the template header. To retrieve the property name, proceed as follows:
   
   a. Right-click the column area in Manufacturing Process Planner and choose Insert Column. The Change Columns dialog box appears.

   ![Change Columns dialog box]

   b. Copy the desired BOM line property name to the clipboard.

   c. In the Change Columns dialog box, click Cancel.

9. In the Tag box, paste the BOM line property name.

10. In the Tag box, preface the BOM line property name with one of the following:
    
    - LNP_ for a line property, for example, LNP_bl_rev_object_name for a line object name or LNP_bl_rev_item_revision_id for a line revision identification number
    
    - LPP_ for a line parent property, for example, LPP_bl_rev_object_name for a parent object name

11. In the Content Control Properties dialog box, click OK.

   **Note** Ignore the Locking and Plain Text Properties sections.

12. Open the target template and double-click the top to activate the header area.

13. Type the name of the property you want to insert in the template header.

14. From the Word document, copy the content control you created and paste it after the property name you typed.

15. Add as many BOM line properties as you want to the template header.
Note You can add a table to the document header and insert the BOM line properties into the table. This enables you to add multiple properties in an orderly way.

16. Save and close the template.

17. Discard the original Word document in which you created the content control.

Add or modify template styles

This procedure can be performed directly from Teamcenter.

1. Add or modify styles as needed.

2. Close and save the template file.

Remove symbols section

This procedure requires you to access the template from the %Root%\install\cmtmes directory.

Symbols associated with the standard text or the work instructions are displayed in a dedicated symbols section at the top of the document. When editing templates, leave this section at the top of the document (first line) or remove it if symbols should not be displayed. Do not insert the symbols section in the header or footer.

By default, the symbol section exists in the template and is protected. To prevent symbols from appearing, delete the symbol section as follows:

1. In Word, choose Options→Popular and select Show Developer tab in the Ribbon.

2. Click the Developer tab.

3. The symbol section is not visible, therefore, to place your cursor inside the symbol section, click anywhere in the body of the document and then press the Home key and the right arrow (→) key.

   The symbol section becomes active.

4. From the Controls section, click Properties.

   The Content Control Properties dialog box appears.
5. Clear the Content control cannot be deleted and Contents cannot be edited check boxes.

6. Click OK.

7. Select the symbol section and press the Delete key.
   Siemens PLM Software recommends deleting any empty lines in your template file.

Add symbol section
To add a symbol section if you deleted it:
1. From the Developer tab, choose Controls→RichText Aa.
   Word adds a new control.
   
   **Note** Word automatically places the new control at the top of the document, below the header.

2. Type a single space in the new control.

3. Right-click the control and choose Properties.
   The Content Control Properties dialog box appears.

4. Edit the Title box, adding, for example, Symbols Section.

5. In the Tag box, type SymbolsSection.
   
   **Note** You must type SymbolsSection exactly or the symbol section does not work.
6. Select the **Content control cannot be deleted** and **Contents cannot be edited** check boxes.

7. Click **OK**.

Word creates the symbol section.

Follow these steps exactly or the symbol section does not work.

**Configuring 3D PDF generation**

Teamcenter enables you to create interactive 3D documentation reports that derive directly from process data stored in the Teamcenter database. The reports are in **PDF** format and contain process information and 3D data. 3D PDF reports always consist of a single page.

Teamcenter uses graphic data stored in datasets attached to the process structure. The data originates from Teamcenter or Process Simulate. The recipients of the report do not require Teamcenter to view them; Adobe Acrobat Reader (Version 9.4.3 or later) is sufficient.

Additionally, you can do the following:

- Update existing reports after making changes to the process in Teamcenter.

- Distribute reports as work instructions for the shop floor.

To assist you in quickly creating and formatting effective 3D PDF reports, Siemens provides the following standard templates:

- Templates for use when data originates in Teamcenter:
  - 3DPDF Work Instructions
  - 3DPDF Rich Textual Work Instructions
  - 2DPDF Work-instructions
  - 2DPDF Work Instructions with 2D Images

- Templates for use when data originates in Process Simulate:
  - 3DPDF Work Instructions (Process Simulate)
  - Animated PDF Work Instructions

Creating a typical 3D PDF report can be time-consuming. Therefore, the Teamcenter external server (ETS), consisting of Dispatcher Server and translators, is used to generate the report remotely and free up your computer for other tasks.

For more information about generating 3D PDF reports, see the *Manufacturing Process Planner Guide*.
Importing customized templates

Teamcenter is installed with default 3D PDF templates to suit many needs. If, however, you design your own customized 3D PDF templates, you must import them, as follows:

1. Prepare the following import files:
   - PDF report file.
   - Thumbnail image file in JPG format—this is used by the 3D PDF wizard to represent the template. The recommended size is 94x85 pixels.
   - Stylesheet file in XSL format.
   - Javascript file in JS format (not required for 2D reports).
   - Transfer name file in XML format—the name of this file is used to name the transfer mode dataset.

The import files for the default templates are located at Teamcenter_root_directory\install\cmtmes\pdftemplates. You can duplicate any of these and use them as a starting point for your customized files.

2. Compose an XML file listing the import files and other information, as shown in the following example:

```
<?xml version="1.0" encoding="utf-8" ?>
<ME_3DPDF_ImportData>
  <Templates>
    <TemplateInput name="PDF_Template">
      <PDF file=".\PDF_Template.jpg" type="binary"/>
      <Thumbnail file=".\PDF_TemplatePreview.jpg" type="binary"/>
      <Stylesheet file=".\PDF_Stylesheet.xsl" type="text"/>
      <DynamicsJS file=".\PDF_JavaScript.js" type="text"/>
      <TM name=".\PDF_ExportTM" file=".\ME_Gen3DPDF_ExportTM.xml" overwrite="false"/>
      <Contain2D val="False"/>
      <Contain3D val="True"/>
      <ContainSimulation val="False"/>
      <DataSource val="TC"/>
    </TemplateInput>
  </Templates>
</ME_3DPDF_ImportData>
```

The first entry contains the TemplateInput name node. This is the name of the dataset in Teamcenter and is displayed by the 3D PDF wizard when creating the report.
• Nest the import files under the TemplateInput name node.

• Configure the following parameters:
  o Contain 2D
    Set to True if the 3D PDF report contains 2D elements; otherwise, set to False.
  
  o Contain 3D
    Set to True if the 3D PDF report contains 3D elements; otherwise, set to False.
    
    ![Note] If your report contains both 2D and 3D elements, set Contain 2D and Contain 3D to True.
  
  o ContainSimulation
    Set to True if the 3D PDF report contains simulations originating in Process Simulate; otherwise, set to False.
    
    ![Note] If the data source for the report is Teamcenter, set ContainSimulation to False.
  
  o DataSource
    If the data source for the report is Teamcenter, set this parameter to TC; if the data source is Process Simulate, set it to PS.
  
• Run the install_mes_templates.exe import script to install the template.
  
  o The script is located at Teamcenter_root_directory\bin\.
  
  o Run the following command from the command line:
    install_mes_templates -u -p -g -file.
    
    -u=administrative_user
    -p=password
    -g=group
    -file=inputFile.xml

3. Add the names of the new templates to the ME3DPDFAllowedReportTemplates preference.

### Configure 3D PDF report generation

Teamcenter uses the Dispatcher infrastructure to generate 3D PDF reports.

1. Verify that the VisAutomationApp and Professional or Mockup components of Teamcenter lifecycle visualization are installed on the same host on which 3DPDF Translator is installed.

2. On the same host, define the system variable TC3DSS_ENABLE_PRUNED_LAUNCH and set it to True.
Chapter 8  

**Configuring the manufacturing environment**

3. Set `FMS_HOME` as a system environment variable with the appropriate value and restart the Dispatcher host.

4. Use the Install the TEM installer to do the following:
   - Install **Dispatcher Client for Rich Client**, **Dispatcher Server**, and **Dispatcher Client**.
   - Install the **Generate 3D PDF Reports** translator.
   - Install the Work Instruction templates.

5. For four-tier installations, use the Web Application Manager to add **Dispatcher Client for Rich Client Solution** to the third stage.

6. Verify that the ACL rule for Dispatcher is set in Teamcenter.
   
   For more information, see **Getting Started with Dispatcher (Translation Management)**.

7. Open `Dispatcher_Root_Directory\Module\conf\translator.xml` in a text editor and ensure the `isActive` attribute in the `Tc3DPDFTrans` extension is set to `true`.

8. Open Windows services and restart the following two services in the following order:
   
   a. **Dispatcher Scheduler**
   
   b. **Dispatcher Module**

9. Run the Dispatcher client manually by starting `Dispatcher_Root_Dir\DispatcherClient\bin\runDispatcherClient.bat`.
    
    Teamcenter opens a console.

3D PDF Report prerequisites

The following are required to execute and view 3D PDF reports:

- Teamcenter lifecycle visualization license (Teamcenter lifecycle visualization professional or Teamcenter lifecycle visualization mockup) on the translator machine.

- Acrobat Reader (version 9.4.3 or later) on the client machine.

- Teamcenter Manufacturing 3D PDF documentation license on the same machine as the Teamcenter licenses.

**Configuring the comparison of design to manufacturing features**

If you work with design features in 4th Generation Design (4GD), you can assign these features to the MBOM when working in Manufacturing Process Planner. The design features are stored in design container elements in 4GD. During the assignment action, Teamcenter copies the mapped property values from the design
feature to the corresponding properties on the BOM line of the manufacturing feature. The mapped attribute group properties are copied to their corresponding form properties on the item revision. The forms themselves that contain mapped properties to be copied are created as part of the assign process.

For more information about assigning design features to an MBOM structure, see the *Manufacturing Process Planner Guide*.

To enable the comparison, you must configure the mapping of the properties between the design features and the manufacturing features.

The following constants affect the behavior of the assignment process:

- **Cpd0DFToBOMPropertyMapping** global constant
  Specifies the design feature properties that are mapped to form properties on a BOM line.

- **Cpd0DFPropertiesForMBOM** global constant
  Specifies the initial set of properties carried over during assignment. Your administrator can define a different set of properties for assignment based on a feature type.

- **CPD0AttachToItemRevision** business constant
  Specifies whether attachments are associated to the item revision or BOM line in the context of the root level.

- **Cpd0ItemTypeForDesignFeature** business constant
  Specifies what type of item is automatically created when the corresponding design feature is assigned to the MBOM.

  **Note** When assigning welds (design features) from a 4GD subset definition to an MBOM, Teamcenter displays a system identifier in the ID of the feature form attached to the weld in the MBOM. To avoid assigning the system identifier, your administrator must remove the **Cpd0DesignFeature:cpd0design_feature_id:Mfg0id** value from **Cpd0DFToBOMPropertyMapping** and the corresponding property from **Cpd0DFPropertiesForMBOM**.

### Mapping property and attribute groups between design elements and BOM lines

When assigning a design element into a manufacturing structure, some of the properties of the design element are copied into the new BOM line. The mapping of design element properties to BOM line properties is based on the values of the **Cpd0DFToBOMPropertyMapping** global constant in the Business Modeler IDE. This constant has the following format:

\[
\text{BOM-line-property:design-element-property}
\]

For example:

- **Bl_absolute_transform:DE_absolute_transform**
- **Bl_effectivity:DE_effectivity**
- **Bl_note_type1:DE_custom_attribute1**
The selection of properties that are carried forward during the assign process is specified in the `Cpd0DEPropertiesForMBOM` global constant. The value of this constant consists of a list of design element properties, for example:

```
DE_absolute_transform
DE_effectivity
DE_custom_attribute1
```

The properties that are carried forward to the BOM line are the properties that are available for propagation after running an accountability check.

Additionally, if the design element has an associated attribute group, these attributes are copied to a designated form on the BOM line. The mapping between attribute group and form type is specified in the `Cpd0DEAttrGroupTypeToFormType` global constant. This constant has the following format:

```
attribute-group:form-type
```

For example:

```
EngAttributes:FormEngAttributes
```

**Caution** You should only map one attribute group to one form. Mapping multiple attribute groups to one form results in errors in the partial match report of the accountability check.

For more information about assigning design elements to an MBOM, see the *Business Modeler IDE Guide*.

### Configuring 4D planning

Adding a time component to process planning to simulate and visualize construction over an extended period of time is referred to as **4D planning**. 4D planning is used in very large structures to ascertain what parts are already assembled on a specific date in a specific part of the structure. For example, if you build a ship, you may want to begin fitting the cabins with furniture. It is necessary to know if the piping and electrical work is complete by April 30th to begin subsequent work. You can query your structure, taking the date into consideration, to see what has been assembled by April 30th.

Teamcenter uses an embedded **Schedule Manager** view to help you associate scheduling information with process planning operations. The **Manufacturing – 4D Scheduling** perspective already contains this view, in addition to structure views to assist you. This is the same view that is contained in the Schedule Manager application. Making a change in one changes the data in the other.

To configure 4D planning:

1. Install the **4D Planning** extension using Teamcenter Environment Manager (TEM).
2. Set the time zone on the computer running the rich client to **UTC** (GMT). This is the only supported time zone.
3. In the rich client, open the Organization application and modify the **Calendar** setting.
Configuring the manufacturing environment

a. Set the calendar time zone to **GMT (GMT +00:00)**.

b. Remove lunch breaks in the calendar by clicking **Details** next to the weekday entries and modifying the times.

4. Set the following preferences:

<table>
<thead>
<tr>
<th>Preference</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>MECopyIdInContextToAssignedLine</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>SiteTimeZone</strong></td>
<td>Etc/UTC or Etc/GMT</td>
</tr>
<tr>
<td><strong>ScheduleTaskAvailableAttributesWithRelations</strong></td>
<td>ScheduleTask.fdp0AssociatedProcesses</td>
</tr>
<tr>
<td><strong>QS_QSEARCH_ENABLED</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>QSEARCH_update_enabled</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>QS_BBOX_GENERATION_ENABLED</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>QS_BBOX_GENERATION_FROM_NX_ENABLED</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>QS_TRUSHAPE_GENERATION_ENABLED</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>QS_SPATIAL_ENABLED</strong></td>
<td>true</td>
</tr>
<tr>
<td><strong>QSEARCH_foreground_processing_halted</strong></td>
<td>false</td>
</tr>
</tbody>
</table>

Teamcenter sets the following preferences for you automatically.

<table>
<thead>
<tr>
<th>Preference</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ME4DResultsViewDisplayNameColumnsShownPref</strong></td>
<td>Mfg0BvrProcess.bl_indented_title</td>
</tr>
<tr>
<td><strong>Mfg0BvrProcess.ScheduledStartDate</strong></td>
<td>Mfg0BvrProcess.ScheduledFinishDate</td>
</tr>
<tr>
<td><strong>Mfg0BvrProcess.ScheduledCalculatedDuration</strong></td>
<td>Mfg0BvrProcess.ScheduledCalculatedDuration</td>
</tr>
<tr>
<td><strong>ME4DResultsViewColumnsWidthPref</strong></td>
<td>45, 15, 15, 15</td>
</tr>
</tbody>
</table>
Chapter 8  Configuring the manufacturing environment

<table>
<thead>
<tr>
<th>Preference</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ME4DSearchCriteriaDate</td>
<td>fdp0ScheduledStartDate</td>
</tr>
<tr>
<td>MEPProcessIconPropertyNames</td>
<td>start_date</td>
</tr>
<tr>
<td>MECopyIdInContextToAssignedLine</td>
<td>true</td>
</tr>
</tbody>
</table>

For more information about 4D planning preferences, see the Preferences and Environment Variables Reference.
Appendix

A  Glossary
Appendix

A Glossary

A

absolute occurrence
Relationship between a parent assembly and an item one or more levels down in the structure. The parent assembly is the context in which the absolute occurrence exists. When you select the context assembly and view the structure, you can define data on the absolute occurrence that overrides the data stored on the parent. Compare to occurrence.

activity (manufacturing)
Individual action to be performed within an operation. Each activity is associated with a start time and duration. The total time for the operation is based on the cumulative duration of all activities within the operation.

Activities can be sequenced using time data and predecessor activities.

appearances
Cache of configured BOM lines in the context of a specific top-line item. Appearances are created to allow quick searches of the entire structure.

appearance set
Collection of objects that define the use of every part ever included in an end item. The appearance set enables Teamcenter to rapidly search the product structure of a family of related configurations without caching the entire product structure for each configuration. The appearance set is optional and typically maintained for a few end items that represent large assemblies when fast searching is critical. The system administrator defines the items requiring appearance sets.

Application Interface Viewer
Viewer data tab in applications that allows users to monitor and control PLM XML data exchanges between Teamcenter and another application through a collaboration context.

assembly
Compound object that is assembled from other objects and may add additional information to their interpretation.

• In the context of an assembly, other assemblies are called subassemblies, while noncompound objects are called components.

• A single-level compound object that is distinct from a multilevel product structure or bill of materials, which is a hierarchy of assemblies. Assembly data is stored in a BOM view revision.
assembly classification
Resource assembly ID and name of the class where the assembly is classified. An assembly classification is also the process of associating an assembly with a class and entering the attribute values that characterize it within its class.

assembly view
View of the product data. See also BOM and manufacturing view.

attribute
Named storage variable that describes an object and is stored with the object. Users can search the database for objects using object attributes.

In an object, an attribute is a name/value pair; in the database, an attribute is a field.

attribute propagation
Process of transferring attribute values from one object to another. In Resource Manager, attribute values are propagated from a propagation start point up the assembly structure to the assembly object. For example, on a cutting tool, the cutting material of the insert, the cutting diameter of the extension, and the holder type of the machine adapter can all be propagated to the assembly level. This provides a mechanism to make all attributes required for a cutting tool definition available at the tool assembly level.

B

base product view
The as designed bill of materials of the product. Contrast with manufacturing bill of materials.

BOM
Bill of materials.
  • 100% BOM
    The as sold product configuration, for example, the configuration of a car to be built and shipped to the dealer.
  • 120% BOM
    Partial overlay of selected variant conditions. You cannot build the product from a 120% BOM.
  • 150% BOM
    Overlays of all possible variant configurations. You cannot build the product from a 150% BOM.

See also design bill of materials and manufacturing bill of materials.

BPV
See base product view.

C

class
Set of objects that share the same list of attributes but distinguishable by the value the attributes acquire for specific objects. For example, the Automobile class can be
defined by the brand, color, and price, but each car associated to the Automobile class has a different brand, color, and price combination.

**class folder**
Representation of aggregation of objects. In the Classification and Classification Search dialog class hierarchies, classes are displayed with class folder icons because they represent an aggregation of objects.

**class hierarchy**
Structure defining subclasses that inherit the attributes of their superclasses, also called their parents or ancestors.

**classification**
Process of categorizing objects according to similarity in characteristics. While the objects in one class share the same characteristics, the values of these characteristics may differ. For example, drill bits all share the length and diameter characteristics, but drill bit objects differ in length and diameter.

**classification hierarchy**
Structure used to categorize a company’s data by common attributes.

**classification instance**
Lowest-level component of the classification hierarchy. Also referred to as an ICO (internal classification object).

**client tier**
Teamcenter architectural tier that comprises the Teamcenter clients, Teamcenter integrations with third-party applications, and the third-party applications associated with the integrations.

**clone and reference**
Copy action rule when using templates to create process structures. The object is copied to the database and a reference is made to the database clone. The reference is the same type as the one in the template.

**cloning rules**
In a collaboration context, a set of rules that defines how structures are created when copying other structures of the same type. A cloning rule can also be applied when creating a structure from a template.

**collaboration context**
Teamcenter object that holds a collection of data contained in structure and configuration contexts. This data allows you to capture multiple different Teamcenter structures in one container. You can open a collaboration context in the Multi-Structure Manager application, in Manufacturing Process Planner, or in Part Planner. You can also use a collaboration context to collect data to share with a third-party application. See also **structure context**.

**component**
- Objects used to build up an assembly or subassembly.
- Part in a product structure defined in Teamcenter. A component is the lowest level part in the product structure: it cannot be broken down into subparts.
composition
Special kind of structure context that allows components to be added from one or more structure contexts, each of which may contain a different product structure. Compositions are used for design studies and manufacturing processes that contain data from both product and plant structures.

configuration context (Multi-Structure Manager)
In Multi-Structure Manager, a revision rule, variant rule, and closure rule applied to a collaboration context. See also collaboration context.

consumed item
Item that is required during a manufacturing process or operation and must be ordered regularly to meet the production quota. A consumed item can include parts or components from the product structure in addition to materials such as oil, grease, and gloves.

copy action rule
Predefined rule that defines how a template is used to generate processes. Actions commonly performed include ignore, reference, and clone.

Copy by Reference
Copy action rule when using templates to create process structures. The same database object is referenced in the clone. The relation type is the same as in the template.

D
delivery unit
Subassembly that is manufactured separately and delivered to the assembly plant as a consumed part. One of the operations in the assembly process uses the delivery unit as a consumed part. The components of a delivery unit are not consumed in any of the operations.

design bill of materials
List of components and subassemblies used to define an assembly structure, and the representation of the assembly structure. Compare with manufacturing bill of materials.

device
Complex assembly that is represented as a simple component. A device is designed to work as part of a machine tool or robot.

E
effectivity rule
Rule used to set effective dates on released products and processes with a released status.

end item
Top-level node of an assembly that can represent a product or a factory structure.

enterprise tier
Teamcenter architectural tier that comprises a configurable pool of Teamcenter C++ server processes and a server manager. Larger sites can distribute the pool of
server processes across multiple hosts. Smaller sites can run the pool of servers on the same host as the Web tier.

**equipment**
Description of the equipment used to perform manufacturing operations.

**F**

**feature**
Physical or geometric object associated with a product, component, or part. Alternatively, a logical attribute of a product, component, or part. Examples: a weld point, a signal, or a geometric pattern. A feature may be represented by a generic design element (GDE) in a BOM. See also *generic design element*.

**folder**
Graphical representation of an aggregation of objects, such as a group, class, or subclass. For easy distinction in the class hierarchy, each of these aggregations has a different type of folder icon associated with it: a group folder icon, a class folder icon, or a subclass folder icon.

**four-tier architecture**
Teamcenter architecture that includes four tiers: resource tier, client tier, Web tier, and enterprise tier. Contrast with *two-tier architecture*.

**four-tier deployment**
Deployment of the Teamcenter four-tier architecture. The Web tier, enterprise tier, resource tier, and client tier can each be hosted on the same or separate computers.

**G**

**GDE**
See *generic design element*.

**generic design element**
BOM item that cannot have different revisions. See also *feature*.

**group**
Type of class that does not have a list of attributes associated with it; highest level in the classification hierarchy.

**group folder**
In the classification hierarchy, group folders represent a group of related classes.

**H**

**hierarchy**
Structure in which each node can have only one parent but possibly multiple siblings and children.

**I**

**ICM root folder**
Root folder in the classification hierarchy. There is one root per database.
Appendix A  Glossary

Ignore
Copy action rule when using templates to create process structures. No action is taken to duplicate the object in the cloned structure.

in-process model
Product resulting from application of a manufacturing operation.

input data
In-process model and other data generated from the previous steps in an operation.

instance
Single data object that is associated to a class. The instance can correspond to a line in the BOM.

M

manufacturing bill of materials
Defines how the product is manufactured, rather than how it is designed. Compare with design bill of materials.

manufacturing process
Collection of manufacturing subprocesses, operations, and activities that make up a process plan. Processes can have both sequential and parallel ordering. They are associated with a product and a work area.

Manufacturing Process Planner
Teamcenter manufacturing process management application that enables a user to design a plan detailing how to manufacture a product.

manufacturing view
Hierarchical structure of occurrence groups. The manufacturing view describes the components and subassemblies used by the assembly operations.

The components of subassemblies in the manufacturing view represent references to lines in the targeted product structure. These components can be consumed in operations if their parent assembly is not consumed.

See also assembly view.

MBOM
See manufacturing bill of materials.

method
Description of how equipment is used to perform work on a feature. Each method can be used by several manufacturing operations to perform work on different features.

Multi-Structure Manager
Teamcenter application that enables users to view and manipulate data in a specific context.

O

occurrence
Hierarchical structure relationship between the immediate parent assembly and its child component item or item revision in a precise assembly. Sometimes called relative occurrence.
occurrence group
Collection of occurrences and absolute occurrences in the BOM. An occurrence group typically represents an assembly.

occurrence path
Representation of the path from a top-level assembly to an occurrence of a component or subassembly. An occurrence path is unique to the context of a specific BOM; different BOMs cannot contain the same occurrence paths. The occurrence path does not change if the configuration of the BOM changes.

output data
Data generated as a result of applying instructions to input data. Output data can be the resulting in-process model and any instructions for the next step.

P

Part Planner
Teamcenter manufacturing process management application that enables a user to design a plan detailing how to manufacture a part.

plant
Manufacturing facility described by a hierarchical structure of work areas.

Plant Designer
Teamcenter manufacturing process management application that allows users to design, modify, import, and export a factory structure.

plant structure
Hierarchy of the physical layout of a work area. Different levels in the hierarchy represent the plant, a work cell, and individual workstations. Compare with product structure and process structure.

process operation
Step in the manufacturing process executed at a specific work area. It is the lowest revisable element in the manufacturing process structure.

process revision
Modified version of a process. A process revision can be used to handle different configurations of assemblies, alternative methods for building the target item, or changes to methods.

process structure
Hierarchy of manufacturing processes and operations with a sequenced relationship that together describe how a related product is manufactured. Compare with product structure.

product
Item or assembly (hierarchy of components and subassemblies) to be manufactured.

product appearance
Persistent representation of a product line. All appearances of a product are collected into a set that is associated with one revision rule. This allows one set of appearances that can be configured by the effectivity of the part they represent.
**product structure**
Hierarchy of assembly parts and component parts with a geometric relationship between them, for example, a bill of materials (BOM). Variant and revision rules define the generic BOM. This BOM can then be loaded to display the configured variant.

**product view**
Saved configuration of the assembly viewer, including the selection of objects, zoom factor, rotation angle, and pan displacements.

**propagation**
Process of transferring characteristics of one object to another object.

**propagation start point**
Component within an assembly structure that is the starting location for propagation.

**PSP**
See *propagation start point*.

**R**

**raw material**
Initial in-process model before any manufacturing operations are performed.

**relative occurrence**
See *occurrence*.

**Report Generator**
Teamcenter manufacturing process management application that provides a format for producing reports about information in Teamcenter manufacturing process management.

**resource**
Item used to perform an operation or define a process. Examples of resources include robots, tools, and machines. Both standard equipment and custom tools can be identified as resources.

**resource assembly**
Set of resource components and/or subassemblies that are grouped to create an assembly to be used in a process, such as a manufacturing process.

**Resource Browser**
Plug-in component that allows users to retrieve classification-related data, such as a hierarchy with corresponding groups, classes, and classification objects from a Teamcenter database when working in an external application.

**resource component**
Object that is a component of a resource assembly or subassembly.

**Resource Manager**
Teamcenter manufacturing process management application that enables a user to store and retrieve resource-related data such as tools, fixtures, machines, and process templates from a company-wide accessible database.
resource structure
Structure in which resource assemblies are hierarchically built.

resource tier
Teamcenter architectural tier comprising the database server, database, file servers, and volumes.

root
Starting point of a hierarchy. Hierarchies are usually displayed as hanging trees with the root of the structure at the top and the leaves at the bottom.

S

setup
In a manufacturing environment, configuration of the work area. The setup also identifies the parts consumed and the resources used.

spare part
Small item that in Resource Manager is commonly entered as other components are. Examples include nuts, bolts, washers, and screws.

structure
Representation of multiple objects and their interdependencies. For example, a classification structure represents classes and their inheritance dependencies, and an assembly structure represents how components and subassemblies are associated to build up an assembly. The structure can be viewed in several applications, including Structure Manager, Manufacturing Process Planner, Part Planner, Multi-Structure Manager, and Resource Manager.

In Resource Manager, most structures are hierarchical. For example, they acquire the form of a tree where each node can have only one parent but multiple siblings and children.

structure context
BOM or assembly structure contained in a collaboration context. The structure context can contain occurrence groups, items, and item revisions. See also collaboration context.

subassembly
Assembly that is built into the assembly structure of another assembly or intended for that use. In a manufacturing view, either a delivery unit or a workpiece. See also delivery unit and workpiece.

subclass
In the Classification Search Dialog, subclass instances represent a subset of attributes corresponding to a class. Subclasses inherit the attributes of their parent classes. Unlike classes, which inherit every attribute of their parent classes and cannot be edited, users can define the inherited attributes assigned to a subclass.

T

Teamcenter Environment Manager (TEM)
Tool with a wizard-style interface that installs Teamcenter servers and two-tier and four-tier rich clients. TEM also performs maintenance operations, such as upgrading servers, applying maintenance packs, and installing patches. Teamcenter installers
launch TEM using the `tem.bat` command (on Windows systems) or the `tem.sh` command (on UNIX or Linux systems).

**TEM**
See *Teamcenter Environment Manager (TEM)*.

**top level**
Object at the root of a product structure where a process plan is being developed. The top level can be either an end product being manufactured or a subassembly used in the end product (for example, an engine for a tractor where the tractor is the end product).

**two-tier architecture**
Teamcenter architecture that includes a resource tier and a client tier. The resource tier comprises the database server and database. The client tier comprises the Teamcenter rich client, third-party applications that integrate with the rich client, and a local server. This architecture supports only the Teamcenter rich client. Contrast with *four-tier architecture*.

**V**

**variant condition**
- Rules applicable to one component in a product structure.
- Condition set on an occurrence to specify the option values required to configure that occurrence (for example, Load IF engine = 1200).

**variant rule option**
Description of product variations that are usually applied to higher level assemblies or the product itself.

**view (Classification)**
Tailored representation of attributes within a class. Views are associated with abstract and storage classes. Attribute properties can also be applied. For example, a class may define the physical and accounting attributes for its objects, but a view for tool designers may display only the physical attributes, and a view for accountants may display only pricing and order number attributes.

**W**

**Web tier**
Teamcenter architectural tier that comprises a Java application running in a Java 2 Enterprise Edition (J2EE) application server. The Web tier is responsible for communication between the client tier and enterprise tier.

**work area**
Plant location performing an operation. The work area can represent the entire plant, the work line, an individual work cell, or a station within the plant. Work areas are described by their location on the shop floor and the process capabilities they provide. Users can generate a hierarchy of work areas that is unique to their organizations.
**work instruction (manufacturing)**
Document that describes how an operation should be completed at a work area. There are two categories of work instructions:

- Human readable instructions are primarily used to provide information to the operator about how to perform the operation.

- Machine instructions are program files that include numerical code used to run numerically controlled machines, such as robots and NC machines.

**workpiece**
Intermediate state of the product during the manufacturing process. In each step of the manufacturing process, the workpiece is positioned in the work area and the work instructions are performed. The resulting workpiece then flows to the next operation in the sequence, where the next operation is performed.
Index

Numerics/Symbols
3D PDF report
  Configure .................................. 8-47
  Importing customized templates ........ 8-48
3D PDF Reports
  Generate 3D PDF reports ................. 8-49

A
Absolute occurrences ....................... 2-18
Access control ............................. 2-25
Accountability check ....................... 8-1
  Configuring net effectivity ............ 8-3
  Customizing net effectivity user exit 8-3
  Inclusion rules ........................... 8-2
ACLs ......................................... 2-25
Activities
  Assign to operations ..................... 3-6
  MTM data card ............................ 8-6
  Operation ................................ 2-2
Alias file
  Building sample .......................... 8-15
  Create dataset ........................... 8-23
  Defining ................................ 8-13
  Load or reload ........................... 8-24
  Navigating through PLM XML ........... 8-21
  Specifying end object .................. 8-21
  Strategy for working with update .... 8-23
Aliases, publishing pages ................. 8-23
Alignment, checking EBOM-MBOM .......... 2-10
Allocating
  Operations to a plant BOP ............. 7-11
  Processes to a plant BOP ............... 7-11
  Assembly operations ..................... 2-8

B
Baseline .................................... 3-8
BOM lines, customizing quantity ........ 8-27

C
CAM data, importing ....................... 5-3
CAM Integration, additional data
types ......................................... 5-5
Changes
  Configuring ............................... 2-22
  Configuring with intents ............... 2-22
  Controlling access ...................... 2-23
  Managing incremental ................... 2-21
  Tracking .................................. 2-21
Classification hierarchy
  Automatically export ...................... 6-4
  Manually add new classes ............... 6-5
Cloning incremental change ............... 2-14
Cloning rules, define ..................... 2-12
Collaboration context ..................... 2-29
Command Suppression ...................... 1-6
Composition ............................... 4-5
Compositions ............................... 2-18
  Associating data in context .......... 2-19
  Reference structures ................... 2-20
Configuring ............................... 1-1, 3-19
Configuring accountability check ....... 8-1
Configuring processes with rules ....... 2-16
Connection points ......................... 6-11
Consumed items ........................... 2-3
Consumed material ......................... 2-3
controllingOcsForProcessConfiguration
  preference ............................... 2-13

D
Data card ................................... 8-6
  Create custom ........................... 8-7
  Install .................................. 8-7
Documentation ............................. 2-6

E
EBOM, associate with MBOM ............... 4-2
EBOM/MBOM
  Multi-Site ................................ 4-3
EBOP
  Understanding the plant BOP .......... 7-8
  Editing standard text templates ...... 8-36
Effectivity ............................... 2-20–2-21
## Index

**Embedded viewer, installation** .................. 3-13  
**Enabling** ........................................ 1-1  
**Enterprise BOP** ................................. 7-1  
  Allocating to plant BOP ......................... 7-11  
  Assigning tools .................................. 7-12  
  Comparing structures ......................... 7-13  
  Configuring ........................................ 7-16  
  Consuming parts ............................... 7-11  
**Generic BOP** ................................. 7-7  
**Logical assignments** ......................... 7-6  
**Logical designators** ......................... 7-5  
**Logical designators for specific types** 7-6  
**Multi-Site Collaboration**  
  multi-site Collaboration environment .......... 7-16  
  Operations ........................................ 7-10  
  Origins .......................................... 7-9  
  Overview ......................................... 7-2  
  Partitions ........................................ 7-9–7-10  
  Prerequisites .................................... 7-5  
  Processes ......................................... 7-10  
  Product BOP ...................................... 7-8  
  Time analysis .................................... 7-13  
  Visualization .................................... 7-12  
**Equivalence**  
  EBOM/MBOM ....................................... 4-3  
**Excel**  
  Importing structures from ..................... 3-7  
**External integrations** ......................... 1-3  

## F

**Features** .................. 2-4, 2-28  
**Fixture, designing** ......................... 3-11  
**Flexible components** ....................... 4-7  
**Forms**  
  Adding GD&T symbols ......................... 3-19  
  Adding rich text ............................... 3-19  

## G

**GCS** .............................................. 6-11  
**GD&T symbols in forms** ..................... 3-19  
**Generic BOP** ................................. 7-7  
  Comparing ........................................ 7-13  
  Configuring ...................................... 7-16  
  Origins .......................................... 7-9  
**Guided component search** ................... 6-11  

## I

**IDC**, see Intermediate data capture  
**import_nxcam_post_files utility** ............... 5-8  
**Importing NX CAM templates** ................ 5-3  
**Imprecise structures** ....................... 4-3  

**In Context Editor (ICE)** ....................... 3-1  
**In-process model** ............................... 2-3  
**Incremental change when cloning** ............. 2-14  
**Incremental changes** ............................ 2-21  
  Configuring ................................... 8-27  
  Configuring changes .......................... 2-22  
  Configuring with intents .................... 2-22  
  Controlling access .......................... 2-23  
  Intents .......................................... 2-21  
  Tracking changes ............................. 2-21  
**Intents** .......................................... 2-21  
**Intermediate data capture, creating** .......... 3-8  
**Items, consumed** ............................... 2-3  

## J

**Java runtime environment (JRE)** ............... 3-16  
**Java security** .................................. 3-13  

## K

**Kits, managing** .................................. 2-11  

## L

**Load NX CAM data** ............................. 5-3  
**Logical assignments** ......................... 7-6  
**Logical designators** ......................... 7-5–7-6  

## M

**Manufacturing execution systems** ............... 2-1  
**Manufacturing features** ..................... 2-4  
**Manufacturing process**  
  Create .......................................... 3-2  
  Description .................................... 2-2  
  Simulating ..................................... 3-9  
  Verify .......................................... 3-6  
**Manufacturing Process Management, scope** ..... 2-1  
**Manufacturing Process Planner, purpose** .... 1-2  
**Manufacturing process structure** ............... 1-4  
**Manufacturing Resource Library** ............... 6-1  
**Mapping view** .................................. 6-4  
**MBOM** ............................................. 1-4  
**MBOM, associate with EBOM** .................. 4-2  
**MENC Machining** ................................ 5-10  
**MES Integration overview** ..................... 4-8  
**mrn_export_resource utility** ................ 6-4  
**MTM data card** .................................. 8-6  
  Create custom ................................ 8-7  
  Install .......................................... 8-7  
**Multi-Site Collaboration (EBOP)** ............... 7-16
## N
- NC machining ........................................ 2-7
- NX CAM .................................................. 5-2
- NX CAM data, transfer .................................. 5-1
- NX CAM Integration ..................................... 5-10
- MENCMachining ........................................ 5-4
- NX CAM PAX files, managing ......................... 5-3
- NX CAM templates ..................................... 5-3
- Importing ............................................... 5-4
- NX mapping view ....................................... 6-4
- NXLIB_SearchCriteria view ......................... 6-4
- NXLIB_SearchResult view ............................. 6-4

## O
- Occurrences ............................................ 4-4
  - Assigning between structures .......................... 2-9
  - Examples .......................................... 4-4
  - Flexible components .................................. 4-7
  - Synchronizing aligned ................................ 2-10
- Operation activities ................................... 2-2
- Operation types, NX CAM Integration ................ 5-4
- Operations ............................................. 2-6
  - Add resources ...................................... 3-7
  - Assign activities ................................... 3-6
  - Create ............................................. 3-4
  - Creating ........................................... 3-4
  - Creating (EBOP) ..................................... 7-10
  - Definition ......................................... 2-2
  - Revising ........................................... 2-20
  - Visualizing (EBOP) .................................. 7-12
- Origins ............................................... 7-9
- Overview .............................................. 4-8

## P
- Part Planner, purpose .................................. 1-2
- Partitions .............................................. 7-9
  - Creating (EBOP) ..................................... 7-10
- Parts .................................................. 7-11
  - Consuming ......................................... 7-11
  - PAX files .......................................... 5-3
  - PDF reports, generating ............................. 3-21
  - PDI .................................................. 3-8
  - Permissions ......................................... 1-1
- Perspectives ........................................... 1-6
  - In general ......................................... 1-6
- Plant BOP .............................................. 7-13
  - Comparing ......................................... 7-13
  - Plant Designer ..................................... 3-1
  - Plant Designer, purpose ............................. 1-2
- Plant structure ........................................ 2-2
  - Creating from template .............................. 2-11
  - Definition ........................................... 2-2
- Postprocessor files, storing in ......................... 4-4
  - Teamcenter ......................................... 5-6
  - Precise structures ................................... 4-4
  - Preferences, setting .................................. 2-28
  - Preferences, thin client reports ...................... 3-16
- Preliminary data indicator ............................. 3-8
- Process .................................................. 2-2
  - Configuring with variants ............................ 2-17
  - Create structure .................................... 3-3
  - Creating (EBOP) ..................................... 7-10
  - Releasing .......................................... 2-26
  - Revising ............................................ 2-20
  - Visualizing (EBOP) .................................. 7-12
- Process design, managing .............................. 2-28
- Process operation ....................................... 2-26
- Process operation data .................................. 2-6
- Process operations ..................................... 2-2, 2-6
  - Assembly and weld .................................. 2-8
  - Defining NC machining ................................ 2-7
  - Process plan, definition ............................. 2-4
  - Process Simulate ................................... 3-9
  - Process Simulate, purpose ........................... 1-3
  - Process structure .................................... 2-5
    - Creating from template ............................ 2-11
    - Definition ......................................... 2-2
  - Process templates .................................... 2-15
  - Product BOP ......................................... 7-8
    - Comparing ......................................... 7-13
    - Configuring ....................................... 7-16
- Product structure ...................................... 2-2
  - Creating from template .............................. 2-11
  - Definition ........................................... 2-3
  - Open .................................................. 3-2
  - Product views ........................................ 2-27
    - Update multiple .................................... 8-1
  - Properties, aligning ................................ 2-8
  - PublishBatch ........................................ 8-25
    - Enabling .......................................... 8-25
    - Troubleshooting translator ......................... 8-26
- Publishing page ......................................... 8-25
  - Batch mode .......................................... 8-25
  - Create ............................................... 8-24
  - Roles .................................................. 8-9
  - Set environment variables ........................... 8-11
  - Set preferences ..................................... 8-11
  - Update alias file .................................... 8-23
- Publishing work instructions ......................... 8-9

## R
- Raw material .......................................... 2-3
Relationships, create .................................. 3-5
Relative occurrences ................................. 4-4
Releasing processes ................................. 2-26
Report designs, updating (thin client) ........... 3-18
Report Generator, purpose .......................... 1-3
Reports
  Configuring in rich client .......................... 3-12
  Creating in rich client ............................ 3-11
  Generate ....................................... 3-12
  Prerequisites for thin client ....................... 3-16
  Run batch reports (thin client) .................. 3-18
  Troubleshooting ................................ 3-14
  Updating designs (thin client) ................... 3-18
  View ........................................... 3-12
Resource Browser, purpose ......................... 1-3
Resource Manager, purpose ......................... 1-3
Resources
  Add to operations ................................ 3-7
  Explanation ................................... 2-3
  Managing ...................................... 2-26, 6-1
  Revision rules .................................. 2-16
  Revisions ...................................... 2-20
    Incremental changes ........................... 2-21
    Rules ........................................ 2-16
Rich client perspectives and views ............... 1-6
Rich client viewer
  Java security .................................. 3-13
  Rich text in forms ............................. 3-19
  Rules, configuring processes .................... 2-16

S
Search Criteria dialog box .......................... 6-4
Search Results dialog box .......................... 6-4
Security, Java .................................... 3-13
Setup
  Operation ...................................... 2-3
Simulation drivers, storing in
  Teamcenter ..................................... 5-6
Standard text ..................................... 8-32
  Editing templates .............................. 8-36
  Standard text templates ......................... 8-36
Starting ........................................... 1-2
Studies .......................................... 4-5
Study ............................................ 4-5

T
tcexcel_import utility ............................... 3-7
tcin_cam_template_setup.bat ....................... 5-4
Tecnomatix process designer ....................... 1-4
Templates
  Creating plant structure from ................... 2-11
  Creating process structure from ................ 2-11
  Creating product structure from ............... 2-11
  NX CAM ....................................... 5-2
  Work instructions .............................. 8-9
Templates, process ................................ 2-15
Templates, standard text .......................... 8-36
Thin client ........................................ 2-28
Reports ........................................... 3-16
TiCon integration .................................. 8-8
  Map attributes of elements ....................... 8-8
Time analysis (EBOP) ................................ 7-13
Tool design package
  Integrated workflow ............................. 4-6
Tools
  Assigning (EBOP) ................................ 7-12
Transfer mode
  Installing for Publishing pages .................. 8-13
  Transfer mode (thin client) ..................... 3-18
  Transfer NX CAM data ........................... 5-1

U
upgrade_nx_cam_templates utility .................. 5-2
Using standard text templates ..................... 8-36
Utilities
  import_nxcam_post_files ......................... 5-8
  mrm_export_resources ........................... 6-4
  tcexcel_import ................................ 3-7
  tcin_cam_template_setup.bat ..................... 5-4
  upgrade_nx_cam_templates ....................... 5-2

V
Variants, configuring processes .................... 2-17
Views
  NXLIB_SearchCriteria ........................... 6-4
  NXLIB_SearchResult ............................. 6-4

W
Weld operations ..................................... 2-8
Weld points ....................................... 2-28
Work area ........................................ 2-2, 2-7
  Assign ........................................ 3-2
Work instruction data ................................ 2-7
Work instructions .................................. 2-3
Workflow .......................................... 2-25
Workstation, description ........................... 2-2