



# TEAMCENTER

## Model-Based Systems Engineering

Teamcenter 2412

Unpublished work. © 2025 Siemens

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

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

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

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

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

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

## About Siemens Digital Industries Software

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

Support Center: [support.sw.siemens.com](http://support.sw.siemens.com)

Send Feedback on Documentation: [support.sw.siemens.com/doc\\_feedback\\_form](http://support.sw.siemens.com/doc_feedback_form)

# Contents

## Overview of Model-Based Systems Engineering

<b>What is Model-Based Systems Engineering (MBSE)?</b>	1-1
<b>Overview of Model-Based Systems Engineering roles and business process</b>	1-2
The integrated MBSE approach	1-2
MBSE roles	1-3
Collaborating requirements, design, and changes in MBSE	1-4
Model-Based Systems Engineering business process	1-5

## Configuring MBSE

<b>MBSE Prerequisites</b>	2-1
<b>Configuring requirements</b>	2-1
Requirements engineering configuration process	2-1
Configuring Requirements microservices on .NET-based machines	2-4
Run in Background functionality	2-6
Export to Excel button functionality	2-6
Export to PDF with Korean characters on Linux problem resolution	2-7
Install translators	2-7
Configuring trace links	2-8
Configuring HTML object templates	2-11
Convert requirements created prior to Active Workspace 2.3	2-16
Selectively convert requirements from HTML format to Microsoft Word format	2-16
Update custom Word and Excel templates	2-16
Set preferences for requirements	2-17
Enable full text searches	2-24
<b>Configuring system modeling</b>	2-24
General configuration of system modeling	2-24
Configure how system modeling objects and relations appear in Architecture Modeler	2-27
System modeling preferences	2-28
Enable the Open in SMW menu item	2-29
<b>Install and configure embedded software</b>	2-30
Install embedded software using Teamcenter Environment Manager	2-30
Install embedded software using Deployment Center	2-30
Configure and customize embedded software	2-31
Install the Collaboration Service	2-31
<b>Configuring targets</b>	2-35
Modify the target type color	2-35
<b>Installing and configuring parameter management</b>	2-36
Installing parameter management	2-36
Set preferences for parameter management	2-36
Configure Excel import and export	2-37
Legacy parameter project and group migration utility	2-38

Create or update complex multidimensional parameter value with SOA	2-39
<b>Configuring units of measure</b>	2-39
Display sets for units of measure	2-39
Tool-specific unit ID	2-43
<b>Configuring test management</b>	2-44
Install Teamcenter server features, Active Workspace server extensions, and Active Workspace client features	2-44
General test management configuration	2-45
Configuring global parameters	2-47
Configure Where Used tab to show tests	2-47
Migrate test objects	2-50
Set preferences for test management	2-53
Updated test objects and workflows	2-58
Add participants to the test workflow	2-60
Make parameter definitions required or optional	2-66
Upgrade from versions of Active Workspace before 4.1	2-67
Upgrading from customizations before Active Workspace 5.1	2-69
Data Auto-migration to Active Workspace 5.2	2-69
<b>Configuring views</b>	2-70
MBSE views licensing	2-70

## Managing requirements

<b>Requirements management with MBSE</b>	3-1
<b>Acquiring and authoring product requirements</b>	3-2
Requirement creation process	3-2
Requirement structure	3-3
Requirements management user interface tabs	3-3
Working in Requirements Manager mode with a requirement-specific user interface	3-4
HTML Templates	3-7
Importing and exporting requirement content	3-13
Authoring requirement and paragraph content	3-63
Modify requirements structure	3-91
Assessing requirement quality	3-100
Associating requirements with system model blocks or workspace objects using trace links	3-104
Working with Parameters and Requirements	3-124
View test results in the requirement tree view	3-127
Assign a reviewer to a requirement for approval	3-129
Approval process for a requirement	3-130
Submit requirements to workflow for approval	3-133
<b>Locating and viewing requirements</b>	3-134
Requirement locate and view process	3-134
Tips for locating requirements	3-134
Locate requirement specifications, requirements, and paragraphs	3-134
View requirement specifications, requirements, and paragraphs	3-135
View requirement hierarchies in the Table or Tree view	3-136
<b>Collaborating on requirements</b>	3-137

Collaborate on requirements	3-137
Manage comment threads for requirements	3-139
View revision history for selected requirement objects	3-142
Compare requirements	3-142
Generate a report	3-148
Download requirement components for offline viewing	3-149
<b>Establishing product design targets</b>	3-149
Establish design targets	3-149
Create the assessment method and set the product targets	3-150
Applying targets for the product	3-151
Defining the desired analysis results	3-155
<b>Modeling systems</b>	
<b>What is system modeling?</b>	4-1
System modeling with MBSE	4-1
System modeling business process	4-2
<b>System Modeling tabs</b>	4-3
<b>Creating diagrams of models</b>	4-4
Overview of model diagramming	4-4
Open model diagrams	4-5
Create a diagram from a system block	4-6
Preview and open diagrams associated with a system block	4-7
View and manage system diagrams in Diagrams tab	4-7
Drag content into a diagram	4-8
Manually position systems in a diagram	4-9
Multi-select, move, and align systems in a diagram	4-9
Align systems manually	4-10
Snap to grid	4-10
Add an annotation to a diagram	4-11
Manually route connections in diagrams	4-12
Jumpers and connection arrows	4-14
Submit a diagram to a workflow	4-14
Delete a diagram from a system	4-15
Author component details	4-15
Print model diagrams	4-16
Using anchor nodes	4-16
Understanding rolled-up connections	4-17
Find external connections	4-19
View parts with 3D data and trace links to systems	4-22
View requirements with trace links to systems	4-23
Tips for navigating model diagrams	4-23
Modify your view of models using options in the Architecture tab	4-25
Modify models using options in the Architecture tab	4-34
View and modify models using Relations Controls	4-37
Saving a view of an architecture model	4-41
<b>Working with ports and connections</b>	4-42
How you use ports and connections	4-42
Adding, moving, and removing connections	4-42

Managing ports	4-45
Editing port and connection properties	4-47
<b>Managing interfaces</b>	4-49
Interface management with MBSE	4-49
View interfaces for a system	4-50
Display interfaces and related connections, exchanges, parameters, and requirements	4-52
<b>Associating system model blocks with other objects using trace links</b>	4-53
View trace links	4-53
Create a trace link from a system model block	4-56
Modify the trace links in a model diagram	4-57
<b>Managing embedded software projects</b>	4-59
What is Embedded Software Management?	4-59
Representing embedded software in Teamcenter	4-61
Linking embedded software components to artifacts in Polarion	4-63
Creating links between Embedded Software Management components	4-69
Generating reports of trace links between Embedded Software Management components	4-69
Upload binary or software package files to Teamcenter from remote systems	4-71
View and edit resource information for external applications in Active Workspace	4-73
View links to objects in external applications using the Relations tab	4-74
<b>Open models in standalone Lifecycle Visualization</b>	4-74

## Managing tests

<b>Design verification and validation with MBSE</b>	5-1
<b>Recommended design validation process</b>	5-4
Design validation process through testing	5-4
Create and link requirements to systems, assemblies, and subassemblies	5-6
Plan test activities	5-7
Verifying and testing against requirements	5-10
<b>Managing test parameters</b>	5-21
Parameter management overview	5-21
Associating target values with verification requests using parameters and tests	5-22
Defining and specifying target values using parameters	5-25
Test and parameter project configuration relationships	5-33
Manage parameters for an item	5-33
Manage input and output parameters	5-33
Create trace links with parameters	5-34
Find where an object is used	5-34
Synchronize and publish testing results parameters back to system model and requirement parameters	5-35
<b>Managing and monitoring test packages</b>	5-36
Managing test management workflows	5-36
Create a test or verification request	5-39
Create a verification or test request in context	5-40
Managing worksets	5-41
Manage test packages and test objects	5-44
Common test management screen actions, panels, tables, and charts	5-61

Run tests, monitor test coverage, and view test results	5-68
<b>Managing simulation analyses with Multi-disciplinary Analysis and Optimization (MDAO)</b>	5-74
MDAO simulation in Active Workspace	5-74
Manage simulation requests	5-74
Manage multidisciplinary model and analysis creation and execution from the verification request	5-78
Include MDAO model and MDAO analysis object in a verification request	5-79
Automap parameters	5-80
Manually map parameters from system models, requirements, and integrated simulation products to simulation model parameters	5-80
Perform automated simulations and record test results	5-81
Display MDAO optimization results	5-82
<b>Reporting test results</b>	5-83
Create a test report	5-83
View completion status and test results	5-83
<b>Managing global parameters</b>	
<b>About parameter management</b>	6-1
<b>Creating and populating parameter dictionaries</b>	6-3
Manage parameter dictionaries	6-3
Create a parameter dictionary	6-4
<b>Creating parameter definitions</b>	6-5
About parameter definitions	6-5
Create a parameter definition within a dictionary	6-5
Revise a parameter definition in a parameter dictionary	6-6
Release parameter definitions	6-7
<b>Creating parameters</b>	6-8
About parameters	6-8
Create a parameter in a project, set, or other location	6-8
Create a parameter directly in the product	6-10
Override parameter value in context	6-12
Create a weight parameter	6-12
Managing table value parameters	6-14
<b>Managing parameters with parameter projects</b>	6-17
Parameter project lifecycle	6-17
Associating parameters with a project	6-17
Create a parameter project	6-18
Create a parameter project from another parameter project	6-18
Open a parameter project	6-19
Create a new parameter set in a parameter project	6-19
Freeze parameter projects or elements	6-20
Associate a parameter dictionary with a parameter project	6-22
<b>Comparing project parameters</b>	6-22
Performing tradeoff analysis	6-22
Compare parameters	6-23
Compare multidimensional table value parameters	6-25
<b>Configuring parameters with variants</b>	6-27

Managing variants outside the product or program	6-27
Managing parameter variants in the product or program	6-29
<b>Add parameters to a verification request</b>	6-31
<b>Revising parameters</b>	6-32
Configuring revisions of parameters	6-32
Revise a parameter value	6-33
<b>Importing and exporting parameters for a project</b>	6-34
Sharing parameters with external applications	6-34
Importing parameter definitions	6-35
Importing parameters	6-36
Export parameter definitions from the parameter dictionary	6-44
Configure an export template for parameters	6-45
Configuring Excel templates	6-50

## Managing characteristics

<b>About characteristics</b>	7-1
Characteristics and CAD designs	7-1
The process for saving characteristics to Teamcenter	7-2
The characteristics file	7-3
<b>Characteristics prerequisites</b>	7-3
<b>View characteristics data for a part</b>	7-4

## Exploring data with customized queries using recipes

<b>About recipes and samples provided</b>	8-1
<b>Create and execute a recipe</b>	8-1
<b>Create a recipe in context</b>	8-2
<b>Load, edit, save, or copy a recipe</b>	8-3
<b>Execute a recipe in context</b>	8-3
<b>Change the configuration</b>	8-5

# 1. Overview of Model-Based Systems Engineering

## What is Model-Based Systems Engineering (MBSE)?



*Model-Based Systems Engineering (MBSE)* is a systems engineering methodology that focuses on creating and using domain models as the primary means of information exchange between engineers, rather than on document-based information exchange. MBSE is generally defined as a formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases

Products typically include many different types of components, such as mechanical, electrical, and software parts. These products are not developed in a single domain but rather must capture the whole-product architecture in a product life cycle environment. Teamcenter provides a cross-domain framework to capture and map the relationships needed to make global and cross-domain design decisions required to develop these products.

MBSE brings together multi-domain (such as mechanical, electrical, and software) product development along with considerations like cost, reliability, and manufacturability. MBSE helps define what will be built, verify the design, and orchestrate the downstream development process.

Although each of the MBSE feature areas has more granular roles, the initial content areas of interest are for administrators and business end-users involved in product planning and development. The following table describes these roles and their general functions using MBSE.

### Where do I go from here?

 Administrator	MBSE administrators includes those who install MBSE and those who configure the specific areas of MBSE such as requirements management or verification management.
Configure specific MBSE features	See Configuring MBSE.
 Business User	MBSE business end-users include all team members involved in product planning, such as: engineers, analysts, testers, and designers.
Learn more details about the MBSE roles, communicating during the product life cycle, and the MBSE business process.	Start with the topic The integrated MBSE approach.
Get me started with the basics of using Active Workspace.	See Active Workspace Fundamentals.
Get me started with a specific MBSE main feature area.	See:

	<ul style="list-style-type: none"> <li>• Requirements management</li> <li>• Product architecture and system modeling</li> <li>• Test management</li> </ul>
Get me started with other important features that are integrated throughout the MBSE main features.	See: <ul style="list-style-type: none"> <li>• Change management</li> <li>• Parameter management</li> <li>• MBSE Views</li> </ul>

## Overview of Model-Based Systems Engineering roles and business process

### The integrated MBSE approach

With integrated MBSE approach, Active Workspace helps you understand how your development process pieces fit together:

#### Product definition

The product definition as part of an integrated MBSE approach is the primary creation phase where all the required assets are authored such as requirements, parameters, system architectures, interface diagrams, and domain architectures to 1D/ 3D models, along with software and test specifications. The collection of these elements is then used to build models that represent system and subsystem physics, and the connection of those systems through interfaces across all relevant development domains. Once available, these reusable digital models are used in later development stages of simulation and product validation. This results in a fully specified and traceable product aligned to stakeholder requirements.

#### Connected engineering

Connected engineering has two objectives in support of an integrated MBSE approach.

- Simplify access to system, product, and process models authored in the product definition phase.
- Make these models available and actionable to anyone to those in the development process.

With this approach, software, electrical, and mechanical engineers can collaborate in a true co-development environment.

#### Product verification and validation

Verification and validation are similar but distinct processes as part of MBSE used to check that a product or system meets requirements and specifications established to fulfill its intended purpose. Validation assures that the product meets the demands of the customer and any other identified stakeholders. Verification, on the other hand, is meant to evaluate whether a product, service, or system complies with a regulation, requirement, specification, or other imposed condition. Validation can be expressed as

"Are you building the right thing?" and verification by "Are you building it right?". Building the right thing refers to the user's needs while building it right checks that the specifications are correctly implemented by the system.

## Quality engineering

Quality engineering is part of the MBSE approach and process of meeting requirements for safety, reliability, and security of a product. This involves a transformation from "quality assurance" (typically downstream testing) with a shift left for engineering quality during the initial concept and preliminary design phases. Safety, reliability, and security can be expressed by the following: Product Safety ensures the product does not represent harm or hazard to its users. Product Reliability emphasizes the ability for the product to function under stated conditions for a specified time period. Product Security highlights the prevention of external influence that could impact product safety or reliability. Safety, reliability, and security are important in their own rights and should be maximized; however, the needs of each are often at odds due to conflicts in the priority of stated requirements or how risk will be managed. Each is typically measured by comparing test or simulation results to the government or industry standards.

## MBSE roles

The following roles participate in Model-Based Systems Engineering (MBSE) processes:

- System analyst

Responsible for planning, designing, and implementing systems. Reviews the customer needs, identifies use cases, derives and identifies system requirements, and performs function behavioral analysis.

- System architect

Responsible for planning the architecture and defining the reuse of parts and assemblies. Has insight into assemblies used. Manages the configurable system architecture, including structure, allowed variability, and view/viewpoints. Sometimes referred to as the *technical architect*.

- System designer

Responsible for implementing the system architectural, logical, and physical designs. Creates and manages systems, performs functional allocations, and models system interactions. Sometimes referred to as the *system engineer*.

- System tester

Responsible for verifying that the system meets defined requirements. Reviews the output of the system analyst and system designer to define appropriate test cases for the system.

- Domain engineer

Realizes the system model according to an expertise area (software, electrical, simulation, and so on) to further analyze and document the system. The domain engineer role is typically fulfilled by one of the following:

### **Electronic (ECAD) designer or engineer**

Responsible for designing printed wiring boards (PWBs). Writes firmware. Implements the system model electrical, electronic, and firmware architecture.

### **Mechanical (MCAD) designer or engineer**

Responsible for specifying the structural properties of a product design and designing materials, structures, and systems while considering the limitations imposed by practicality, regulation, safety, and cost. Implements the system model physical architecture.

### **Simulation engineer**

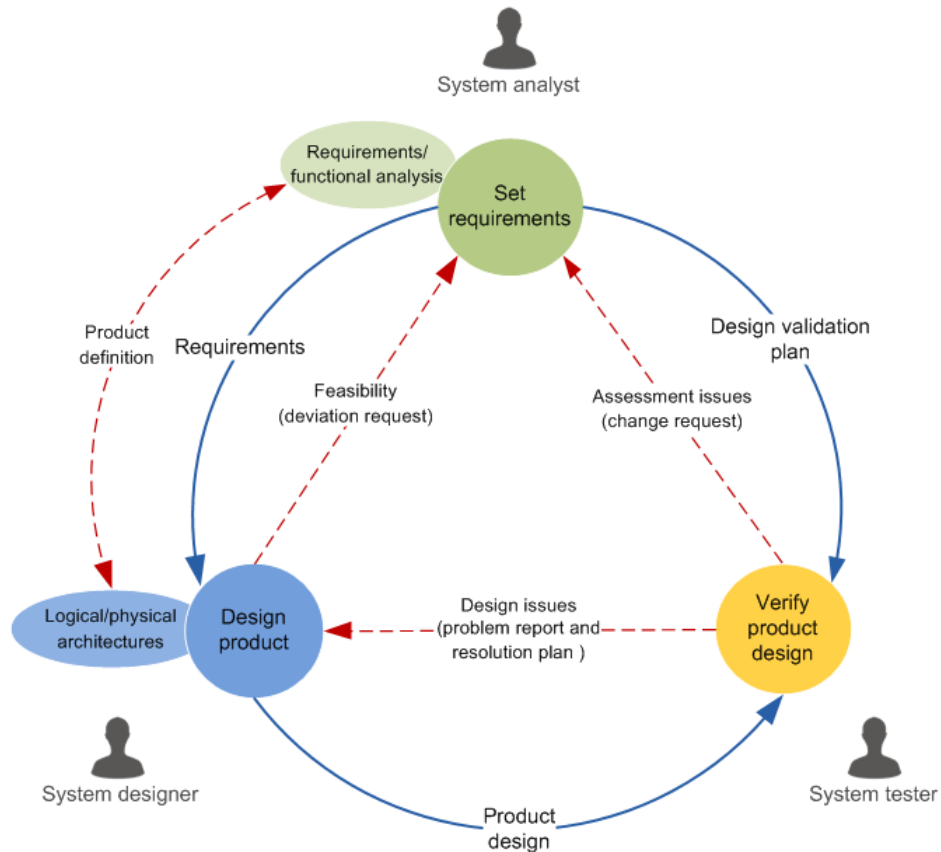
Configures and operates computer-based simulation models. Provides analysis of a functional area and advises best methods of implementing a course of action given the strengths and weaknesses of the simulation. Runs simulations against models and requirements.

### **Application administrator**

Responsible for the configuration of software and data to meet company requirements and user needs. Configures Active Workspace and Teamcenter.

## **Collaborating requirements, design, and changes in MBSE**

The following graphic shows the general process for tracking requirements and changes in MBSE during the product development lifecycle.



System analysts use requirements to communicate decisions to system designers and system testers.

System designers create system models to meet the requirements. The system designers communicate how the system model should be tested and what targets should be made in a design validation plan.

System testers verify that the system models meet the requirement targets. Both the system designers and system testers follow the requirements throughout the development process.

System designers and system testers communicate design issues, feasibility issues, and requirement assessment issues to each other and the system analysts through deviation requests, change requests, problem reports, and resolution plans. System analysts then work with the team to refine the requirements, models, or verifications to correct the issues.

When the product conforms to all requirements, the product is ready to begin physical development.

## Model-Based Systems Engineering business process

The Model-Based Systems Engineering business processes consist of generic processes required to allow designated roles in your organization to develop requirements, model systems, and manage simulations. The following figure shows the business processes in the context of the overall PLM life cycle.

Business process	Description	Roles
<i>Plan program and product</i>	<p>(This is a prerequisite process that is initiated before the Model-Based Systems Engineering process can begin.)</p> <p>Provides the steps to set up the program and product, including:</p> <ul style="list-style-type: none"> <li>• Create the program or project. Tasks include: <ul style="list-style-type: none"> <li>Manage schedule templates.</li> <li>Define milestones and virtual build streams.</li> <li>Identify and manage risk.</li> <li>Define program-level targets.</li> <li>Manage schedules, tasks, resources and deliverables.</li> <li>Monitor program status.</li> <li>Design release to manufacturing.</li> </ul> </li> <li>• Identify and manage changes. Tasks include: <ul style="list-style-type: none"> <li>Identify needs.</li> <li>Analyze, propose and authorize change.</li> <li>Implement change and validate results.</li> <li>Close change.</li> </ul> </li> <li>• Identify and manage issues. Tasks include: <ul style="list-style-type: none"> <li>Raise issues.</li> <li>Assign and track issues.</li> <li>Analyze and dispose issues.</li> </ul> </li> <li>• Plan features and create variability. Tasks include: <ul style="list-style-type: none"> <li>Define initial product architecture.</li> <li>Manage global features, options, and rules.</li> </ul> </li> </ul>	<p>System architect</p> <p>Configuration developer</p>

Business process	Description	Roles
	Define product specific features and product models.	
<b>Requirements management</b>	<p>Provides the steps to acquire and manage requirements. Also includes steps to set targets that satisfy the requirements. Tasks include:</p> <ul style="list-style-type: none"> <li>• Monitor the requirements process.</li> <li>• Create and manage requirements.</li> <li>• Develop verification plans.</li> <li>• Set and validate targets.</li> <li>• Manage requirement and product verification issues through change and deviation requests.</li> </ul>	System analyst System designer System tester
<b>System modeling</b>	<p>Provides the steps to create models that accurately represent the planned product. Tasks represented by this process include:</p> <ul style="list-style-type: none"> <li>• Manage reusable assets (model templates, system architecture templates, and interface specification libraries).</li> <li>• Author the system model from scratch or reusable assets.</li> <li>• Decompose the system model into subsystems and components represented by blocks.</li> <li>• Define system model interfaces, that is, communications points between system model blocks.</li> <li>• Assign requirements and functions to system model blocks.</li> <li>• Allocate parameters and targets to system model blocks.</li> <li>• Define next-level requirements.</li> </ul>	Technical architect System analyst System designer
<i>Test management</i>	<p>Provides the steps to validate if you modeled the correct design. Tasks represented by this process include:</p> <ul style="list-style-type: none"> <li>• Link requirements to systems, assemblies, and subassemblies</li> <li>• Manage libraries of reusable simulation architectures and mathematical models, both global and project-specific.</li> </ul>	System designer System architect Simulation engineer

Business process	Description	Roles
	<ul style="list-style-type: none"> <li>• Verify requirements through inspection, physical, or simulation testing.</li> <li>• Monitor and report test results.</li> </ul>	
<i>Global parameter management</i>	Define and manage different parameter types in a global dictionary. Parameters include: variables, characteristics, measurements, calibration, configuration, and requirements.	System designer System architect Simulation engineer
<i>Explore data with customized queries using recipes</i>	Create recipes, which are customized queries using a form-based interface to automate data collection for specific engineering activities across domains such as verification, analysis, simulation development, design, electrical, and so on.	System designer System architect Simulation engineer
<i>Define physical solution and create layouts</i>	<p>(This is a dependent process that can begin once the Model-Based Systems Engineering process is complete.)</p> <p>Provides the steps to create the design data from which the product will be manufactured. Tasks include:</p> <ul style="list-style-type: none"> <li>• Define physical solutions, including identifying carryover parts and defining new parts, and also releasing product solutions.</li> <li>• Create mechanical layout and manage mechanical design data, including PMI.</li> <li>• Create electrical layout and manage electrical design data, including messages, wiring harnesses, and printed wiring boards.</li> <li>• Develop and manage embedded software.</li> <li>• Manage global parameters, calibration specifications, and version compatibility.</li> </ul>	Lead designer Engineer or designer

# 2. Configuring MBSE

## MBSE Prerequisites

### Prerequisites

Before configuring specific Model-Based Systems Engineering functionality, configure the following features on which MBSE depends:

- Active Content framework, which provides the navigation panel you use to browse search results.
- Search indexing. MBSE data must be indexed to appear in the search results.

Enable users to search for requirements that have rich text associated with them and to search for plain text present in those requirements. Open the *TcFtsIndexer\_objdata.properties* file in `TCRoot\TcFtsIndexer\conf`. Change the value of `indexDatasetFileContent` to **true** and reindex.

- Microservices is installed and **configured**.

## Configuring requirements

### Requirements engineering configuration process

#### Prerequisites

Ensure that the following are installed before you begin the Requirements Management installation and configuration:

- Microsoft .NET Framework
- Microsoft Office with primary interop assemblies (PIAs) for Microsoft Office

The ReqMgtWordToHtml and ReqMgmtHtmlToWord translators require the following:

- Microsoft .NET Framework
- Microsoft Windows

The ReqMgmtPerformOperation and CompareTranslator translators require the following:

- Microsoft .NET Framework
- Microsoft Windows
- Microsoft Word
- The REQ\_BootstrapClientIP environment must be added in the microservice json/yml file. This environment specifies the FMS bootstrap client IP address to be used for the assignment. On Linux hosts, enter the internal IP address of the machine where the import/export microservice is running.

Single IP address is supported. When migrating from TEM to DC, this environment variable may not get populated. In this instance, the value for the REQ\_BootstrapClientIP environment must be updated manually.

**Note:**

For supported versions of third-party software, see the Hardware and Software Certifications knowledge base article on Support Center.

For planning purposes, ensure that you plan to install these components on the same server where you install the Dispatcher. For more information, see *Dispatcher — Deployment and Administration* in the Teamcenter documentation.

### Installation

Unless otherwise noted, perform the configuration in the order indicated.

Follow these steps to install Requirements Management on Active Workspace.

1. *(Required for Export to Excel and Baselining capabilities)* Enable **AsyncService**. For more information, see *Dispatcher — Deployment and Administration* in the Teamcenter help documentation.
2. Install translators that support requirements capabilities.
3. *(Required for importing requirements)* Ensure that the dispatcher is started. For more information, see *Dispatcher — Deployment and Administration* in the Teamcenter help documentation.
4. *(Required to export from Microsoft Word back to Active Workspace)* Install the **ReqMgmtWordToHTML** and the **ReqMgmtHTMLToWord** translators. For more information, see *Dispatcher — Deployment and Administration* in the Teamcenter help documentation.
5. *(Required for suspect trace link notification management)* Install the **Async** translator. For more information, see *Dispatcher — Deployment and Administration* in the Teamcenter help documentation.
6. Enable full text searches.

### Additional Requirements Management instructions and preferences

Follow these steps to install Requirements Management into Active Workspace.

- Convert requirements created prior to Active Workspace 2.3.
- Configure trace links.

- Configure HTML object templates.
- Update custom Microsoft Word templates.
- *(Optional)* Install the Active Workspace Launcher using Teamcenter Environment Manager (TEM) or with a standalone installer.
- *(Required for Excel Live functionality)* Install the Teamcenter Extensions for Microsoft Office extensions.
- Configure the requirement management preferences for the following capabilities:
  - General requirements management preferences.
  - Import Microsoft Word documents.
  - Microsoft Word editing.
  - Convert requirements from Microsoft Word to HTML.
  - Suspect trace link notification management.

## Settings for comparing requirements

These preferences must be set by your system administrator to generate and compare specification history.

- **REQ\_CompareSpecificationHistory=true**

Note:

This preference must be set to **true** to generate specification history. If it is not, historic versions and version information is unavailable.

- **REQ\_Microservice\_Installed=true**

The following also applies:

- The **req\_perform\_operation** must be installed.
- The **compare** microservice must be installed.
- The **History** tab is available when using Teamcenter 12.4 and above, regardless of Active Workspace version.

## Configuring Requirements microservices on .NET-based machines

Follow these steps to configure Requirements microservices on .NET-based machines.

1. If the microservice framework version is 2.0.0, unzip the .NET microproxy file and move its contents to the IIS bin folder. For example:

```
from <tcmsf_kit_location>\dotnet_microproxy-1.0.1.zip\net_webtier\bin
```

```
to %TC_ROOT%\net_webtier_<config_name>\<config_name>\bin
```

2. Go to %TC\_ROOT%\net\_webtier\_<config\_name>\tc\ApplicationConfiguration.xml.
3. Modify the following parameters:

Parameter	Value
microserviceEnabled	true
microserviceAddress	http://localhost:9090 This can be different in case of distributed environment
microserviceKeyStore	signer_keystore.p12
microserviceKeyStorePassword	Pa55w_rd This password is provided while installing microservices.

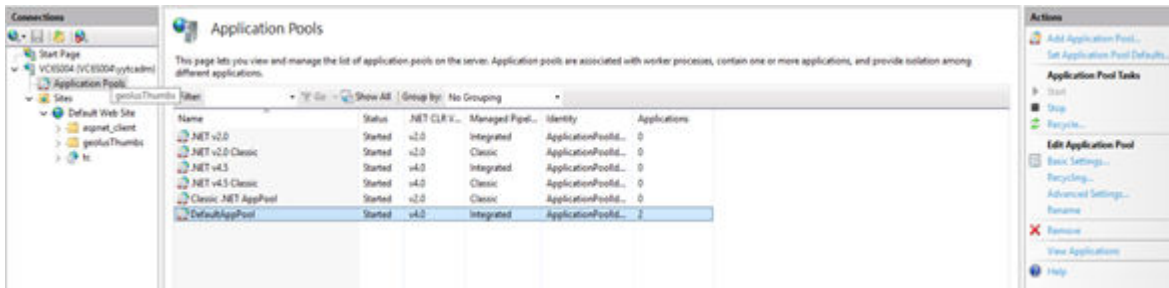
```

118 <!-- Enable/disable microservice requests. (Value: true/false) -->
119 <param name="microserviceEnabled" value="true"></param>
120
121 <!-- Specify the URL(s) to access backend microservices. This is a
122 comma-separated list of addresses for service dispatchers.
123 Each address should include the scheme, host and (optional)
124 port portions of a URL (e.g., "http://localhost:9090"). An IP
125 address can be used for the host. If the port is omitted the service
126 dispatcher must be listening on the default port for the scheme. -->
127 <param name="microserviceAddress" value="http://localhost:9090"></param>
128
129 <!-- Interval (in seconds) between ping requests to detect when a microservice
130 address (see microserviceAddress) has come back on line. -->
131 <param name="microservicePingInterval" value="20"></param>
132
133 <!-- Name of the keystore file that stores private key to sign JWT. -->
134 <param name="microserviceKeyStore" value="signer_keystore.p12"></param>
135
136 <!-- Password to read private key to sign JWT. -->
137 <param name="microserviceKeyStorePassword" value="Pa55w_rd"></param>
138

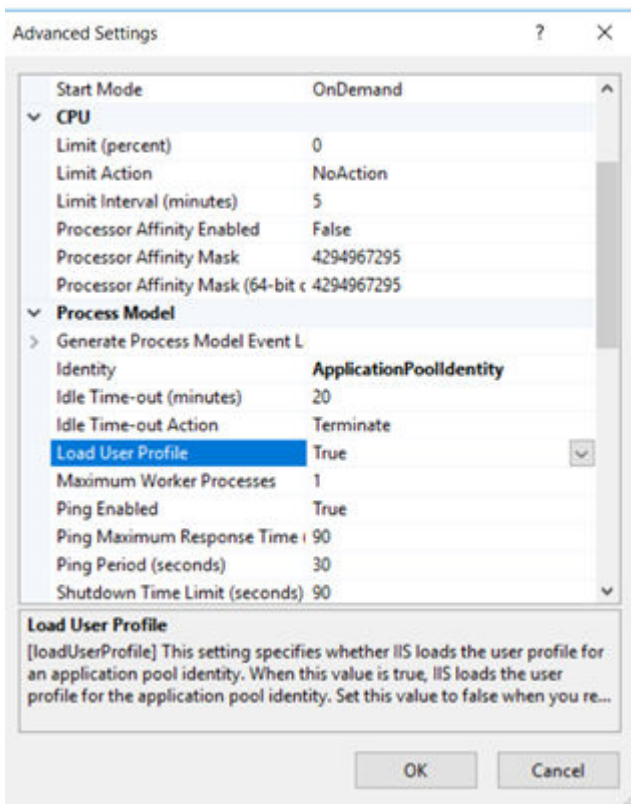
```

4. Copy all files from C:\apps\tc\tc12\Clients\law\signer\_config to C:\Windows\System32\inetmgr.
5. Open IIS Manager and do the following:

- Expand the VC6S004(VC6S004\yytcadm) node.
- Select **Application Pools > DefaultAppPool**.



- Click **Advanced Settings** from the column on the right.
- Change **Load User Profile** to **True**.



6. Restart the IIS server and Microservices pool manager.

## Run in Background functionality

You can enable **Run in Background** functionality to allow the background processing of activities while you continue working. This functionality enables the **Run in Background** check box on panels in Active Workspace.

### Prerequisites

The administrator must configure the AsyncService translator and its environment as described in the *Configuring the AsyncService* topic in the *Enable default translators* chapter in *Dispatcher — Deployment and Administration* in the Teamcenter help collection.

### Procedure

1. During Teamcenter installation with Teamcenter Environment Manager (TEM), open the **Select Translators** page.
2. Select the **AsyncService** check box.
3. Select the **Subscription** option under the **Server Extensions** section.
4. Select the **Subscription** option under the **Active Workspace**→**Client** section.

## Export to Excel button functionality

You can enable the Export to Excel button during Active Workspace installation.

See **Export and import Excel requirements using round-trip** for information on using Export to Excel functionality.

### Restrictions and limitations

- Export to Excel functionality does not support Multi-Field Key (MFK) properties which are typed/ untyped Reference. Only numeric or string properties are supported.
- Creating a packed structure by making use of quantity from Excel import in rich client or Active Workspace is also not supported.

### Procedure

1. When installing Active Workspace, select **Requirements Management** in **Active Workspace client extensions**.

## Export to PDF with Korean characters on Linux problem resolution

When exporting to PDF with Korean characters on Linux, the characters do not export properly. For microservices, we use `req_export_service`. On Linux, microservices are deployed in a Docker container. Even if Korean characters are available on the host Linux machine, the characters are not available to the Docker container where the service is running.

To resolve this problem, map the Korean fonts from the host Linux machine to the Docker container.

1. Edit `req_export_service.yml` and add a section named **volumes**:
2. Map the Korean fonts folder from host Linux machine to Docker container fonts folder:

*Folder\_path\_of\_Host\_Machine\_where\_Korean\_Fonts\_are\_Installed:Default\_Fonts\_Folder\_Path\_in\_Container*

For example:

```
volumes:
  - /usr/share/fonts/KR:/usr/share/fonts
```

## Install translators

You can install and configure the feature to compare the content of two requirements in Active Workspace.

Note:

- Multibyte characters are not supported for the HTML compare feature.
- All translators are Windows-based, and are installed and supported only on Windows systems.

1. Install the Subscription feature during Active Workspace server and client installation:

In the **Features** panel, under **Server Extensions** section, select **Subscription** and under the **Active Workspace** → **Client** section, select **Subscription**.

- a. In the **Features** panel, under **Server Extensions** section, select **Subscription**.
- b. Under the **Active Workspace** → **Client** section, select **Subscription**.

2. Install the Compare Translator as described in *Configuring Teamcenter*, but also perform the following steps:
  - a. In the **Select Translators** panel, select the following translators:

- **ReqMgmtWordToHtml**: Required for converting requirements in Microsoft Word format to HTML.
  - **ReqMgmtHtmlToWord**: Required for converting requirements in HTML format to Microsoft Word.
  - **ReqMgmtPerformOperation**: Required for keyword import from Active Workspace and to generate specification history.
  - **CompareTranslator**: Required for comparing two requirements in Microsoft Word format.
- b. Specify the user and password for the compare translator.
  - c. (Requirements Baseline feature only) Install the Async Translator:

In the **Select Translators** panel, select the **Async Translator** option.

For this option to work, the administrator must have configured the Async service translator and its environment as described in the *Configuring the AsyncService* topic in the *Enable default translators* chapter in *Dispatcher — Deployment and Administration* in the Teamcenter documentation.

## Configuring trace links

### Disable or enable trace links for requirements

Trace links are disabled by default. You can disable or enable the trace link functionality.

A *trace link* is a directional relationship between requirement specifications, functional models, system models, or physical models. In the relationship, one object is the defining object and one is the complying object. The defining object specifies a condition that a product or a component must fulfill. The complying object must partially or completely fulfill the condition specified by a defining object.

1. In the Teamcenter rich client, choose **Edit → Options**.
2. In the **Options** dialog box, select **Systems Engineering**.
3. Select or clear the **Trace Link Mode** check box.
4. Click **Apply** to commit your edits and keep the **Options** dialog box open, or click **OK** to apply edits and close the dialog box.

### Modify style sheets for custom trace links

To view and delete custom trace links, modify the Active Workspace style sheets.

To add the custom *Type\_TraceLink* (real name), you change the corresponding **xrt** for the object.

Example:

For a *Type* that is a requirement revision object, the summary style sheet is **Awp0RequirementRevisionSummary**. Therefore, for a new custom trace link type of **A4BZ\_FND\_Tracelink** (real name) and **BZ\_FND\_Tracelink** (display name) between two requirements revision objects, you add the type name to the trace link section of the style sheet, in both the defining and the complying objects.

Review the sections highlighted in the example code that follows and consider the following rules:

#### <section titleKey="tc\_xrt\_DefiningObjects">

- Add the **S2P:** prefix for secondary-to-primary traversal only.
- Add the **S2P:** prefix to the relation that the object set source traverses.
- To traverse multiple relations or subtypes of the same relation, enter a comma-delimited list of the relation and subtype names. Prefix each relation and subtype name with **S2P:**.

#### <section titleKey="tc\_xrt\_ComplyingObjects">

- The **S2P** prefix is not required for primary-to-secondary traversal; the prefix is required for secondary-to-primary traversal only.
- The **relation** property displays the table column with the relation **display name** as defined in the Business Modeler IDE.

```
<page titleKey="FND_TraceLink" visibleWhen="has_trace_link==Y">
  <section titleKey="tc_xrt_DefiningObjects">
    <objectSet source="S2P:FND_TraceLink.WorkspaceObject,
                  S2P:A4BZ_FND_Tracelink.WorkspaceObject"
              sortdirection="descending" sortBy="object_string"
              defaultdisplay="tableDisplay">
      <tableDisplay>
        <property name="..." />
        ...
      </tableDisplay>
    </objectSet>
  </section>
  <section titleKey="tc_xrt_ComplyingObjects">
    <objectSet source="FND_TraceLink.WorkspaceObject,
                  A4BZ_FND_Tracelink.WorkspaceObject"
              sortdirection="descending"
              sortBy="object_string">
```

```

        defaultdisplay="tableDisplay">
    <tableDisplay>
        <property name="..." />
        ...
    </tableDisplay>
    <listDisplay/>
    </objectSet>
</section>
</page>

```

## Display trace link targets other than ItemRevision in suspect trace link management

By default, users can only see requirement specifications, requirements, and paragraphs in the **Targets** section of the suspect links notifications. To enable viewing of other object types, perform the following:

1. Edit the following XML style sheet:

```
solutions\laws2\businessdata\laws2\install\data\AwpOEPMSignoffSummary.xml
```

2. Locate the following tag:

```

<objectSet source="root_target_attachments.ItemRevision"
  defaultdisplay="listDisplay" sortBy="object_string"
  sortdirection="ascending">

```

3. Change ItemRevision to another object type such as WorkspaceObject:

```

<objectSet source="root_target_attachments.WorkspaceObject"
  defaultdisplay="listDisplay" sortBy="object_string"
  sortdirection="ascending">

```

4. Save the file.

## Create custom properties for trace links

You can optionally create custom property fields that are available when users create trace links.

### Business Modeler IDE configuration

Complete information for configuring your business data model in the Business Modeler IDE is available in the Teamcenter help collection.

1. Create a subclass/subtype of **FIND\_TraceLink** business object type.
2. Add custom properties to the subclass of **FND\_TraceLink**.

3. Add the properties to the createInput tab of the **FND\_TraceLink** business object.
4. Set the custom properties to the following:
  - **Enabled=true**
  - **Modifiable=true**
  - (Optional) **Required=true** or **false**

### Style sheet configuration

1. Create an XML file that defines the style sheet.
2. Create a dataset named *typeNameCreate* of type **XMLRenderingStylesheet** with the XML that you import in the **Select an import file** option in the **New Dataset** dialog.
3. Create the following preference:
  - **AWC\_type.CREATERENDERING**
  - **Category: Active Workspace**
  - **Value:** The name of the dataset that you created (*typeNameCreate*) in Step 2.

## Configuring HTML object templates

### What are HTML object templates?

HTML object templates define and format the properties of requirement revisions and paragraph revisions in the Active Workspace **Documentation** tab for all users.

The templates have text datasets that define the formatting for the requirement revision and paragraph revision objects. The templates also apply to all custom types of requirement revisions and paragraph revisions. The content of the text dataset defines the format. Users can display properties as in the same format as Microsoft Word object templates. The text dataset content of the default template HTML appears as follows:

```
<html>
<b>
{%object_name}
</b>
{%body_text}
</html>
```

The requirement title appears in bold and the paragraph body text appears as plain text. You can modify the formatting with any standard HTML tags.

### Template types

There are three template types:

- |                |   |
|----------------|---|
| <b>default</b> | (Updated by database administrator) Teamcenter Foundation installs the default HTML object template, which applies to all users. The default HTML object template applies to all object types, unless you configure another object template (custom type) or a user creates their own template. |
| <b>custom</b>  | (Updated by database administrator) Custom templates that override the default HTML template by an object type (such as specification, requirement, or paragraph) and applies to all users.   |
| <b>user</b>    | (Updated by end-users) Custom template that overrides the default and custom type templates and applies only to the user that creates this template.  |

### Template priority

Active Workspace applies the *first template available* in the following order:

1. user
2. custom
3. default

### Create a custom HTML object template

You can create an HTML template that customizes the requirement formatting.

1. Navigate to your **Home** folder.
2. Click **More commands ... > New ⊕ > Add**.

The **Add** panel appears.

3. Use the in-context search to find and select **HTML Object Template**.

The **HTML Object Template** fields appear.

4. Enter a template name.
5. (Optional) Select an object type to associate the template with such as requirement or paragraph.

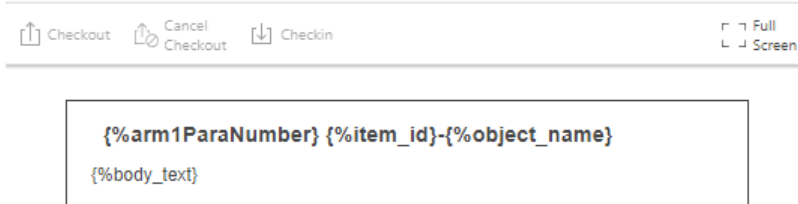
- Click **Add**.



Note:

You can only create one custom HTML template per object type.

A preview of the template appears in the **Overview**.

▼ Preview



- Click the **Documentation** tab, and then click **Edit** .
- Click **Save** .

## Modify default HTML object template

You can modify the default HTML object template that defines and formats the requirement and paragraph properties in the Active Workspace **Documentation** tab. This is not the recommended approach, however. We recommend that you create a custom HTML template. This custom template will override the default template, but you can return to the default template by simply removing the custom template.

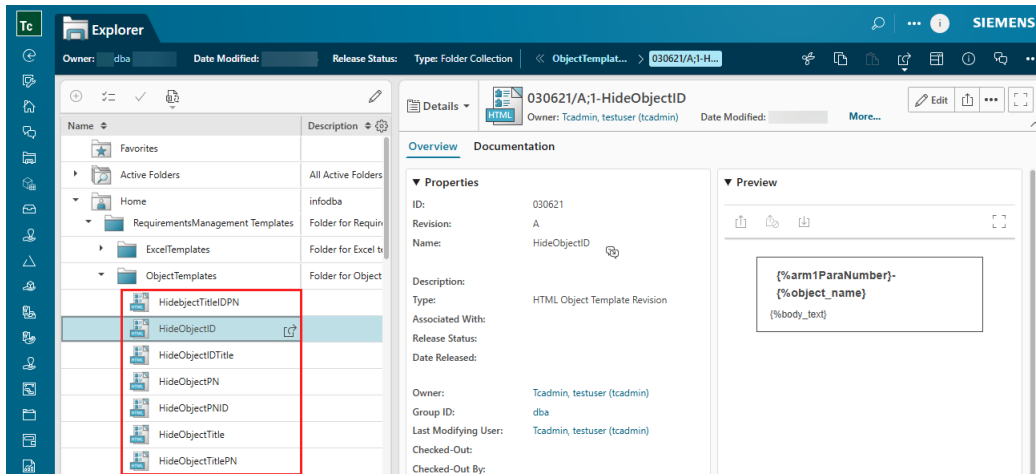
- In the Teamcenter rich client, navigate to **Home\RequirementsManagement Templates\ObjectTemplates**.
- Select the default HTML object template named **RM\_default\_HTML\_object\_template**.
- Click the template object to expand to **RM\_default\_HTML\_object\_template** revision and then to the text dataset associated with the revision.
- Check out the dataset and download the template text named reference file.
- Open the downloaded file with a text editor, modify the HTML formatting, and then save the file.
- Upload the file, and then check the text dataset back in.
- Open a requirement in the Active Workspace **Documentation** tab to verify that your updated HTML formatting is applied.

## Hide object headers using HTML template



You can hide any or all three object headers, the object ID, paragraph number, and the object header name or title in the **Documentation** tab or Word document after export.

Note:

- Teamcenter provides sample out-of-the-box HTML templates.



- Each object type can be defined in only *one* HTML template.

- In the Teamcenter rich client, navigate to **Home\RequirementsManagement Templates\ObjectTemplates**.
- Select a template from the sample out-of-the-box HTML templates depending on which object header you wish to hide.
- Click **Edit** .
- (Optional) Modify the name of the template in the **Name** field.
- In the **Associated With** field, select one or more object types, such as **Requirement** or **Paragraph**, to apply the template to.
- Click **Save** .
- Open a requirement in the Active Workspace **Documentation** tab to verify that the object header or headers are hidden for the selected object types.

## Override default HTML object template for a specific object type

You can create an HTML object template that overrides the default template for a specific object type, such as a requirement revision.

1. In the Teamcenter rich client, navigate to **Home\RequirementsManagement Templates\ObjectTemplates**.
2. Choose **File→New→Item**.
3. In the **New Item: Business Object Type** box, select **HTMLObjectTemplate**, and then click **Next**.
4. In the **New Item: Object Create Information** dialog box, enter an HTML object template **name**, and then click **Next**. You use this template name in step 10 when you create a new object preference, for example, **MyHTMLTemplate**.
5. Accept the defaults for all remaining dialogs by clicking **Next**.

The new HTML object template object appears in the **ObjectTemplates** folder.

6. Select the new HTML object template, and then click to expand to the revision.
7. Expand to the text dataset associated with the revision.
8. Check out the text dataset and download the text template named reference file.
9. Open the text template file with a text editor, modify the HTML formatting, and then save the file.
10. Upload the text file, and then check in the text dataset.
11. Create a preference in the format *Type***HTML\_object\_template** where *Type* is the object type.

Example:

If the object type is a requirement revision, then create the case-sensitive preference **Requirement Revision\_HTML\_object\_template**.

12. Set the value for the new preference to the template you created in step 4.

Example:

Using step 4 as an example, type **MyHTMLTemplate**.

13. Open a requirement that is of the type that you defined the preference for in the Active Workspace **Documentation** tab to verify that your updated HTML formatting is applied.

Example:

Using step 11 as an example, you created a preference for a requirement revision, so you open a requirement revision object.

## Convert requirements created prior to Active Workspace 2.3

You can convert requirements created prior to Active Workspace 2.3, using the `req_word_html_converter` utility.

Note:

To use this utility, install **ReqMgmtWordToHtml Translator**. This translator also converts requirement content imported to Teamcenter or edited and saved in Microsoft Word from Teamcenter (stored as a full-text dataset), so that it can be viewed in the rich text editor in Active Workspace. The translator does not convert Word files (Microsoft Word datasets) that are stored in the system and may be related to requirement objects or other objects. Run Dispatcher so that the translator performs conversions. For more information about the Dispatcher and Translators, see *Configuring Teamcenter* in the Teamcenter collection.

## Selectively convert requirements from HTML format to Microsoft Word format

You can selectively convert requirement content in HTML format to Microsoft Word format (or the reverse, depending on the option selected using the `req_word_html_converter` utility).

## Update custom Word and Excel templates

If you use Microsoft Office 2013 or later, you must update the old custom specification templates and object templates that were created in Microsoft Office 2010. You must update the custom templates because Microsoft has changed the format of specification and object templates. If this is not done, the export to Word and Excel and the compare contents features fail to export the document.

Note:

For supported versions of third-party software, see the Hardware and Software Certifications knowledge base article on Support Center.

Perform one of the following:

- The upgrade runs the **add\_req\_templates** utility and updates the templates.
- Update the standard templates manually:
  1. Directly edit the template in the rich client and save it from Microsoft Office.
  2. Repeat the process for all the standard templates for Word and Excel.
  3. Delete the standard Word and Excel templates (including items) from the rich client and run the following utility at the server prompt:

```
add_req_templates -u=username -p=password -g=dba
```

## Set preferences for requirements

You can set the following Teamcenter preferences for requirements management.

These preferences are specific to requirements management. All preferences are documented in the Teamcenter rich client. For more information about all available preferences, see *Configuring Teamcenter*.

### General requirements management preferences

Set the following preferences that support general requirements management functionality.

#### AWC\_req\_do\_html\_conversion

Converts requirements from Microsoft Word to HTML format for preview only in the **Documentation** tab CKEditor; however, the conversion does not change the content type status from Microsoft Word rich text to HTML.

#### AWC\_req\_viewer\_page\_size *n*

Defines the number of requirements that are loaded together within the Active Workspace **Documentation** tab. Valid values are:

**0**: All requirements in a specification are loaded together.

*n*: The number of specification requirements that are loaded together; the remaining requirements load as consecutive sets of *n*. For example, a value of **50** loads 50 requirements at a time.

#### REQ\_DefaultReqSpecType and REQ\_DefaultReqType

Defines the default requirement specification type and default requirement type. The value added to each preference appears in the **Import Specification** panel as the default **Specification Type** and **Default Requirement Type** drop-down list. In the **Create Specification** panel, the preference appears as the default type. When creating children under a requirement specification, the child object type is displayed based on what is defined in the preference.

#### Note:

If you change the display rules in BMIDE, then you should ensure those display rules are in sync with the preferences **REQ\_DefaultReqSpecType** and **REQ\_DefaultReqType**. For example, if the BMIDE display rules on **RequirementSpec** are hidden, then the preference **REQ\_DefaultReqSpecType** should have a different subtype name instead of **RequirementSpec**.

#### TCAAllowedChildTypes\_RequirementSpec

Specifies the default for the object added after creating another object. For example, when an end user creates a specification, the next object added by default is a requirement. For the **RequirementSpec** type, the preference **TCAAllowedChildTypes\_RequirementSpec** is queried and

the first value in that preference is taken as the preferred type if that type is valid. If that type is not creatable or valid then the next value is taken from the preference. If none of the values are valid then the selected type is taken as the preferred type.

### **REQ\_DashboardSearchTypes**

Defines the available types available on the **Requirements Manager landing page** CONTEXT Type menu.

### **REQ\_allow\_types\_for\_Insert\_from\_Structure**

Enables functionality to copy and paste requirement objects from one structure to another. This includes all types, such as specifications, requirements, and paragraphs.

### **REQ\_ImportSpec\_As\_RAC\_Structure**

Enables the use of keywords as parameters for importing a specification. The imported document will be parsed using specified keywords to create requirements or paragraphs. This preference enables the use of dash characters in paragraph numbers, for example, 2.1.1-1.

### **AWC\_Arm0HeaderFooter.CREATERENDERING**

Defines the creation stylesheet for objects of the type Header Footer template.

### **AWC\_Arm0HtmlSpecTpl.CREATERENDERING**

Defines the creation stylesheet for objects of the type HTML Specification Template.

### **AWC\_Arm0HeaderFooterRevision.SUMMARYRENDERING**

Registration of the summary stylesheet for the Header Footer template.

### **AWC\_Arm0HeaderFooterRevision.INFORENDERING**

Registration of the information stylesheet for the Header Footer template.

### **AWC\_Arm0HtmlSpecTplRevision.showObjectLocation.SUMMARYRENDERING**

Registration of the summary stylesheet for the HTML specification template object.

### **AWC\_Arm0RequirementElement.SUMMARYRENDERING**

Registration of the summary stylesheet for the requirement element.

### **AWC\_TcRMWorkspace.Arm0RequirementElement.SUMMARYRENDERING**

Registration of the summary stylesheet for the Teamcenter Requirements Management workspace requirement element.

### **Arm0RequirementElement.INFORENDERING**

Registration of the information stylesheet for the requirement element.

### **AWC\_Arm0ParagraphElement.SUMMARYRENDERING**

Registration of the summary stylesheet for the paragraph element.

### **AWC\_TcRMWorkspace.Arm0ParagraphElement.SUMMARYRENDERING**

Registration of the summary stylesheet for the Teamcenter Requirements Management workspace paragraph element.

**Arm0ParagraphElement.INFORENDERING**

Registration of the information stylesheet for the paragraph element.

**AWC\_Arm0RequirementSpecElement.SUMMARYRENDERING**

Registration of the summary stylesheet for the requirement specification element.

**AWC\_TcRMWorkspace.Arm0RequirementSpecElement.SUMMARYRENDERING**

Registration of the summary stylesheet for the Teamcenter Requirements Management workspace requirement specification element.

**Arm0RequirementSpecElement.INFORENDERING**

Registration of the information stylesheet for the requirement specification element.

**AWC\_REQ\_Reuse\_URL**

Stores the URL to the Reuse Web API, which can be used to allow Requirement Quality Compliance from Teamcenter. The value is in the format <Protocol>://<Domain>:<Port>/<Site>, for example, https://reusecompany.url.com:1234/siteVersion.

**AWC\_TcRMWorkspace.Requirement****Revision.showObjectLocation.OccurrenceManagementSubLocation.SUMMARYRENDERING**

Registration of the summary stylesheet for the sublocation of the requirement revision.

**REQ\_compare\_properties\_for\_merge**

Defines the list of properties on revision that need to be exported for compare and merging of derived specifications and master specifications. The valid values are property internal names.

**REQ\_DefaultReqSpecType**

Defines the default Requirement Specification type that is to be displayed on the **Create Specification** and **Import Specification** panel.

**REQ\_DefaultReqType**

Defines the default Requirement type that is to be displayed on **Import Specification** panel.

**REQ\_RelationsInAttachmentColForSummaryTab**

Defines the list of relations that need to be queried when showing the **Attachments** column in the **Summary** tab. Valid values include:

- IMAN\_specification
- IMAN\_reference
- IMAN\_manifestation
- IMAN\_Rendering
- TC\_Attaches

- LisOSpecification
- LisOReference
- LrmOMasterRelation
- LrmOAffectedBy
- LrmOConstrainedBy
- LrmOConstrains
- LrmODecomposedBy
- LrmODecomposes
- LrmOElaboratedBy
- LrmOElaborates
- LrmOImplementedBy
- LrmOSatisfiedBy
- LrmOSatisfies
- LrmOTrackedBy
- LrmOValidatedBy

### REQ\_DashboardSearchTypes

Defines the list of types that will be queried on the Requirements Manager dashboard. The valid values are the internal names of the types. Only Requirement related types are supported, such as: RequirementSpec Revision, Requirement Revision, Paragraph Revision, and their subtypes.

### AWC\_req\_viewer\_page\_limit

Defines the maximum limit for number of requirements that load in a single page within the Teamcenter Requirements **Documentation** tab. The **Page Up** and **Page Down** buttons appear for specifications exceeding this limit of requirements. This helps to navigate to the next or previous page. The valid value is represented by **n**, with **n** being an integer.

### REQ\_Matrix\_Settings\_Properties

Controls the display name of objects shown in the row and columns header of a generated Traceability matrix in Teamcenter. Valid values should be a list of keywords in the format <RealTypeName>:<propertyName>. The real type can be any WorkspaceObject subtype. Runtime types are not supported.

### AWC\_show\_documentation\_tab\_for\_types

Defines the list of Item revision types for which the documentation tab will be displayed in the Teamcenter Requirements Content Viewer.

### Import Microsoft Word document preferences

Set the following preferences that support importing requirement specifications in Microsoft Word documents so that users can edit the requirements in the Active Workspace **Documentation** tab with the embedded **CKEditor**. Teamcenter 11.3 or later required for this feature.

#### Import\_Spec\_Info\_EXCLUDETYPES

Defines the list of object types to hide in the **Type** and **Subtype** drop-down lists of the **Import** dialog. For a given value, its associated subtypes are also excluded. Valid values are internal type names.

#### TcAllowedImportExportPeakSpecObject

Specifies the allowed object types that can be used as the top line for importing a document. The format for valid values are *object\_internal\_type\_name:true* or *object\_internal\_type\_name:false*. If the value is **true**, then the specified type and its sub-types are considered as top lines for import and export purposes. If the value is **false**, only the specified type (and not its sub-types) are considered.

#### AWC\_ReqImportAsHtml

Activates the visibility of the **Enable editing** check box on the **Import Specification** panel in Teamcenter.

- |              |   |
|--------------|---|
| <b>true</b>  | User will not be shown the <b>Enable editing</b> check box on the <b>Import Specification</b> panel and the requirements will be imported as HTML. This is the default value. |
| <b>false</b> | User will see the <b>Enable editing</b> check box on the <b>Import Specification</b> panel and will have control to choose HTML or Word during import.                        |

#### REQ\_ImportSpec\_As\_RAC\_Structure

Import Word document to create a structure in the primary work area that is the same in as in Teamcenter rich client.

#### AWC\_ReqImportAsHtml

Activates the visibility of the **Enable editing** check box on the **Import Specification** panel in Teamcenter.

- |              |   |
|--------------|---|
| <b>true</b>  | User will not be shown the <b>Enable editing</b> check box on the <b>Import Specification</b> panel and the requirements will be imported as HTML. This is the default value. |
| <b>false</b> | User will see the <b>Enable editing</b> check box on the <b>Import Specification</b> panel and will have control to choose HTML or Word during import.                        |

#### REQ\_Microservice\_Installed

Enables import preview functionality.

- |              |   |
|--------------|---|
| <b>true</b>  | All microservices-based features are available.   |
| <b>false</b> | This is the default value. The following features are disabled: <ul style="list-style-type: none"> <li>• Preview before import</li> <li>• View change history</li> <li>• Word round-trip export</li> <li>• Export requirement to PDF</li> </ul> |

### Export to Microsoft Word, Microsoft Excel, or PDF preferences

Set the following preferences that support exporting requirements to Microsoft Word file format.

#### **apply\_bom\_crawling**

To enable editing requirement properties in a Microsoft Excel export, locate the **excel\_template\_rules** property, and then select the **Apply BOM Crawling** value to enable this feature. Apply this option to a template and select that template during export. For more information about the **Apply BOM Crawling** property, see *Administering Teamcenter* in the Teamcenter help.

#### **TC\_Enable\_Implicit\_CO**

The **TC\_Enable\_Implicit\_CO** site preference specifies if a user can edit an object without explicitly checking out an object.

Allows different users to simultaneously edit the same object. In such a concurrent editing scenario, when saving the changes, if changes by another user have already been committed, then the **TC\_overwrite\_protection** preference determines if the changes by the current user should overwrite the other user's changes or if the current user must redo the changes.

- |              |   |
|--------------|---|
| <b>true</b>  | Enables users to edit an object without using the <b>Check-out</b> menu.                                  |
| <b>false</b> | Enables users to edit an object by clicking <b>Check-Out</b> and <b>Edit</b> . This is the default value. |

#### **TypeAllowableWordSpecTemplateForExport and TypeAllowableExcelSpecTemplateForExport**

Currently you must set the specification templates for export to Microsoft Excel and Microsoft Word, respectively, each time you want to change them. For some exports, you set a specific template based on the object type that you selected for export. The **TypeAllowableWordSpecTemplateForExport** and **TypeAllowableExcelSpecTemplateForExport** preferences allow you to set default templates for specific object types for export. For example, if you want to use a specification template that is named **MySpecificationTemplate** for the

requirement revision subtype that is named **MyRequirement**, then you create the preference **MyRequirementRevisionAllowableWordSpecTemplateForExport=MySpecificationTemplate**

### REQ\_Export\_WithAddin

Activates the traditional **Edit in Word**, **Export to Excel**, and **Export to Word** in Teamcenter (using Addins). Ensures that each HTML requirement is converted to Word in the background.

- true**            User should be able to use all traditional features like **Edit in Word**, **Export to Excel**, and **Export to Word**.
- false**            User will not be able to use any of the above features and the commands will be disabled. This is the default value.

### REQ\_ShowExportMatrixToPDFCommand

Activates the Export to PDF command within the Teamcenter traceability matrix.

- true**            User should be able to view the PDF command and export the generated traceability matrix in PDF format.
- false**            User will not be able to view the PDF command in the generated traceability matrix. This is the default value.

## Convert requirements from Microsoft Word to HTML preferences

Set the following preferences that support converting Microsoft Word documents to HTML.

### RM\_default\_HTML\_object\_template

Specifies the default HTML object template that the Active Workspace **Documentation** tab uses to render requirement and paragraph object formatting. The default value points to the default HTML object template, which is of the type **HTMLObjectTemplate**.

### REQ\_WordToHTMLTranslatorUsingLoggedInUser

Determines the user profile that the Word to HTML conversion utility uses for translation. This utility converts the requirements in Microsoft Word format into HTML for display in Active Workspace.

When rich text is saved in Microsoft Word, for example, when you import a requirement specification or update a requirement using the **Documentation** tab, a Dispatcher request is created. This request is handled by the **ReqMgmtWordToHtml** translator, which invokes the **req\_migrate\_docm\_to\_html** utility to convert the Microsoft Word format to HTML.

- true**            (Default): user the currently logged-in user.
- false**            Use the **dcproxy** user.

**Note:**

This preference is not applied if you directly run the `req_migrate_docm_to_html` utility or the `req_bulk_upgrade_docm_to_html` utility from the command line.

### Microservices for Requirements

**Note:**

These preferences must be configured after installation for microservices to function.

#### REQ\_microservice\_installed

Enables use of the Requirements microservices, including importing specifications with preview, requirements change history, Microsoft Word round-trip, and exporting requirements to PDF.

- |              |   |
|--------------|---|
| <b>False</b> | (Default): Features are inactive.                               |
| <b>True</b>  | All microservice-based features for requirements are available. |

### Enable full text searches

You can enable full-text searches on requirements.

1. As an administrator user, set the `AWS_FullTextSearch_Index_Dataset_File_Content` preference to **true**.
2. Reindex.

## Configuring system modeling

### General configuration of system modeling

Before using system modeling, you must configure the following:

- Verify that **Systems Modeler** is installed when you run Teamcenter Environment Manager (TEM):
  - The **Systems Modeler** feature is located under **Base Install**→**Active Workspace**→**Client**.
  - The **Systems Modeling** feature is located under **Base Install**→**Active Workspace**→**Server Extensions**.
- Update the following constants using Business Modeler IDE:

- **Awb0SupportsStructure:** Update this global constant with the value **Fnd0SystemModelRevision**. This ensures that the **Content** tab is visible.

```
<TcGlobalConstantAttach constantName="Awb0SupportsStructure"
value=" ">
  <ConstantValue value="Fnd0SystemModelRevision"/>
</TcGlobalConstantAttach>
```

- **Awb0AvailableFor:** Update this type constant with the value **Fnd0SystemModelRevision**. This ensures that the **Architecture** tab is visible.

```
<TcTypeConstantAttach constantName="Awb0AvailableFor"
typeName="Ase0ArchitectureFeature" value="Functionality,
Fnd0LogicalBlock,RequirementSpec,Requirement,Paragraph,
Fnd0SystemModel">
</ TcTypeConstantAttach >
```

- **Awb0BOMArchetypeToOccurrence:** Update this type constant with the value **Fnd0SystemModelRevision**. This ensures that the system model revision is associated with the occurrence.

```
<TcTypeConstantAttach constantName="Awb0BOMArchetypeToOccurrence"
typeName="Ase0LogicalElement" value="Fnd0LogicalBlockRevision,
Fnd0SystemModelRevision"/>
```

- (Optional) Update the values of the following preferences using Teamcenter rich client:
  - **Awb0Element.CellProperties:** Displays the properties of the node in the diagram.
  - **AWBBackgroundContextCleanupDays:** Specifies the maximum age in days for a **BackgroundContext** element. After the specified period, the **BackgroundContext** element is deleted.
  - **AWBBackgroundContextMaxCountPerUser:** Specifies the maximum number of **BackgroundContext** elements that can be created by a user.
  - **AWBBackgroundContextAutoSaveFrequency:** Specifies the frequency in seconds for a **BackgroundContext** element to be saved. If the value is 0 or negative, it is not saved. Also if you refresh the diagram before the save occurs, the diagram is not saved.
  - **AWC\_Relations\_Panel\_Tabs:** Specifies the order in which the tabs are displayed in the **Relations** panel.
  - **SE\_connection\_indicator\_property:** Enable the visual indicator for trace links and connections when the value is set to **true**.

- **AWV0HostAWInVisUponLaunch**: Enables Active Workspace session in Vis when you open a model in Vis. If the value of this preference is **True** the Active Workspace session is enabled. If it is **False**, only the Vis session is enabled.
- When you customize the Relations Legend for system models, you must update the following preferences:
  - **SE\_diagram\_legend\_configuration\_file\_name**: Registers the name of a dataset that stores an XML file containing the **Architecture Modeler** UI configuration. The default dataset name is *ArchitectureModelerLegend*.
  - **SE\_diagram\_presentation\_styles\_file\_name**: Registers the name of a dataset that stores an XML file containing the presentation styles of the shapes. The default dataset name is *ArchitectureModelerPresentationStyles*.
  - **SE\_DiagramViews.Architecture**: Specifies the list of view names for the **Architecture** tab. The view names are defined in Architecture UI configuration file, which is stored in the dataset named by the value of preference **SE\_diagram\_legend\_configuration\_file\_name**.
- Use XML rendering template (XRT) files to configure the layout in Active Workspace based on the object type, user group, and role. You edit XRT files in the rich client. You can locate Active Workspace style sheet preferences in the rich client by choosing **Edit**→**Options**→**Search** and looking for preferences whose names follow this format:

#### **AWC\_<type-name>.\*RENDERING**

XRT files provide descriptions of their use.

- Ensure that you create search prefilters for your System Modeling elements using either the **Awp0BusinessObjectCategories** Business Modeler IDE constant or the **AW\_FullTextSearch\_TypeCategories** preference. For example, you can group the following elements in a category called **Systems** that includes the following elements:
  - **FunctionalityRevision**
  - **Fnd0SystemModelRevision**
  - **Fnd0LogicalBlockRevision**
  - **Fnd0LogicConnRevision**
  - **Requirement\_0\_revision\_alt**
  - **Att0MeasurableAttribute**
  - **Att0TargetRevision**
  - **Awb0SavedBookmark**
- When you collapse a parent element and if the child elements have trace links with elements outside the parent, these trace links end with T-lines instead of straight lines. To configure if the T-lines appear, add the following parameter to the presentation styles XML document:

```
<parameter name="hasRollUpStyle">
  <value>true</value>
</parameter>
```

## Configure how system modeling objects and relations appear in Architecture Modeler

To configure how System Modeling objects and relations appear in the Relation Browser:

1. Log in as a system administrator and check out the dataset that is described in the **SE\_diagram\_legend\_configuration\_file\_name** preference.

By default, the name of this dataset is *ArchitectureModelerLegend* and the name of this named reference file name is *SE\_Legend\_Configuration.xml*.

2. Download the named reference to your local workstation and open it in an XML editor.
  - The XML file has the following elements:

```
<ArchitectureModeler>
  <OptionPanel>
    <view name="RFLP" diagramMode="true">
      <defaultLayout>IncrementalHierarchic</defaultLayout>
      <group name="objects">
        <filter name="Requirement" parameterSet="Requirement"
color="(64,100,142)"/>
      </group>
      <group name="relations">
        <filter name="Connectivity" parameterSet="Connectivity"
color="(221,115,115)"/>
      </group>
      <group name="ports">
        <filter name="Flow" parameterSet="Flow" color="(0,0,0)"/>
      </group>
      <group name="annotations">
        <filter name="Rectangle" parameterSet="Rectangle"
color="(235,235,227)"/>
      </group>
    </view>
  </OptionPanel>
  <parameterSets>
    <parameterSet name="Traceability" scopeFilter="Global"
styleId="TraceabilityStyle">
      <typesForCreation>
        <type>FND_TraceLink</type>
      </typesForCreation>
    </parameterSet>
  </parameterSets>
</ArchitectureModeler>
```

- The Option Panel section contains the following elements:

- **view name:** Specifies the name of the view, for example, **General**.
  - **defaultLayout:** Specifies the default layout of the diagram such as **IncrementalHierarchic**.
  - **group name:** Specifies if this is a list containing objects, relations, ports, or annotations. These lists are presented in the **Relation Controls** section of the Relations Browser.
  - **filter name:** Specifies the name of the relation or object name.
  - **parameterSet:** Specifies a name that matches the **parameterSet** name in the **parameterSets** section. The **parameterSet** defines what styling and relation types to display.
  - **color:** Specifies the display color of the object, relation, ports, or annotations.
- The **parameterSets** section specifies the different types of object, relations to be created for the specified object. It contains the following elements:
    - **name:** Specifies the name of the parameterSet element.
    - **scopeFilter:** This can be either **Global** to represent the occurrence tracelink or **Context** to represent the normal tracelink.
    - **styleID:** Specifies the style to apply. The style is defined in the another XML file. This is the presentation file that defines the styling. The name of this file is defined in the **SE\_diagram\_presentation\_styles\_file\_name** preference.

The ID defined in this element must match the ID in the presentation file.

    - **type:** Specifies the type of relation to be created.

3. Save the additions, upload the Named Reference, and then check in the dataset.

## System modeling preferences

Preference Name	Purpose	Valid Values	Default Values	Default Protection Scope
<b>AWB_default_secondary_page</b>	Defines the default tab to display in work area.	Any feature page string constant.	<b>Fnd0LogicalBlock-Revision:Ase0-ArchitectureFeature</b>	All
<b>AWC_Connection_Type_Categories</b>	Defines the connection PMEI types.	One or more of: <ul style="list-style-type: none"> <li>• P - PhysicalConnRevision</li> <li>• M - MaterialConnRevision</li> <li>• E - EnergyConnRevision</li> <li>• I - InfoConnRevision</li> </ul>	None	Site
<b>TCAAllowedChildTypes_Fnd0LogicalBlock</b>	Defines the revision types that	A list of item revision types that a system	None	Site

Note:  
You must add the **AWC\_Connection\_Type\_Categories** preference.

Preference Name	Purpose	Valid Values	Default Values	Default Protection Scope
	can be included in a system.	can comprise, including connections.		
<b>ASE0_Show_3D_Data_On_Selection</b>	Defines the list of system types that are traversed for trace linked parts to display in the 3D viewer.	BusinessObject types	Ase0LogicalElement Awb0Connection	None
<b>AWC_LaunchToVis_TraverseTypes</b>	Defines the list of system types that are traversed for trace linked parts to display in the 3D viewer.	BusinessObject types	Fnd0LogicalBlockRevision PSConnectionRevision	None

## Enable the Open in SMW menu item

Users can launch System Modeling Workbench (SMW) project models directly in SMW.

The **Open in SMW** command is available for the following project model types:

- Capella library
- Capella configuration project
- Capella project
- Capella project model elements
- Capella project diagrams from **Diagrams**

Additional System Modeling Workbench installation and configuration documentation is available on Support Center.

If you want to enable this menu item, ensure that the following features are installed in Teamcenter:

- *Server*: System Modeling Workbench integration
- *Active Workspace client*: System Modeling integration

Note:

For more information about installing the Active Workspace client, see the *Deployment Center — Usage*.

## Install and configure embedded software

### Install embedded software using Teamcenter Environment Manager

To install embedded software using Teamcenter Environment Manager select the following features.

#### Procedure

1. **Extensions > Mechatronics Process Management > ESM Base**
2. **Extensions > Mechatronics Process Management > ESM Processor**
3. **Extensions > Mechatronics Process Management > ESM Software**
4. **Extensions > Mechatronics Process Management > Embedded Software Design Data Management**
5. **Active Workspace > Server Extensions > Embedded Software Management For Active Workspace**

### Install embedded software using Deployment Center

To install embedded software using Deployment Center select the following applications.

#### Procedure

1. **Teamcenter > Active Workspace > Embedded Software Management**
2. **Teamcenter > Mechatronics Process Management > Embedded Software Design Data Management**

## Configure and customize embedded software

You can configure what child-types are allowed for embedded software by updating certain preferences.

You can also customize the linking to external applications using Linked Data Framework.

### Install the Collaboration Service

You can install the collaboration service to support concurrent and collaborative multi-user authoring, commenting, and editing on the same requirement specification.

See [Collaborate on requirements](#) for information about using the collaboration service.

To allow collaborative editing of the requirements, Siemens integrates with CKEditor to use their collaborative editing capabilities: <https://ckeditor.com/collaboration/>. The CK plugin embedded within the Requirements Management Active Workspace Client communicates with the collaboration service.

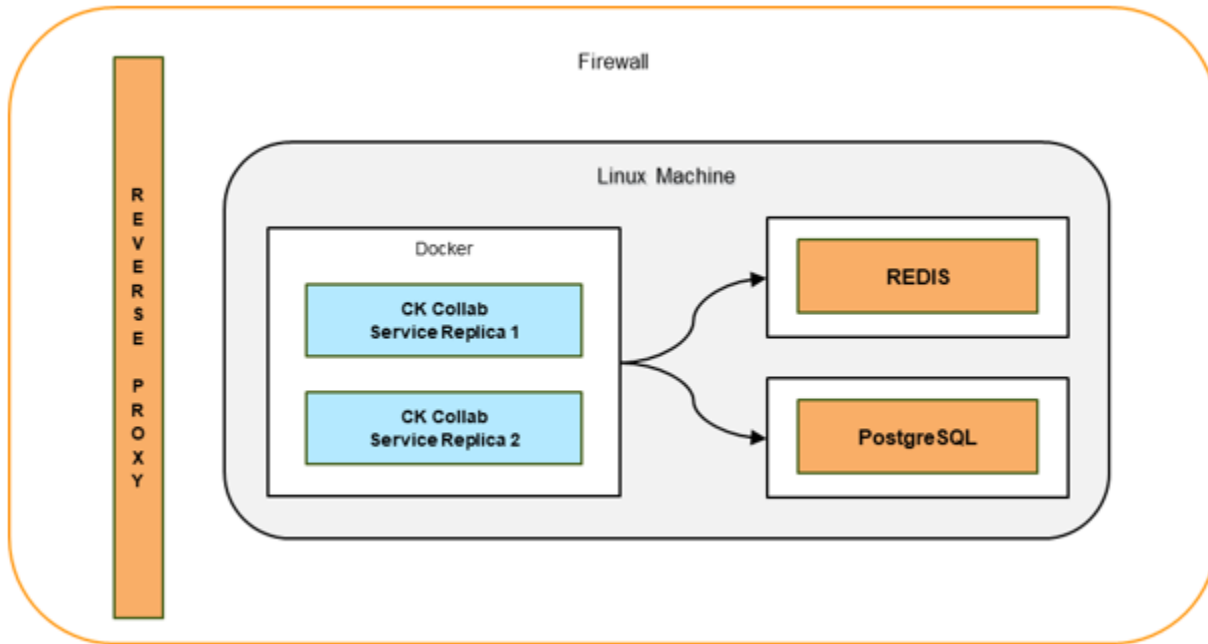
Siemens recommends the following components to achieve a secure setup:

- Firewall: To secure all inbound and outbound communication from the collaboration service.
- Reverse proxy: To set up SSL certificates that allow an HTTPS communication from the Teamcenter server.
- Docker orchestrator: To allow multiple instances of collaboration service and provide load balancing between them.

To allow WebSocket communication to the CK collaboration service, the following ports must be opened on the Reverse Proxy as well as the Docker orchestrator:

- Communication protocol: WebSocket
- Port: 8000
- Ports to be configured at: Reverse proxy and Docker orchestrator
- Description: The CK Collaboration service communicates with the CK Editor Web browser plugin through WebSocket. This port can be changed from Deployment Center.

This image shows a recommended setup of the collaboration service and its prerequisites.



## Restrictions and limitations

The collaboration service is supported only on Linux machines for an on-premise setup.

## Prerequisites

The following table lists the prerequisites for installing the collaboration service.

System	Recommended version	Recommendations
Linux Machines	Any Linux machine that supports a Docker orchestrator. Customers may use existing Linux machines.	All components may be installed on a single Linux machine. Linux machine hardening guidelines may be found here: <a href="https://www.pluralsight.com/blog/it-ops/linux-hardening-secure-server-checklist">https://www.pluralsight.com/blog/it-ops/linux-hardening-secure-server-checklist</a> .
Redis	3.2.6 or newer	
PostgreSQL	12.0 or newer	Customers may use an existing setup. A new user is created as part of this setup.
Docker Orchestrator	Docker Swarm or Kubernetes	Docker orchestrator is needed to perform the load balancing between the service replicas.

System	Recommended version	Recommendations
Reverse Proxy	Nginx or Apache	Reverse Proxy is recommended to install the SSL certificates that allow HTTPS communication from the Teamcenter server.
Firewall		Firewall is recommended to secure all communication to and from the CK collaboration service.

**Note:**

You must have a **req\_realtime\_collab** license for real-time co-authoring of Teamcenter requirements

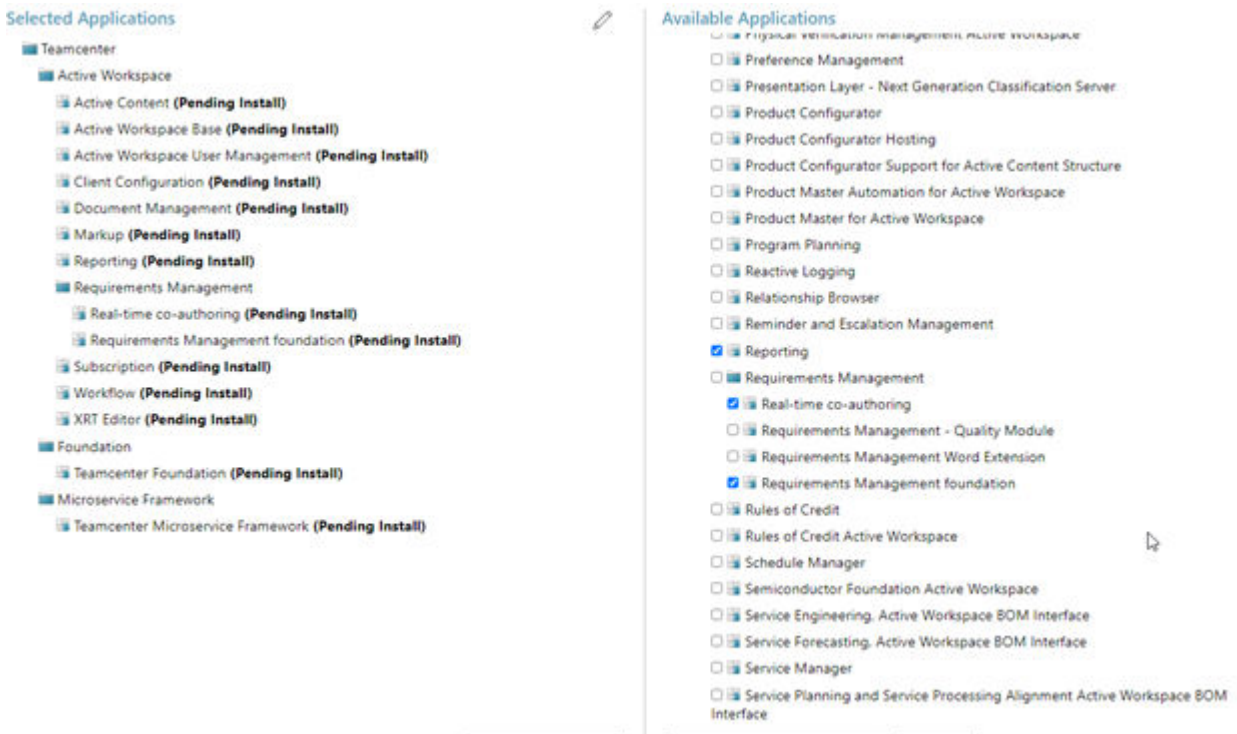
**Procedure**

1. Log on to Deployment Center.

See *Deployment Center — Usage* for information about using Deployment Center.

2. Under Requirements Management in the list of available applications, select the check boxes for **Real-time co-authoring** and **Requirements Management foundation**.

The **Real-time co-authoring** dialog box appears.



### 3. Complete these steps:

- Enter the Linux machine on which the docker orchestrator is installed in the **Machine** field.
- Select the operating system in the **OS** field.
- Enter the number of collaborative editing docker instances to run in the **Number of Replicas** field. The default value is 2.
- Enter the port number in the **Port** field.
- Enter the **Cloud Service Management-Dashboard secret** and then confirm.
- Enter the URL of the CK collaborative service admin page in the **Dashboard URL** field.
- Select a **Database Server** from the list.
- Enter the host machine on which the database is installed in the **Database Host** field.
- Enter the database port number in the **Port** field.
- Enter the name of the database in the **System Database Name** field.
- Enter the database user name in the **User** field.

- Enter the password for the database user in the **User Password** field and then confirm.
- Enter the host machine on which Redis is installed in the **Redis Host** field.
- Enter the Redis port number in the **Port** field.
- Enter the **User Password** and then confirm.
- Select a **Reverse Proxy Protocol** from the list. The default value is HTTPS.
- Enter the **TLS certificate file path**.
- In the **Reverse Proxy URL** field, enter the URL of the Reverse Proxy which is already installed .

4. Click **Save Component Settings**.

## Postrequisites

Once installation of the collaboration service is complete, you must run the following command:

```
docket stack
```

```
docker stack deploy -c req-ck-collab-service.yml CK
```

To enable the collaboration service, you must also run this command, using the IP address of the machine on which the collaboration server is installed:

```
base url = Linux M/c ip
```

```
add_ckeditor_access_keys -u=*** -p=xxxx -g=*** -baseurl=http://  
10.134.208.209:8000 -dashboardsecret=****
```

## Configuring targets

### Modify the target type color

You can add a new element class to the legend and diagram or customize how an existing element class is displayed. The following example describes how to change the default color of targets from gray to dark red. **Target** is a subtype of **Requirement** and displays in the same format unless configured.

1. In the **SE\_Legend\_Configuration.xml** file make the following edits:

```
<parameterSet name="Target">  
  <typesForCreation>
```

```
<type>Att0Target</type>
</parameterSet>
```

and

```
<filter name="Target" parameterSet="Target" color="(185, 90, 80)"/>
```

2. In the **SE\_Presentation\_Rules.xml** file make the following edits:

```
<PresentationRule type="node" styleId="TargetStyle">
  <Conditions operator="and">
    <Type value="Att0TargetRevision" operator="typeOf" />
  </Conditions>
</PresentationRule>
```

No changes are necessary to the **SE\_Presentation\_Styles.xml** file.

## Installing and configuring parameter management


### Installing parameter management

To install the parameter management solution, run Teamcenter Environment Manager (TEM) and choose the following options:

- Under **Extensions** → **Systems Driven Product Development**, choose **Parameter Management**. TEM automatically selects **Attribute and Parameter Base Definitions** when you do this.
- Under **Base Install** → **Active Workspace** → **Server Extensions**, choose **Active Workspace** and **Active Content Structure**.
- Under **Extensions**, choose **Engineering Views**, then choose **Engineering Views Active Workspace**.
- Under **Base Install** → **Active Workspace** → **Server Extensions** → **Systems Driven Product Development**, choose **Parameter Management Active Workspace**. TEM automatically selects **Measurable Attributes and Targets** when you do this.
- Under **Base Install** → **Active Workspace** → **Client**, choose **Active Workspace Client**, and then choose **Workflow, Active Content, and Engineering Views**.
- Under **Base Install** → **Active Workspace** → **Client** → **Systems Driven Product Development**, choose **Parameter Management**.

### Set preferences for parameter management

Before using parameter management, you should review and, if necessary, set the following preferences. See *Working with preferences in Active Workspace* for information on setting preferences.

For more information about a particular preference, search for it using the **PREFERENCES**  tile on your home page.

### PLE\_Map\_Parameters\_With\_Matched\_DataType

Defines the behavior for mapping of the parameters based on data type. If **true**, only parameters with the same data type can be mapped; if **false**, parameters with different data types can be mapped.

### PLE\_Parameter\_Create\_With\_Default\_Usage

Defines the default **Usage** for new parameter creation. If you are importing parameters from Microsoft Excel and the **Usage** is not specified, then the new parameter will have this default usage:

- When using the **Create** (parameter) dialog.
- When creating new parameters in the parameter **Quick Add** feature.

using the **Create** dialog and it will show as the “Starting usage” when creating new parameters in the parameter **Quick Add** feature.

### PLE\_Parameter\_Create\_With\_Definition\_Ux

Defines the requirement for having a parameter definition in order to create a parameter.

The default value for this preference is **false**, indicating that a parameter definition is not required when creating a parameter. Changing this value to true indicates that a parameter definition is required to create a parameter. See **Make parameter definitions required** for information.

## Configure Excel import and export

To configure the import and export of parameters using Microsoft Excel, you can do the following:

- Configure the **Export** panel display such that only relevant templates for parameters are shown as follows.
  1. Create a new Excel template from the existing templates, **Parameter\_template** for parameters or **ParameterDefinition\_template** for parameter definitions.
 

**Name** and **Parameter Definition** are mandatory properties for parameters or **Name** and **Data Type** for parameter definitions.
  2. Add or remove columns as required.
  3. Map new columns to parameter properties by setting the internal property name in row 5.

For example, for the **Name** column, the internal name is **Att0MeasurableAttribute#object\_name** for parameters or **Att0AttributeDefRevision#object\_name** for parameter definitions.

4. Add the Excel template into Teamcenter by executing the following commands:

```
add_req_templates -u=Tc-admin-user -p=password -g=group
-i=<file_name>.xlsm -t=ExcelTemplate
set_excel_template_rules -u=Tc-admin-user -p=password -g=group
-templates==< file_name > -rules=apply_packing,parameter_templates
```

- Customize the parameter import XLS template for your data as follows:

1. Create a new Excel template from the existing templates, **Parameter\_import\_template** for parameters or **ParameterDefinition\_import\_template** for parameter definitions.

**Name** and **Parameter Definition** are mandatory properties for parameters or **Name** and **Data Type** for parameter definitions.

2. Add or remove columns as required.
3. Map new columns to parameter properties by setting the internal property names in row 5.

For example, for the **Name** column, the internal name is **Att0MeasurableAttribute#object\_name** for parameters or **Att0AttributeDefRevision#object\_name** for parameter definitions.

4. Create a new **MS Excel** dataset in Teamcenter and attach the Excel template to it.
5. Update the relevant preference with the dataset name, either **PLE\_Parameter\_Import\_Excel\_Template** for parameters or **PLE\_Parameter\_Definition\_Import\_Excel\_Template** for parameter definitions.

## Legacy parameter project and group migration utility

The legacy parameter project **Att0ParamProject** and parameter group **Att0ParamGroup** were workspace-based objects that you cannot revise. Starting in Teamcenter 13.3 we introduced the item-based parameter project **Att0ParameterPrj/Att0ParameterPrjRevision** to replace the legacy project **Att0ParamProject**.

This utility migrates the legacy parameter project and groups to the corresponding new item-based project and groups.

### Syntax

```
migrate_parameter_projects -u user -p=password -g=group -log=log_file_path
```

### Arguments

**user**

Defines the user with admin rights to the legacy project/group database.

#### password

Provides the database password.

#### group

Identifies the group to which **user** belongs.

#### log\_file\_path

(Optional) Identifies the path to which the utility generates the log file. If you do not specify a path, the utility generates the log file in the default temp folder of not mentioned explicitly.

Consider the following:

- The utility moves the existing Product Configurator from the legacy parameter project to the new item-based project. However, the utility does not migrate the previously applied configuration. If necessary, reset the configuration in ACE and create a saved bookmark.
- The utility migrates the assigned variant configuration to the parameters under the parameter groups. However, the utility does not migrate the variant configurations to the parameters under the parameter project directly.
- The utility does not migrate custom properties.

#### Example

```
migrate_parameter_projects -u=dbauser -p=pw_dbauser -g=dba
-log=C:\temp\migrate_parameter_projects.log
```

## Create or update complex multidimensional parameter value with SOA

To create or update the **complex multidimensional parameter values**, users can consume the SOA:

```
Att0::Soa::AttrTargetMgmt::_2020_12::AttributeTargetManagement::updateParameters2()
```

For more information about SOA, refer to the SOA usage described in the BMIDE.

## Configuring units of measure

### Display sets for units of measure

An administrator can create a unit of measure display set so that units are displayed using the preferred system, rather than the as-authored unit.

## Base units

The unit management system (UMS) defines many values for each unit of measure (UoM). Whether it is metric or non-metric, the type of unit, if it is considered a base unit, and if not, what is the conversion factor from the base unit. All of this information allows **Parameter Management** in Teamcenter to convert between the various units. For example, it knows that 1 inch is equal to 2.54 centimeters.

Following are some examples of length units. There must be one base unit for each system of measurement (metric and non-metric). Within a measurement system, All other units are based upon the base units. To enable conversion between metric and non-metric, all conversion values must be based on just one of the base units.

System	Name	Base Unit?	Conversion
metric	meter	no	1000
metric	centimeter	no	100
metric	millimeter	yes	1
non-metric	feet	yes	304.8
non-metric	inches	no	25.4
non-metric	yard	no	914.4

## Display sets

A *display set* is a list of a preferred unit for each type of unit. For example, the *metric* display set uses *Millimeters* (mm) for length. And the *non-metric* display set uses *Feet* (ft). Common unit types are length, area, volume, mass, temperature, force, energy, and so on.

For consistency, your users can set their preferred display set by managing their user profile settings in Active Workspace, which sets the **UMS\_PREFERRED\_Display\_Unit\_Set** user preference.

When using **Parameter Management**, Teamcenter will convert each type of unit from the unit as-authored to the preferred unit of the display set.

## Provided display sets

Teamcenter's unit management system provides a long list of predefined units of measure, both metric and non-metric. There are many units defined for each type of unit, like length for example. Three display sets are provided to help organize these units for the user. These display sets are **Metric**, **Metric(SI)**, and **Non-Metric**.

Following are several unit types and the preferred unit for each display set.

Unit type	Metric	Metric (SI)	Non-Metric
Length	m	m	ft
Area	mm <sup>2</sup>	m <sup>2</sup>	ft <sup>2</sup>
Capacitance	F	mF	μF
Temperature	°C	°C	°F

## Custom display set

You can define additional display unit sets for your users.

Note:

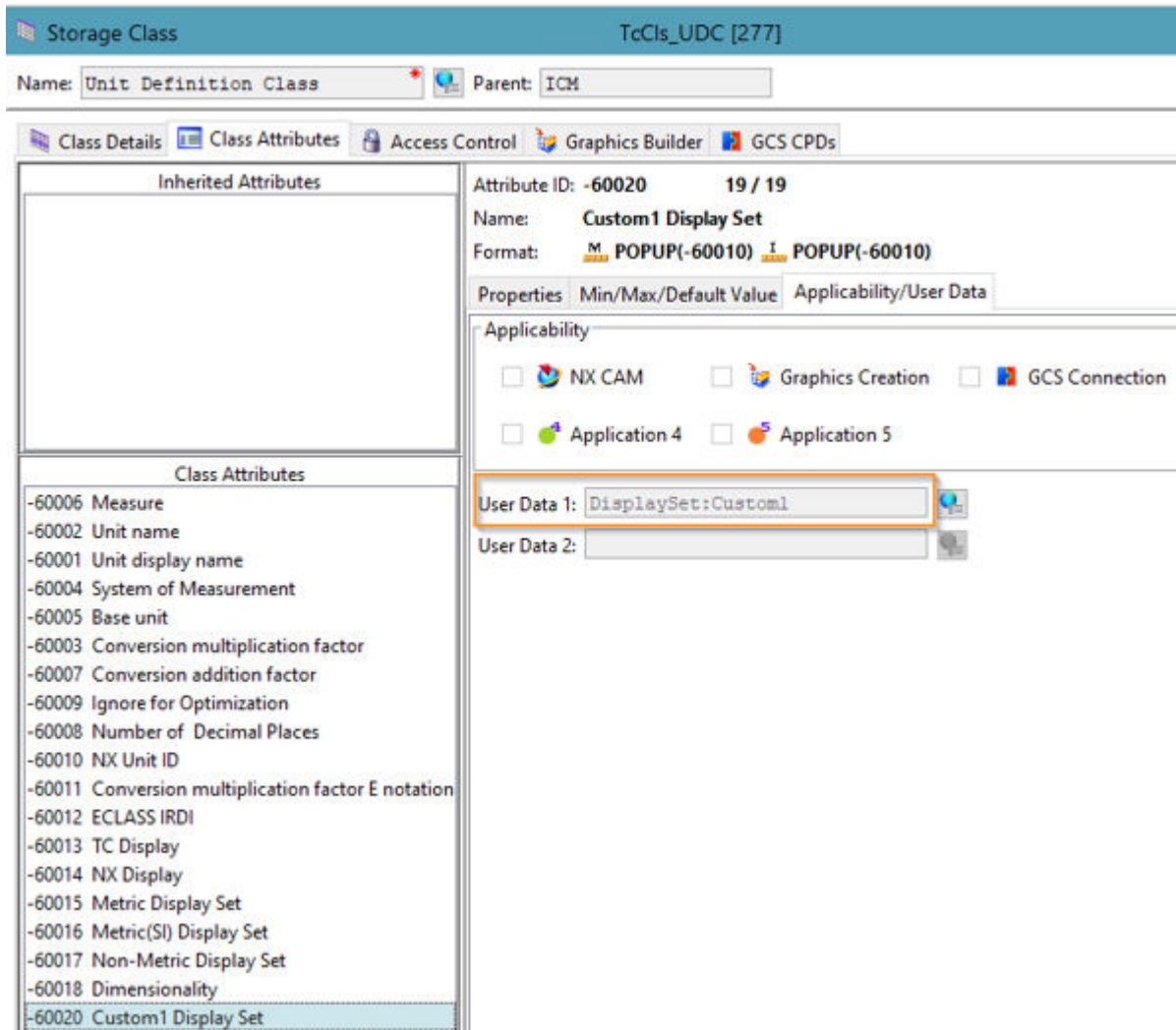
When working with the unit of measure definitions, unit types are called **Quantity**.

UMS stores all of the unit definitions using the Classification system. The attributes of the class align with the columns of the CSV file that is imported. To create a custom display set, you must create a new column in the CSV file. To contain your new column, you must first create a new attribute in the class.

Tip:

Remember to make a backup copy of your CSV file before making any changes.

1. Create a new attribute dictionary definition to the **Unit Definition Class** using Classification Admin in the rich client to store your new column.
  - Create a new attribute in the Classification dictionary named *Custom1 Display Set* for example.
  - Define the format as **POPUP(-60010)** and default value as **EMPTY**.
  - Add the new attribute to the **Unit Definition Class**.
  - Change the **User Data 1** field value to **DisplaySet:Custom1**, where *Custom1* is the internal name of your custom display set.



2. Add a new column named *Custom1 Display Set* to your CSV file.
3. Populate your new column with **Y** or **N** for each unit. Remember to only have one **Y** for each **Quantity** (unit type), and save the file.

A	B	C	D	E	F	O	P	Q	R
ID	Master	Quantity	Unit name	Symbol	System	Metric Display Set	Metric(SI) Display Set	Non-Metric Display Set	Custom1 Display Set
Length_A	Y	Length	Angstrom	Å	m	N	N	N	N
Length_Bc	Y	Length	Barleycorn	Bc	nm	N	N	N	N
Length_ch	Y	Length	Chain	ch	nm	N	N	N	N
Length_ChFr	Y	Length	French	Ch (Fr)	m	N	N	N	N
Length_cm	Y	Length	Centimeters	cm	m	N	N	N	Y
Length_dam	Y	Length	Dekameter	dam	m	N	N	N	N
Length_dm	Y	Length	DeciMeter	dm	m	N	N	N	N
Length_fm	Y	Length	Femtometer	fm	m	N	N	N	N
Length_ft	Y	Length	Feet	ft	nm	N	N	Y	N
Length_fur	Y	Length	Furlong	fur	nm	N	N	N	N
Length_G	Y	Length	Gauge	ga	nm	N	N	N	N
Length_h	Y	Length	Hand	hh	nm	N	N	N	N
Length_hm	Y	Length	HectoMeter	hm	m	N	N	N	N
Length_in	Y	Length	Inches	in	nm	N	N	N	N
Length_km	Y	Length	Kilometers	km	m	N	N	N	N
Length_m	Y	Length	Meters	m	m	Y	Y	N	N
Length_mi	Y	Length	Mile	mi	nm	N	N	N	N

- Run the `ums_import_unit_definitions` utility with the `-allowOverwrite` flag to update the display units.

## Tool-specific unit ID

Use the unit management system (UMS) spread sheet to define the unit ID of your new units of measure.

### NX Unit ID

UMS works with Classification's "NX Unit ID" attribute as described in managing units of measurement. You can configure the unit's value in the `unit_definitions` file and then run the `ums_import_unit_definitions` utility with the `-allowOverwrite` flag to update the display units.

### Custom alias unit ID

You can add a tool-specific unit alias ID for another tool, like HEEDS for example.

- Create a new attribute dictionary definition to the **Unit Definition Class** using Classification Admin in the rich client to store your new column.
  - Create a new attribute in the Classification dictionary named *HEEDS Unit ID* for example.
  - Define the format as **STRING(60)** and default value as **EMPTY**.
  - Add the new attribute to the **Unit Definition Class**.
  - Change the **User Data 1** field value to **Alias:HEEDS**.
- Add a new column named **HEEDS Unit ID** to your `unit_definitions` file and add your values.

3. Run the `ums_import_unit_definitions` utility with the `-allowOverwrite` flag to update the display units.

## Configuring test management

### Install Teamcenter server features, Active Workspace server extensions, and Active Workspace client features

Use Teamcenter Environment Manager to install the following items that support test management:

#### Teamcenter server features

Select the following installation options:

**Note:**

Features marked with an asterisk (\*) are not required for the Teamcenter Test Management Solution (which does not include physical features).

- Extensions
  - Engineering Views
  - Engineering Views Active Workspace
  - Maintenance Repair and Overhaul (\*)
    - MRO Core (\*)
    - As-Maintained Management (\*)
  - Systems Driven Product Development
    - Attribute and Parameter Base Definitions
    - Parameter Management
    - Teamcenter Test Management
    - Physical Verification Management (\*)
  - Systems Engineering and Requirements Management
    - Systems Engineering Base
    - Systems Engineering

#### Active Workspace server extensions

Select the following installation options:

- Server Extensions
  - Active Workspace

- Part Manufacturing Active Workspace (\*)
- Requirements Management
- Systems Engineering
- Systems Modeling
- MRO (\*)
  - MRO Core (\*)
  - As-Maintained (\*)
- Systems Driven Product Development
  - Physical Verification Management Active Workspace Extension (\*)
  - Measurable Attributes and Targets
  - Parameter Management Active Workspace
  - Teamcenter Test Management
  - Physical Verification Management (\*)

### Active Workspace client features

Select the following installation options:

- Client
  - Part Manufacturing (\*)
  - Architecture Modeler
  - Engineering Views
  - Requirements Management
  - Systems Modeler
- Systems Driven Product Development
  - Parameter Management
  - Teamcenter Test Management
  - Physical Verification Management (\*)

## General test management configuration

Before using test management, you must configure the following:

- Run the Teamcenter Environment Manager (TEM) and select the **Verification and Validation Planning and Reporting** server and client extensions under **Systems Driven Product Development**.

This selection also automatically installs the **Attribute and Parameter Base Definitions** and **Measurable Attributes and Targets** extensions.

Note:

**Measurable Attributes and Targets** is also required if you install Systems Modeler.

- Set the required test statuses on the **PLE\_EnabledDefnStatusList** preference.
- Set the **PLE\_Display\_AnalysisRequestPlanTable** preference to turn the plan table on or off. The default is off.
- Define deep copy rules used when revising tests in the **CtrlValidationContract** template.
- Create a workflow that applies an approved status to a completed attribute definition, which makes the definition available for general use. Optionally, you can add review and signoff tasks that must be performed before the attribute definition is released.

A default test approval workflow is provided and you can use this as the basis of your workflow. You must add the necessary participants, ensuring that the design engineer who creates the workflow and the test engineer are members of the same group. The test engineer must also be added as a participant to each study that is created for a test.

- Set the release status that allows an attribute definition to be consumed on the Teamcenter **PLE\_EnableDefnStatusList** preference.
- Configure if parameters are independent in test management with the following preferences:
  - **PLE\_AnalysisRequest\_CopyAttribute**. If set to **false**, no copy is made automatically, but a copy is made if the user edits an input or output. If set to **true**, the system copies all attributes, that is, both inputs and outputs. The **true** option is slower when creating the test, but more efficient when users are editing parameter values in any of the test objects. The default value is **true**.
  - **PLE\_AnalysisRequest\_CopyAttribute\_Options**. If set to **Input**, the system automatically copies the input attributes, but not the outputs. If set to **Output**, the system automatically copies the output attributes, but not the inputs. The default is blank, which enables copying of both inputs and outputs.
- Enable or disable the editing of the domain bookmark to remove associated components.

When they define a test, users may release it by submitting it to the appropriate workflow. By default, once a request is released, you can no longer make changes to the contents of the test. However, you can remove trace links associated with the request, effectively removing the associated components. Likewise, you can edit the domain bookmark to add or remove components. If you want to change this default behavior, change the **Fnd0PreventTracelinkDelete** constant to **Primary** to disallow such removals.

- (Optional) Create a workflow to release a test. You can also use the standard test workflow if it meets your company's needs.
- (Optional) When you have Microsoft Office Online installed, you can view and update the **Goal** file attached to a measurable attribute in the **Viewer** panel of the **Overview** tab in Microsoft Excel. To permit this, you should add **SEARCHORDER.Att0MeasureableAttribute=Att0HasGoalFile** to the

**AWC\_defaultViewerConfig.VIEWERCONFIG** preference. (With this configuration, the **Viewer** panel replaces the **Preview** panel.

- Configure the **PLE\_Plan\_Table\_Allowed\_Child\_Types** preference with the list of object types that show the **Test Results** tab, for example, **PSConnection**, **Interfaces**, **Requirement Revision**, **IAV0TestCaseRevision**, and **Fnd0SEBlockRevision**. If appropriate, add **Requirement Spec** to the list of types.
- Configure the **PLE\_AddObjectsNotAllowedTypes** preference, which defines the list of types that cannot be included in tests. You can add Teamcenter business objects to this preference to allow users to include those objects in tests.

## Configuring global parameters

- When creating parameters to be consumed by NX, ensure you use a unit of measure that is recognized by NX. If this is not possible, leave the **Units** field blank when creating the parameter definition.
- The transfer mode **Parameter\_in\_product\_PLMXML\_TM** to export parameters from a CAE object into a PLMXML file.
- When you copy parameters, configure copying of attributes by setting the following preferences:

- **PLE\_AnalysisRequest\_CopyAttribute**

If set to **false**, no copy is made automatically, and a copy is made only when the user edits an input or output. If set to **true**, the system copies all attributes, that is, both inputs and outputs.

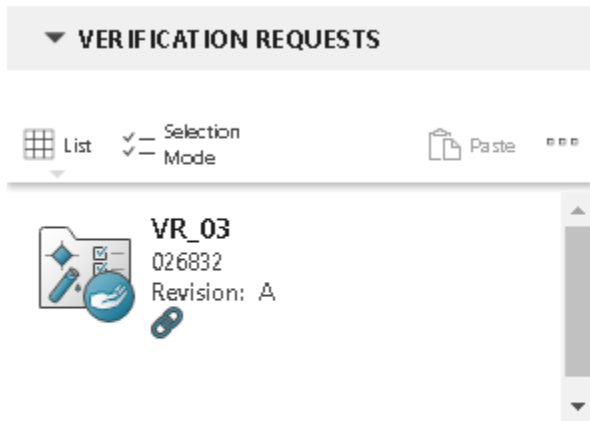
- **PLE\_AnalysisRequest\_CopyAttribute\_Options**

If set to **Input**, the system automatically copies the inputs, but not the outputs. If set to **Output**, the system automatically copies the output attributes, but not the inputs.

By default, **PLE\_AnalysisRequest\_CopyAttribute** is set to **true** and **PLE\_AnalysisRequest\_CopyAttribute\_Options** is blank.

## Configure Where Used tab to show tests

To view the validation requests associated with any workspace object under the **Where Used** tab, you must inject a **WhereUsedAnalyses** section in the required summary stylesheet of (for example) **WorkspaceObject** or **ItemRevision**. The result will be similar to the following example:



Only objects of the **Crt0VerificationRequest** type are displayed, Objects of the **Crt0StudyRevision** type and its subtypes are not displayed.

### Inject method

- Stylesheet: Object Summary
- Under Page: `<page titleKey="web_whereused">`
- Inject tag:

```

<!--
//
=====
// The following column defines the display for when SDPD - Analysis
Request solution is installed.
// If you have not installed this option, this column should be
removed.
//
=====
-->
<inject type="dataset" src="WhereUsedAnalyses" />

```

### Standard objects for which WhereUsedAnalyses is available

```

ase0activeworkspacesyseng\data\AseOPSCConnectionRevisionSummary.xml
att1attrtargetmgmtaw\Att0MeasurableAttributeBoolSummary.xml
att1attrtargetmgmtaw\Att0MeasurableAttributeBoolSummaryForShowObjectLocation.xml
att1attrtargetmgmtaw\Att0MeasurableAttributeDbfSummary.xml

```

att1attrtargetmgmtaw\Att0MeasurableAttributeDbISummaryForShowObjectLocation.xml  
 att1attrtargetmgmtaw\Att0MeasurableAttributeIntSummary.xml  
 att1attrtargetmgmtaw\Att0MeasurableAttributeIntSummaryForShowObjectLocation.xml  
 att1attrtargetmgmtaw\Att0MeasurableAttributePntSummary.xml  
 att1attrtargetmgmtaw\Att0MeasurableAttributePntSummaryForShowObjectLocation.xml  
 att1attrtargetmgmtaw\Att0MeasurableAttributeStrSummary.xml  
 att1attrtargetmgmtaw\Att0MeasurableAttributeStrSummaryForShowObjectLocation.xml  
 aws2\data\Awp0Fnd0LogicalBlockRevisionSummary.xml  
 aws2\data\Awp0Fnd0LogicalBlockRevisionSummaryForShowObjectLocation.xml  
 aws2\data\Awp0Fnd0SystemModelRevisionSummary.xml  
 aws2\data\Awp0Fnd0SystemModelRevisionSummaryForShowObjectLocation.xml  
 aws2\data\Awp0FunctionalityRevisionSummaryForShowObjectLocation.xml  
 aws2\data\requirements\Awp0ParagraphRevisionSummary.xml  
 aws2\data\requirements\Awp0ParagraphRevisionSummaryForShowObjectLocation.xml  
 aws2\data\requirements\Awp0RequirementRevisionSummary.xml  
 aws2\data\requirements\Awp0RequirementRevisionSummaryForShowObjectLocation.xml  
 aws2\data\requirements\Awp0RequirementSpecRevisionSummary.xml  
 aws2\data\requirements\Awp0RequirementSpecRevisionSummaryForShowObjectLocation.xml  
 cdm1awcontractmanagement\data\xrts\Cdm1ContractRevisionSummary.xml  
 cdm1awcontractmanagement\data\xrts\Cdm1ContractRevisionSummaryForShowObjectLocation.xml  
 cdm1awcontractmanagement\data\xrts\Cdm1CorrespondenceRevisionSummary.xml  
 cdm1awcontractmanagement\data\xrts\Cdm1SubmittalRevisionSummary.xml  
 ctm1contmgmtaw\data\AWC\_DC\_TopicRevisionSummary.xml  
 ctm1contmgmtaw\data\AWC\_DCT\_GrphcTrnsItnRevisionSummary.xml  
 ctm1contmgmtaw\data\AWC\_GraphicOptionRevisionSummary.xml  
 ctm1contmgmtaw\data\AWC\_GraphicRevisionSummary.xml  
 ctm1contmgmtaw\data\CmsSummaryForShowObjectLocation.xml  
 eda1edaserveraw\data\EDA1EDACCABaseRevisionSummary.xml  
 eda1edaserveraw\data\EDA1EDACCABaseRevisionSummaryForShowObjectLocation.xml  
 eda1edaserveraw\data\EDA1EDACompRevisionSummary.xml  
 eda1edaserveraw\data\EDA1EDACompRevisionSummaryForShowObjectLocation.xml  
 eda1edaserveraw\data\EDA1EDASchemRevisionSummary.xml  
 eda1edaserveraw\data\EDA1EDASchemRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\AppSoftwareRevisionSummary.xml  
 esw1esmgmtaw\data\AppSoftwareRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\CalibrationRevisionSummary.xml  
 esw1esmgmtaw\data\CalibrationRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\ConfigFileRevisionSummary.xml  
 esw1esmgmtaw\data\ConfigFileRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\Ess0LicenseRevisionSummary.xml  
 esw1esmgmtaw\data\Ess0LicenseRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\Esw0SwArchCompRevisionSummary.xml  
 esw1esmgmtaw\data\Esw0SwArchCompRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\PriBootloaderRevisionSummary.xml  
 esw1esmgmtaw\data\PriBootloaderRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\ProcessorRevisionSummary.xml  
 esw1esmgmtaw\data\ProcessorRevisionSummaryForShowObjectLocation.xml  
 esw1esmgmtaw\data\SecBootloaderRevisionSummary.xml  
 esw1esmgmtaw\data\SecBootloaderRevisionSummaryForShowObjectLocation.xml

```
esw1esmgmtaw\data\SoftwareRevisionSummary.xml
esw1esmgmtaw\data\SoftwareRevisionSummaryForShowObjectLocation.xml
fsh1awfinishmanagement\data\xrts\Fsh1FinishGroupRevisionForShowObjectLocation.xml
fsh1awfinishmanagement\data\xrts\Fsh1FinishGroupRevisionSummary.xml
fsh1awfinishmanagement\data\xrts\Fsh1FinishRevisionForShowObjectLocation.xml
fsh1awfinishmanagement\data\xrts\Fsh1FinishRevisionSummary.xml
iav1verificationmanagement\IAV1MeasureReqmtRevisionSummary.xml
iav1verificationmanagement\IAV1VerifReqmtRevisionSummary.xml
iav1verificationmanagement\IAV1VerifReqmtRevisionSummaryForShowObjectLocation.xml
wpm1awpkgmgmt\data\xrts\Wpm1WorkPackageRevisionSummary.xml
wpm1awpkgmgmt\data\xrts\Wpm1WorkPackageRevisionSummaryForShowObjectLocation.xml
```

### Migrate test objects

You can use the **migrate\_verification\_data** utility to migrate design validation objects created prior to Teamcenter 12.4 and Active Workspace 5.0.

### New validation design structure and objects

Prior to Teamcenter 12.4 and Active Workspace 5.0, the design verification feature was based on the following **test package objects**:

- **test**
- **Test Request**
- **Test Event**

Releases with Teamcenter 12.4 with Active Workspace 5.0 and later support the physical verification process with a new objects. The following summarizes the object name changes:

- **Test request** is now called **Test method**.
- **Test event** is now called **Test lab procedure**.

The following graphic shows the data organization. The objects in the green are design validation objects introduced in Active Workspace 5.0. All other colors indicate pre-existing objects.



## Run the migration utility

The `migrate_verification_data` utility migrates the physical verification data objects to a study/run-based data model.

The migration utility does not run as part of the upgrade process. Run the utility after you upgrade Teamcenter and Active Workspace.

```
migrate_verification_data -u=user_id { -p=password | -pf=password-file } -g=group [-h] -logfile=log
file - vritemids=List of comma-separated Item Ids
```

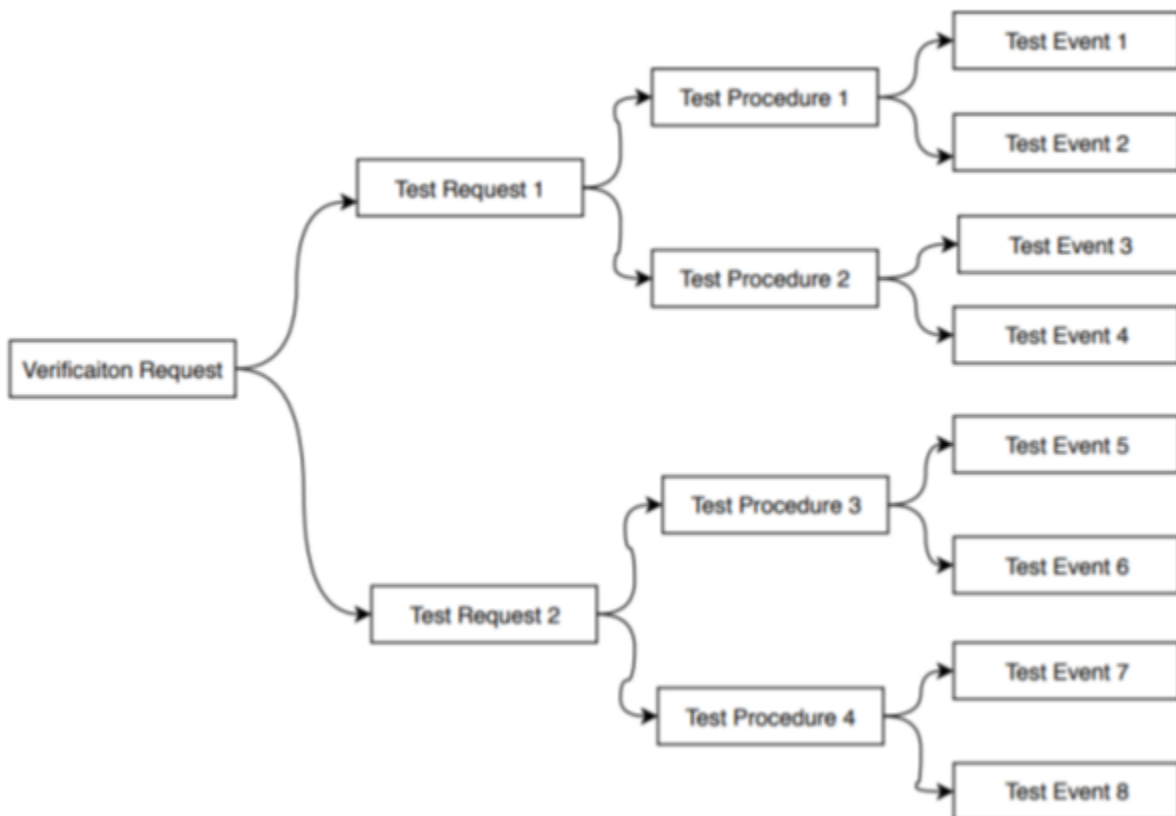
**-u** Indicates the Teamcenter user ID.

- p Indicates the Teamcenter password associated with the `-u=user_id` argument.
- g Indicates the Teamcenter group associated with the `-u=user_id` argument.
- h Displays help for the utility.
- logfile Indicates the file path to the log file.
- vritemids (Optional) Indicates a list of comma-separated test object IDs to migrate. If you do not provide this argument, then all the eligible tests are migrated.

## Data migration example

The following example shows the test data state before and after using the `migrate_verification_data` utility. For each existing combination of **Test Request** and **Test Procedure**, the utility creates a **Test Request**. The utility adds **Test Method** and **Test Procedure** to the new **Test Request**.

The following diagram shows the object structure before migration:



The following table shows the object structure after migration:


Test Request 1    Test Event 1  
 Test Method 1    Test Lab Procedure 1  
 Test Procedure 1    Test Event 2  
                           Test Lab Procedure 2

Test Request 2    Test Event 3  
 Test Method 1    Test Lab Procedure 3  
 Test Procedure 2    Test Event 4  
                           Test Lab Procedure 4

Test Request 3    Test Event 5  
 Test Method 2    Test Lab Procedure 5  
 Test Procedure 3    Test Event 6  
                           Test Lab Procedure 6

Test Request 4    Test Event 7  
 Test Method 2    Test Lab Procedure 7  
 Test Procedure 4    Test Event 8  
                           Test Lab Procedure 8

## Set preferences for test management

Before using test management, you should set the following preferences. For more information about a particular preference, search for it using the **PREFERENCES**  tile on your home page.

### General preferences

#### IAV0SetOccurrenceType

When an interactive duplication is performed during Test EBOM creation, this preference determines if the occurrence type is set on the cloned occurrences. If **true**, the occurrence type is set on the cloned occurrences. If **false**, the occurrence type is not set on the cloned occurrences. When set to **false**, the neutral instruments' tracking is disabled, as it uses the additional occurrence type to follow the workflow. The default value is **false**.

#### IAV0ValidateGRMDelete

This preference manages the validation on deletion of **IAV0VerificationTL** and other GRMs to which the **IAV0CheckGRMDeletePermission** extension is attached. If **true**, validations are enabled. If **false**, validations are disabled. The default value is **false**. For example, the default **IAV0VerificationTLDeletable** condition is attached to **IAV0VerificationTL**, which prevents deletion of the relation between Test Request Revision and Request Execution Info by any user other than **DBA**. This condition is executed only if the **IAV0ValidateGRMDelete** preference is **true**.

### IAV0ValidateGRMCreate

This preference manages the validation on creation of **IAV0VerificationTL** and other GRMs to which the **IAV0CheckGRMCreatePermission** extension is attached. If **true**, validations are enabled. If **false**, validations are disabled. The default value is **false**. For example, the default **IAV0VerificationTLCreatable** condition is attached to **IAV0VerificationTL** which prevents creation of the Test Procedure if the Test Request is not released. This condition is executed only if the **IAV0ValidateGRMCreate** preference is **true**.

### IAV0ValidateVerificationMethod

This preference manages the validation of the **Primary Verification Method** property on the Verification Requirement. If **true**, validations are enabled. If **false**, validations are disabled. The default value is **false**.

If set to **true**, while associating planning artifacts with the Verification Requirement using the **IAV0VerificationTLrelation**, the system verifies the correct object type is associated, depending on the **Primary Verification Method** property set on the Verification Requirement. The **Primary Verification Method** value set on the Verification Requirement must match the value of the **IAV0AllowedVerificationMethod** business object constant of the planning artifact being associated with the Verification Requirement.

### IAV0ValidateSecondaryVerificationMethod

This preference manages the validation of the secondary verification method property on the Verification Requirement. If **true**, validations are enabled. If **false**, validations are disabled. The default value is **false**.

If the value is **True**, while associating planning artifacts with the Verification Requirement using the **IAV0VerificationTL** relation, the system verifies the correct object type is associated, depending on the **Secondary Verification Method** property set on the Verification Requirement. The **Secondary Verification Method** value set on the Verification Requirement must match the value of the **IAV0AllowedVerificationMethod** business object constant of the planning artifact being associated with the Verification Requirement.

If the **IAV0AllowedVerificationMethod** constant is set to **false**, the method verification is completely disabled and the value of the **IAV0ValidateSecondaryVerificationMethod** preference is not considered. If the **IAV0AllowedVerificationMethod** constant is set to **true** and the planning artifact does not match the **Primary Verification Method**, the planning artifact is verified with the values present in the **Secondary Verification Method** property, if the **IAV0ValidateSecondaryVerificationMethod** preference is set to **true**.

### IAV0EnableValidationHardCheck

To ensure readiness before a Test Event is performed, the Request Execution Information workflow verifies the elements that were created during the development of the Test Event. This preference specifies the type of verification performed for request execution information creation. If **true**, the **request execution info** workflow handler checks the content of Test Event Saved Bookmark. If **false**, it bypasses the validations. If **true**, the following are checked:

- The EBOM, Test Method and Test Procedure are released in Test Request Context (not necessarily in the Test Request).

- The presence and release state of the Physical BOM. It must be present in the Test Event Context because it cannot be in the Verification Request.
- The revision rule on the Test EBOM is not **Working**.
- There are no unassigned measurement requirements. All measurement requirements must be related through the Test EBOM to individual Serialized Physical Instruments which have related Calibration Info objects.
- The Test Request is released.

If any of these checks fail, the workflow does not proceed.

### IAV0ReleaseStatusType

Applies when a user releases test requests with a verification management sample workflow such as the Verification Management Test Request Release workflow. It defines all release status values on that are considered when creating request execution information, for example, **TCM Released** and **Approved**. Alternatively it may specify the type of release status to check for on objects such as the Test Method, Test Procedure and Test Request. For example, if the value of this preference is **Approved**, the handler in the Test Request Release workflow checks if the release status of related objects such as Test Procedure and Test Method are set to **Approved**. The default value of this preference is **TCM Released** and **Approved**.

### PLE\_Crt1ResultColour

Enables the user to configure colors displayed in test result elements, such as tables, pie charts, and simple bar charts. You must have administrator rights to update this preference. Values must be entered using the format `<LOV>:<numerical value of the LOV>:<HEX color code>`, for example **Pass:200:#0a9b00**.

#### Note:

Colors for stack bar charts and for vertical bars in the primary work area are updated in the file `\\validationcontractjs\src\_customColorsTheme.scss`. Navigate to the CSS class `aw-verificationManagement-stackedBarColor` to update the colors for stack bar charts. Navigate to the CSS class `aw-verificationManagment-<status>DecoratorColor` to update the colors for vertical bars in the primary work area.

### PLE\_Is\_Create\_VerificationRequest\_With\_Workset

Defines the functionality used to navigate the context of a test. If the value is set to **true**, then the workset functionality is used. If the value is set to **false**, then the saved working context functionality is used.

### PLE\_Publish\_Measurement\_To\_Goal

Defines the behavior to publish measurement from the target to the source parameter. If the value is set to **true**, then the measurement from the target parameter is published to the goal of the source parameter. If the value is set to **false**, then the measurement and goal from the target parameter are published to the corresponding measurement or goal of the source parameter.

### PLE\_Parameter\_Create\_With\_Definition\_UIx

Determines if a parameter definition is required to create a parameter. If set to **true**, then the parameter definition is required; if set to **false**, then the parameter definition is optional.

This preference requires Teamcenter 14.2 or later.

### Overview tab table preferences

**PLE\_Publish\_Measurement\_To\_Goal** Defines the behavior to publish measurement from target to source parameter. If set to **true**, then the measurement from the target parameter is published to the goal of the source parameter. If set to **false**, then the measurement/goal from the target parameter are published to the corresponding measurement/goal of the source parameter.

The **Overview** tab for test management provides a list of tables that users can toggle. You configure the tables available through the following preferences.

### AWC\_Crt0VIdnContract\_DisplayTables

Provides the following features:

- Defines the list of check box to be displayed in **Overview** tab when a test object is selected in **SCOPE** section.
- Allows adding configurable tables:
  1. Add an entry in this preference. For example, we are configuring the table named **One**:

**ConfigTableOne:On:ConfigTableOneColConfig,Crt1ConfigTableOne:Item Revision:'''**

Note:

For subsequent tables, replace the string **One** in each Step with another number (**Two**, **Three**, and so on up to **Nine**) for additional configurable.

2. Define a **ColumnConfig/D**. For example, **ConfigTableOneColConfig**.
3. Define a **ColumnConfigURI** for the ID defined in Step 2. For example, **Crt1ConfigTableOne**.
4. Define a **ColumnConfigURI** for the ID defined in Step 2. For example, **Crt1ConfigTableOne**.
5. Modify the corresponding JSON message on client as **"ConfigTableOne": "section\_name Name**. For example, **"ConfigTableOne": "My Configured Table"**.

The preference supports the following values:

- **realTypeName:defaultStateOfCheckBox**: Sets the default state for a table check box and table display. For example, if you want to display the **Requirements** table by default with the check box selected, then set the preference value to **Requirement:On**.
- **realTypeName\_GroupName:defaultStateOfCheckBox**: Sets the default state for a group table check box and table display. Valid values are **On** or **Off**. For example, if you want to display the table of group **Product And Test EBOMs** by default with the check box selected, then set the preference value to **Part\_ProductAndTestEBOMs:On**

Default preference values are as follows:

- **Requirement:On**
- **Functionality:Off**
- **Fnd0LogicalBlock:Off**
- **Part:Off**
- **Part\_ProductAndTestEBOMs:Off**
- **Att0MeasurableAttribute:On**

#### **AWC\_Crt0Run\_DisplayTables**

Controls the table display for the **Run** object.

#### **AWC\_Crt0Study\_DisplayTables**

Controls the table display for the **Study** object.

#### **AWC\_IAV0TestStudy\_DisplayTables**

Controls the table display for the **Test Request** object.

#### **AWC\_IAV0TestRun\_DisplayTables**

Controls the table display for the **Test Event** object.

### **Rollup preferences**

The following preferences determine how test package results rollup is calculated. View each preference in Active Workspace to view its description. You can search all rollup preferences by the prefix **PLE\_Rollup**.

- **PLE\_Rollup\_Crt0RunRevision**
- **PLE\_Rollup\_Crt0SimStudyRevision**
- **PLE\_Rollup\_Crt0StudyRevision**
- **PLE\_Rollup\_Crt0TestRevision**
- **PLE\_Rollup\_Crt0VerifRequestRevision**
- **PLE\_Rollup\_Crt0VldnContractRevision**

- PLE\_Rollup\_IAV0TestRunRevision
- PLE\_Rollup\_IAV0TestStudyRevision
- PLE\_Rollup\_Result\_Enable

## Updated test objects and workflows

### Install workflows

Install the Design Validation Planning and Reporting **workflows** by executing the **plmxml import** utility from the `%TC_ROOT%\install\iav1\verificationmanagement\data` folder while logging in as a user with Teamcenter administration privileges.

### Obsolete and new workflows

Due to the test object model change, the sample workflows prior to the Active Workspace 5.0 do not support these new test objects. If you are running any of these workflows, then delete the obsolete sample workflows and replace with the following new workflows:

Obsolete workflow	New workflow
VM Test Request Approval Workflow	VM Test Request Release Workflow
VM Test Request Signoff Workflow	
VM Test Procedure Approval Workflow	VM Test Procedure Release Workflow
VM Test Procedure Signoff Workflow	
Not available	VM Test Method Release Workflow

The following list summarizes the new workflows:

#### VM Test Procedure Release Workflow

Releases the object **Test Procedure Revision (VM0TestProcedurRevision)**.

Note:

This workflow replaces **VM Test Procedure Release Workflow**.

#### VM Test Method Release Workflow

Releases the object **Test Method Revision (VM0TestRequestRevision)**.

#### VM Test Request Release Workflow

Releases the object **Test Request Revision (VM0TestStudyRevision)**.

This workflow contains the custom rule handler **VM-check-study-book** that ensures the data is correct before the release of the **Test Request** object.

The handler checks the following items:

- **Test EBOM** and **Product EBOM** release status.
- Unassigned measurement requirements.
- **Test Method** and **Test Procedure** are present in the **Test Request Saved Working Context**.
- **Test Method** and **Test Procedure** release status.
- Revision rule of **Test EBOM**. When releasing **Test Request**, working entries are not allowed in **Test EBOM**.

Note:

If you are working in Active Workspace 5.0, then you must use these workflows instead of **VM Test Request Workflow**.

### Supported existing workflows

These workflows are still supported in Active Workspace 5.0 and later:

- **VM Individual Test Readiness Review Workflow**
- **VM Initiate Verification Activity Workflow**
- **VM Prepare Execution Info Record WF**
- **VM Sample Document Release Workflow**
- **VM Sample Structure Release Workflow**
- **VM Test EBOM Release Workflow**
- **VM Transfer Test EBOM Ownership**
- **VM Verification Assessment Form Release Workflow**

## Add participants to the test workflow

Model-Based Systems Engineering provides a workflow that allows you to release a completed test. This workflow has default roles associated with it that are sufficient to perform the necessary tasks in a typical organization, namely design engineer and test engineer. (In the following example, the test engineer is a CAE analyst.) Depending on the complexity of your organization, you may want to enhance this workflow and the following example shows you how to:

- Introduce a new participant, **CAEAnalyst**.
  - Introduce a new test status, namely, **Submitted-CAE**.
  - Allow the **CAEAnalyst** participant to modify test output data, depending on the test status.
  - Configure the new workflow template for the **CAEAnalyst** participant.
  - Show the **CAEAnalyst** participant in the **Participants** location in Active Workspace.
1. (Optional) Create a custom template project in Business Modeler IDE for the new data. You can also customize the standard test template if appropriate.
  2. Define the new **CAEAnalyst** participant as a subtype of **Participant**, and then define **CAE\_ANALYST** as the **ParticipantHandlerKeyword** on the **Business Object Constants** tab.
  3. On the **Crt0States LOV** tab, add a new **Submitted-CAE** entry to the list of possible test statuses.
  4. Define conditions to allow **CAEAnalyst** to modify the test output data before the test status is **Completed**.
    - a. Define a condition that checks if the current user handling the verification is **CAEAnalyst** or not by including the following parameters:

**Description** Is CAE Analyst

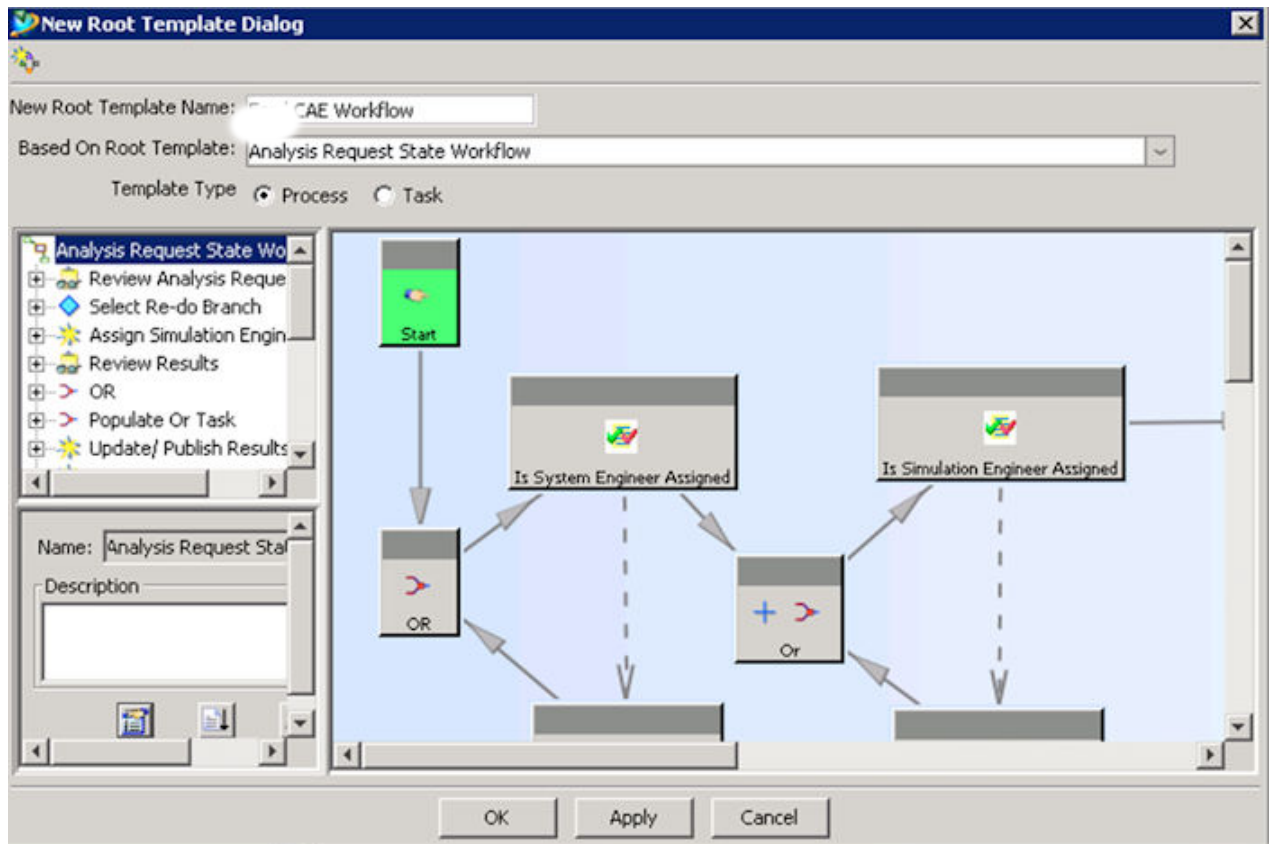
**Signature** FdIsCAEAnalyst ( Crt0VIdnContractRevision o, UserSession u).

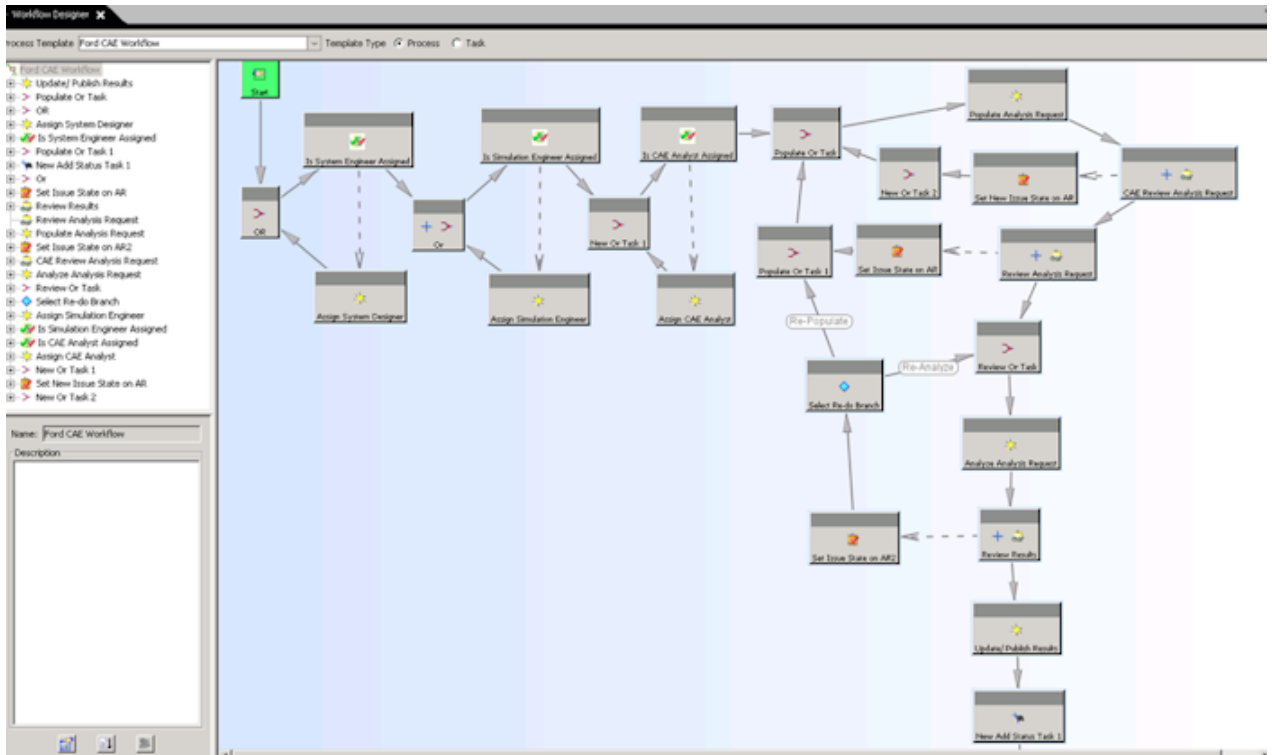
**Expression** o.crt0IsParticipant("CAEAnalyst")

- a. Modify the default **Crt0IsAnalysisRequestOutputModifiable** condition to allow **CAEAnalyst** to change output data until the analysis request is complete by editing the expression as follows:

```
p.crt0State = "" OR ( Condition::Crt0AnalysisRequestRevisionIsModifiable (p, s, u) AND
  Condition::Crt0IsSystemsDesigner( p, u ) OR ( (p.crt0State != "Completed") AND
  (Condition::Crt0IsSSimulationEngineer( p, u) or Condition::IsCAEAnalyst( p, u )
  )
)
```

5. If you want to create a new workflow template, create it from the default test State Workflow template, as follows.
  - a. In Workflow Designer, choose **File**→**New Root Template**.  
  
Workflow Designer displays the **New Root Template** dialog box.
  - b. Select the default template as the basis of your new template as follows.





c. Add the action handlers for CAE Review Analysis Request to **select-signoff-team** as follows.

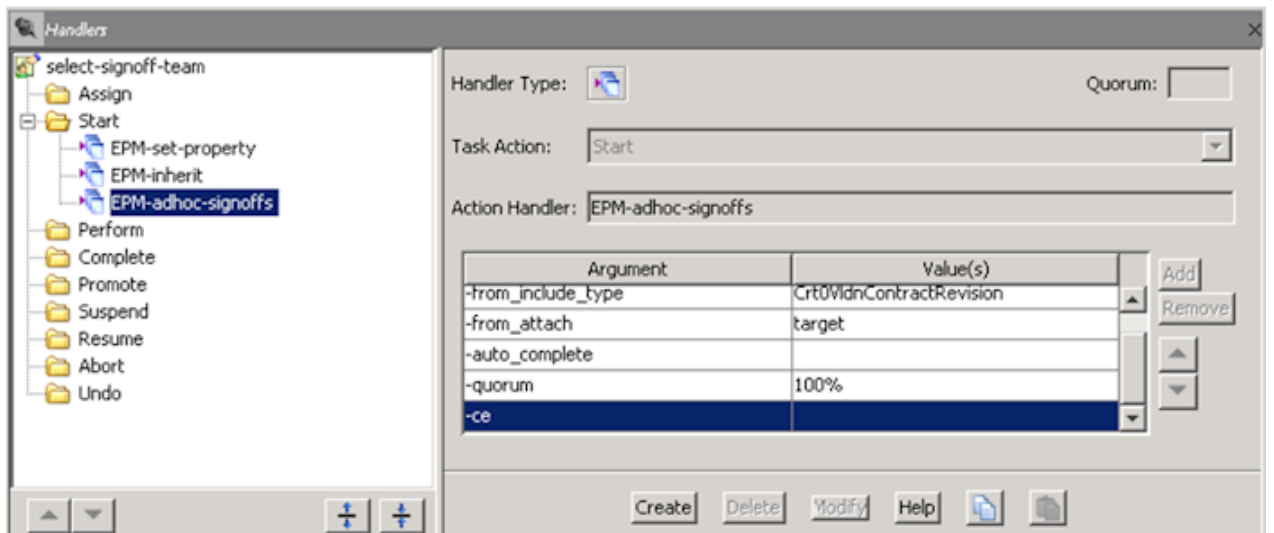
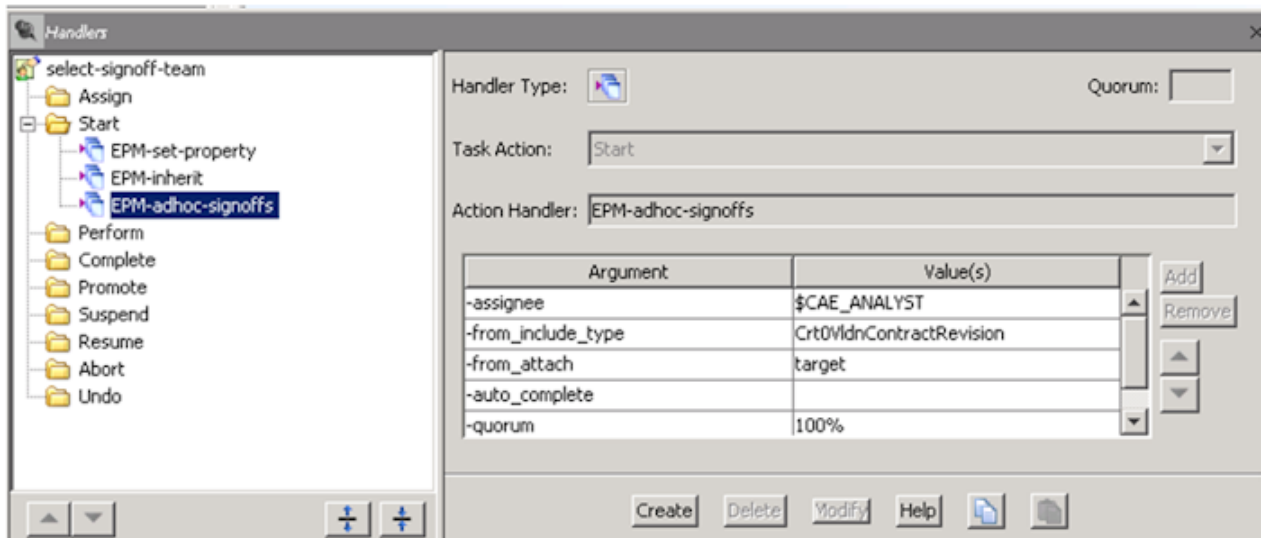
Handler Type: Quorum:

Task Action: Start

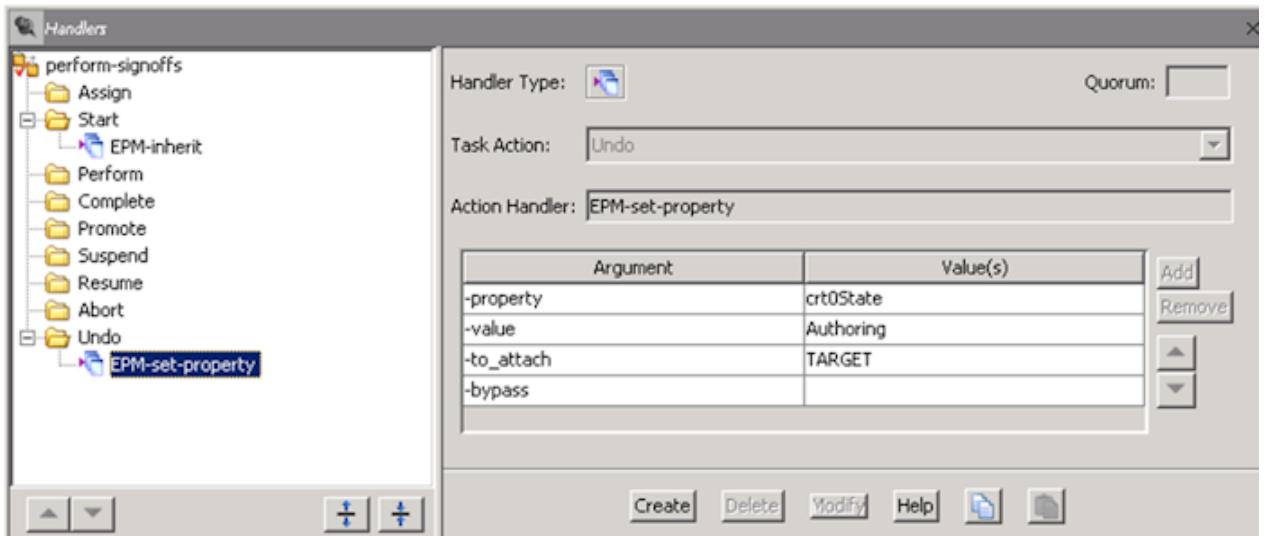
Action Handler: EPM-set-property

Argument	Value(s)
-property	cr0State
-value	Submitted-CAE
-to_attach	TARGET
-bypass	

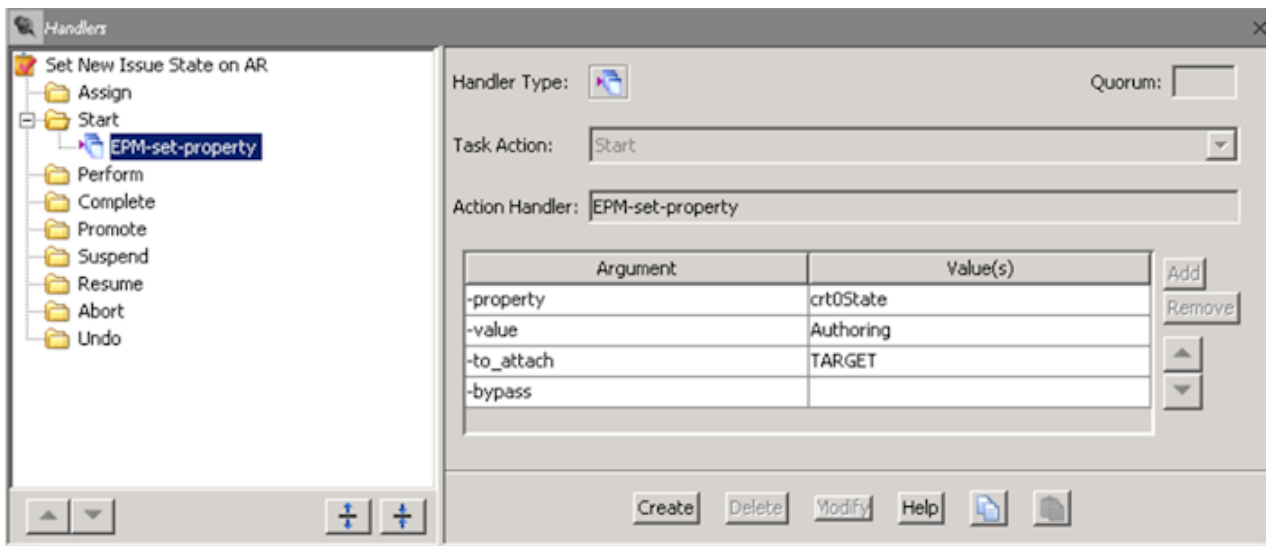
Buttons: Create, Delete, Modify, Help, Add, Remove, Up, Down



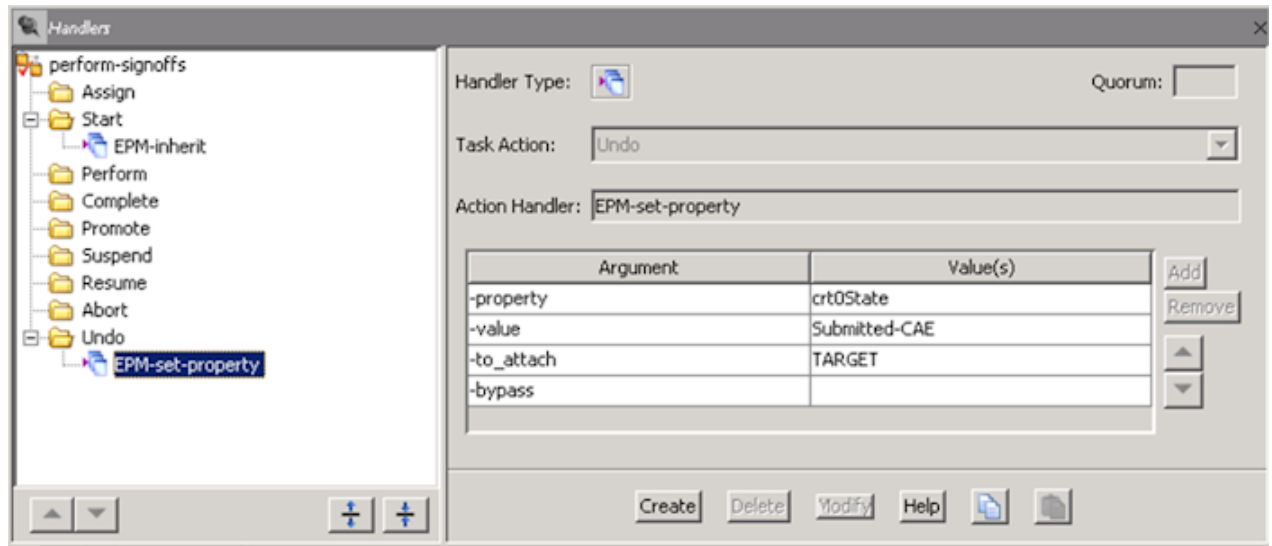
- d. Add the action handlers for CAE Review Analysis Request to **perform-signoffs** as follows.



- e. Add the action handlers for Set New Issue State on the test as follows.



- f. Modify the action handler for Review Analysis Request in **perform-signoffs** to Submitted-CAE if the user performs an undo action, as follows.



6. Add **CAEAnalyst** to the **Participants** tab in Active Workspace.

In the rich client, search for the **Crt0VIdnContractRevSummary** dataset by name, open it in the **Documentation** tab, and then add the following section to the **tc\_xrt\_Participants** page after the existing definitions.

```
<section titleKey="CAEAnalyst">
  <objectSet source="HasParticipant.Fd1CAEAnalyst">
    <command actionKey="newBusinessObjectContextualAction"
      commandId="com.siemens.splm.client.workflow.addParticipant"
      renderingHint="commandbutton" />
    <command actionKey="newBusinessObjectContextualAction"
      commandId="com.siemens.splm.client.workflow.replaceParticipant"
      renderingHint="commandbutton" />
    <tableDisplay>
      <property name="fnd0AssigneeUser" />
      <property name="fnd0AssigneeRole" />
      <property name="fnd0AssigneeGroup" />
      <property name="fnd0AssigneeEmail" />
      <property name="fnd0AssigneePhone" />
      <property name="fnd0AssigneeOrganization" />
    </tableDisplay>
    <listDisplay />
  </objectSet>
</section>
```

7. Open a test in Active Workspace and verify that **CAEAnalyst** appears on the **Participants** tab.

## Make parameter definitions required or optional

You can use the preference **PLE\_Parameter\_Create\_With\_Definition\_Ux** to make parameter definitions required or suppressed when adding parameters to objects. You can also use the preference **PLE\_Parameter\_Create\_With\_Application\_Ux** to allow the selection of **Weight** and **Volume** parameters when creating parameter definitions.

When the selection of a parameter definition is suppressed on the **Add Parameter** panel, the user can still select the unit and the type. The application still creates a definition behind the scenes, based on the selections. The name of this definition is automatically created in the format *<unit\_type>*, for example *km\_double*. The application reuses this definition when the same *<unit\_type>* is used.

Note:

**Weight** is available only when the Weight and Balance feature has been deployed. **Volume** is available when only the Volume feature has been deployed.

### Restrictions and limitations

This feature requires Teamcenter 14.2 or later.

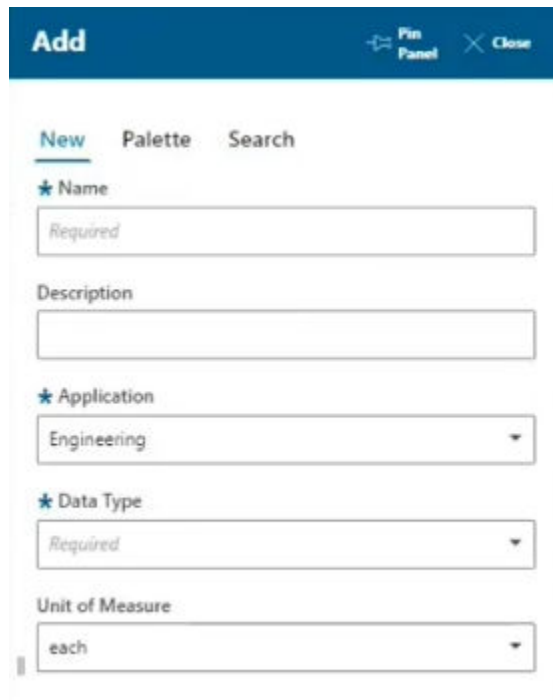
### Procedure

1. Click the **Preferences** tile and then locate the preference **PLE\_Parameter\_Create\_With\_Definition\_Ux**.
2. Do one of the following
  - To make parameter definitions required, set the **Value** to **true**.
  - To suppress parameter definitions, set the **Value** to **false**.
3. (Optional) To allow the selection of weight and balance parameters when creating parameters, locate the preference **PLE\_Parameter\_Create\_With\_Application\_Ux** and set the value to true.

The **Application** field with **Weight** and **Volume** parameters will appear on the **Add Parameter** panel when creating parameter definitions.

### Results

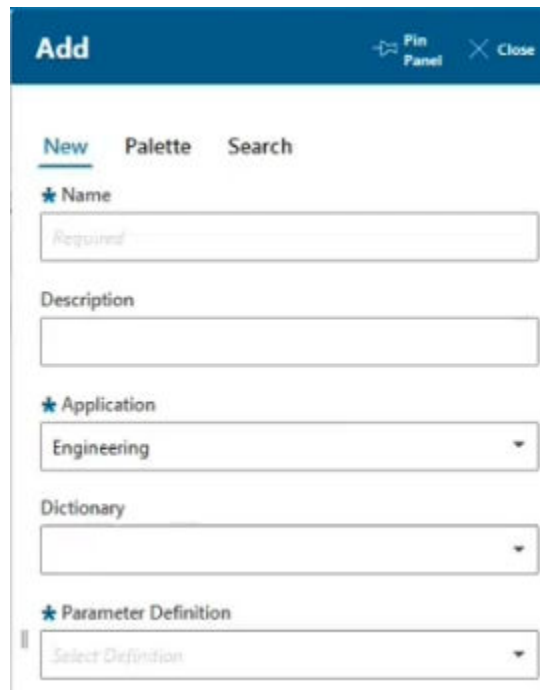
With parameter definitions suppressed, the **Add** panel appears as shown in the following graphic:



The 'Add' panel is shown with a blue header containing 'Add', 'Pin Panel', and 'Close' buttons. Below the header are three tabs: 'New' (selected), 'Palette', and 'Search'. The form contains the following fields:

- Name:** A text input field with the placeholder text 'Required'.
- Description:** A text input field.
- Application:** A dropdown menu with 'Engineering' selected.
- Data Type:** A dropdown menu with 'Required' selected.
- Unit of Measure:** A dropdown menu with 'each' selected.

With parameter definitions required, the **Add** panel appears as shown in the following graphic:



The 'Add' panel is shown with a blue header containing 'Add', 'Pin Panel', and 'Close' buttons. Below the header are three tabs: 'New' (selected), 'Palette', and 'Search'. The form contains the following fields:

- Name:** A text input field with the placeholder text 'Required'.
- Description:** A text input field.
- Application:** A dropdown menu with 'Engineering' selected.
- Dictionary:** A dropdown menu.
- Parameter Definition:** A dropdown menu with 'Select Definition' selected.

## Upgrade from versions of Active Workspace before 4.1

At Active Workspace 4.1, changes to the verification request (analysis request) data model were made to enhance performance. If you are upgrading from an earlier version, you must run the migration utility described below. You must run this utility whether you have a Teamcenter 11 or 12 server platform.

In earlier versions of Active Workspace, when you set the input attributes on an analysis request, the system:

- Copied the attribute from the selected occurrence.
- Associated the copied attribute to the analysis request with the **Att0HasInputRelation** relation.
- Associated the copied attribute to the selected occurrence with the **Att0HasInputOutputRelation** relation.

In later versions of Active Workspace, the measurable attribute is copied to the analysis request only when you edit the measurable attributes in the context of the analysis request. Otherwise, input and output measurable attributes are simply referenced to the main BOM occurrence. Also, the **Att0HasInputRelation**, **Att0HasOutputRelation**, and **Att0HasInputOutputRelation** relations between the analysis request and measurable attributes are replaced by a single **Crt0AttributeElement** data model element.

The measurable attribute is now called parameter.

Note:

As a consequence of these changes, the export through Excel Live of analysis requests is not supported in Active Workspace 4.1 and later versions.

1. To initiate the migration, execute the **migrate\_measurable\_attr.exe** located in the `%TcROOT%/bin` directory. The syntax is as follows:

```
migrate_measurable_attr [-u=user-id {-p=password | -pf=password-file} -g=group]
-logfile=log file
[-h]
```

**-u**

Specifies the user ID.

This is a user with Teamcenter administration privileges.

Note:

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

**-p**

Specifies the password.

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

**-pf**

Specifies the password file.

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

**-g**

Specifies the group associated with the user.

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

**-logfile**

Specifies the path to the log file created by the utility.

**-h**

Displays help for this utility.

2. If you have customized your environment with the analysis request APIs, set the **PLE\_AnalysisRequest\_CopyAttribute** preference to **True**.

If you set this preference to **True**, the system saves measured values back to the analysis request in a copy of the measurable attribute.

If you set it to **False**, the attribute is not copied and instead refers to the in-BOM attribute. That is, the measurement value saved by a direct API writes directly to the BOM.

3. Upgrade Teamcenter Reporting and Analytics. Versions of Reporting and Analytics before 4.1 do not display measurable attributes for the new data model correctly.

## Upgrading from customizations before Active Workspace 5.1

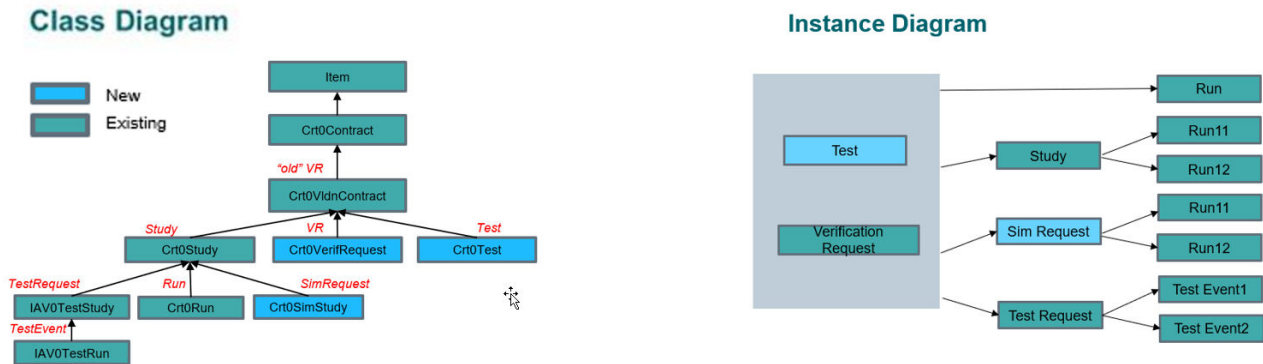
Starting with Active Workspace 5.1, the Verification Request object uses the **Crt1VRContentProxy** run time business object instead of **Awb0Element** in the **Content** table. This change also applies to all other tables that show the content of Verification Request, including the **Test Method** and **Test Procedure** tables. If you have customizations with **Awb0Element**, you should update them to use the new **Crt1VRContentProxy** object type. The source object of **Crt1VRContentProxy** may be a workspace object or an **Awb0Element** that can be accessed using the **crt1SourceObject** property of **Crt1VRContentProxy**.

## Data Auto-migration to Active Workspace 5.2

This release provides the following updates:

- A new type called Test is added. The functionality is currently the same as the Verification Request type.

- Five lower level objects added: Run, Study, Simulation Request, Test Request, and Test Event.
- Auto-migrate users to the new verification request structure. Verification requests from previous releases function the same, but new verification requests are based on the new structure.



## Configuring views

### MBSE views licensing

The MBSE views functionality is licensed, which auto-deploys with the license key **view\_mgmt\_recipe**. The license includes the following features:

- Design Validation Planning and Reporting
- Integration to MATLAB Simulink
- Parameter Management
- Relationship browser
- System Modeling Workbench - Enterprise
- System Modeling Workbench - Personal
- System Modeling Workbench - Professional
- Teamcenter Gateway for EDA
- Teamcenter Gateway for EDA Library
- Teamcenter Gateway for EDA Library for Model Management
- Teamcenter System Modeler





# 3. Managing requirements

## Requirements management with MBSE



The product definition is a phase in an integrated MBSE approach and is the primary creation phase where all the required assets are authored such as requirements, parameters, system architectures, interface diagrams, and domain architectures to 1D/3D models, along with software and test specifications. The collection of these elements is then used to build models that represent system and subsystem physics, and the connection of those systems through interfaces across all relevant development domains. Once available, these reusable digital models can be used in later development stages of simulation and product validation. This results in a fully specified and traceable product aligned to stakeholder requirements.

To manage requirements, an organization may use spreadsheets, linked documents, custom databases, and document-oriented tracing tools. Typical problems in such methods are that requirements are isolated on individual computers with limited access, stored in databases with little resemblance to the product structure, or maintained through complicated user interfaces with significant learning curves. MBSE addresses these problems by bringing requirements into the life cycle and simplifies requirement development and access, and substantially reduces the learning curve. Requirements Management helps you by bringing requirements into the life cycle and simplifies requirement development and access, and substantially reduces the learning curve. Using Active Workspace, system analysts develop requirements with an editor in Active Workspace or Microsoft Office Word or Excel.

The following table shows the high-level roles engaged with requirements management:

 Administrator	MBSE administrators include those who install MBSE and those who configure the specific areas of MBSE such as requirements management or verification management.
 System analyst	Reviews the customer needs, identifies use cases, derives, and identifies system requirements, and performs function behavioral analysis.
 System designer	Creates and manages systems, performs functional allocation, and models system interactions.
 System tester	Reviews the output of the system analyst and system designer to define appropriate test cases for the system and assesses the targets.

## Where do I go from here?

 Administrator	See Configuring MBSE.
 System analyst, System designer, System tester	
Learn more about how the MBSE features work together in Active Workspace.	See What is Model-Based Systems Engineering (MBSE).
Begin authoring or importing requirements.	See Acquiring and authoring requirements.
Set design targets for requirements.	See Establish design targets and how to manage global parameters.
Manage and perform design verifications against requirements.	See Verification management.
Manage requirement and design changes.	See Change management and Workflows and tasks.

## Acquiring and authoring product requirements

### Requirement creation process

You can create requirement specifications, requirements, and paragraphs. Requirements are used to build a design verification plan, which describes the strategy used to verify the requirements against the system model.

Business process	Description	Roles
Create or import requirements	Provides the steps to create the requirement components and add content to the components using supported tools. Tasks include: <ul style="list-style-type: none"> <li>• <b>Manually create the specification.</b></li> <li>• <b>Import a specification.</b></li> </ul>	System analyst, system designer, and system tester
Add requirements content	Provides the steps to author requirement content and manage the requirement specification structure. Tasks include: <ul style="list-style-type: none"> <li>• <b>Create and add requirements and paragraphs directly to the specification.</b></li> <li>• <b>Create sibling or child requirements or paragraphs.</b></li> <li>• <b>Modify the requirements structure.</b></li> <li>• <b>Edit requirement properties.</b></li> </ul>	System analyst and project manager

Business process	Description	Roles
Create requirement baseline and approve requirements	Provides the steps to create baseline requirements for later comparison and the steps to submit requirements to stakeholders for review and approval. Tasks include: <ul style="list-style-type: none"> <li>• <b>Create a requirement baseline.</b></li> <li>• <b>Submit requirements to review workflow.</b></li> </ul>	System analyst, system designer, and system tester
Associate requirements with system models using trace links	Provides the steps to view, create, and manage trace links between requirements and other objects, typically system models. Tasks include: <ul style="list-style-type: none"> <li>• <b>View trace links.</b></li> <li>• Create trace links.</li> <li>• Work with suspect trace link notifications.</li> </ul>	System tester

## Requirement structure

In Active Workspace, you can create a requirement structure.

**Requirement specification** A requirement specification is a top-level container like a Microsoft Word documents that holds the requirements set and paragraphs.

**Requirements** Requirements (along with paragraphs) form the structure of the requirement specification, similar to the headings in Microsoft Word. You can structure requirements in a hierarchy of parent, child, and sibling relationships similar to headings in Microsoft Word such as **Heading 1** and **Heading 2**.

**Paragraphs** You can use paragraphs as simple text blocks within the structure of a requirement specification that may provide supplementary information. As examples, you can add paragraphs to a requirement specification that serve as introductions to the requirements or that provide requirement background information. Paragraphs do not necessarily provide any compliance information.

## Requirements management user interface tabs

You can use the following notable user interface tabs to manage requirements:

- **Architecture:** Provides a graphical diagram of the requirements structure.
- **Documentation:** Provides the editor environment for authoring requirements.
- **Test Coverage:** Provides the test case coverage status and other test information for selected requirements. You can also create a test case and add it directly to a requirement.

- **Overview:** Provides a summary of the requirement specification properties and a preview of the specification.
- **Parameters:** Provides a parameters table for the selected requirements.
- **Test Results:** Provides the test case coverage status and other test information for selected requirements. You can also create a test case and add it directly to a requirement.
- **Where Used:** Provides an overview of contexts, verification/test requests, references, and structures that are associated with the selected requirement.
- **Attachments:** Manages files, documents, and links with the selected requirements.
- **History:** Provides the revision history for the selected requirement.
- **Relations:** Provides the relation diagram for the selected requirement.
- **Collaboration:** Allows team members to post questions and answers for the selected requirement.



## Working in Requirements Manager mode with a requirement-specific user interface

### Enable Requirements Manager workspace mode

The **Requirements Manager** is a role-based user workspace mode that alters your **Home** page and Active Workspace environment specifically for managing requirements.

ID	Revision Name	Revision	Body Cleartext	Trace Link Flag	Is Suspect	Release Status	Type	Last Modified
029186	Freight Vehicle Specification	A					Requirement S...	23-Apr-2024
029192	Body Systems	A	The body systems detail is in this section be...				Paragraph Rev...	23-Apr-2024
029194	Breaking Systems	A	The breaking systems detail is in this section.			Baseline	Paragraph Rev...	22-Apr-2024
REQ-000051	Antilock Brake Systems	A	The system m ust be equipped with a servic...			Baseline	Requirement R...	22-Apr-2024
REQ-000060	Breakaway and Emergency Br...	A	Every motor vehicle, if used to tow a trailer ...			Baseline	Requirement R...	22-Apr-2024
029188	Vehicle Targets	A				Baseline	Paragraph Rev...	22-Apr-2024
REQ-000064	Fuel Economy	A	All light duty vehicles shall comply with loca...			Baseline	Requirement R...	22-Apr-2024
REQ-000050	Fuel Consumption	A	Fuel consumption is a more accurate measu...			Baseline	Requirement R...	22-Apr-2024

- Filters the page information and tiles to requirements-related actions.
- Provides tile-based actions such as create, open, or import requirement specifications.
- Changes the Primary toolbar actions to requirement-specific tasks only, such as import specification, create trace links, and generate a trace link matrix.
- Provides workflow status, test coverage status, and comment status for your selected requirement context.

1. On the Global navigation, click the **Home**  icon.
2. Click your **Profile**  icon.
3. From the **Workspace** drop-down, select **Requirements Manager**.
4. Create, import, or open a specification to open the specification and display the Primary toolbar in the Requirements Manager mode.

You can perform actions in Requirements Manager mode in the following tabs:

- **Documentation**: View and edit the requirement specification and requirements in the editor.
- **Test Coverage**: View the test requests associated with the selected requirements.
- **Overview**: Manage requirement parameters and trace links.
- **Summary Table**: Provides an overview of the requirement structure by paragraph for the specification in a tabular format. Allows editing properties such as trace links, requirement content, and comments.
- **Test Results**: View the test results for the selected requirements.
- **Attachments**: View documents, other files, or URL links associated with the selected requirements.
- **History**: View the change history for the selected requirements.
- **Relations**: View the relationship between the selected requirements and other objects.

## Manage workflow, test coverage, and comment status in the Requirements Manager

You can view the status of requirement workflows, test coverage, and user comments in the **Requirements Manager** mode. The status information helps you determine what action to take on requirements, tests, and user comments.



You can adjust the dividers between panels to minimize or maximize graphs, tables, and other areas.

1. Activate Requirements Manager workspace mode.
2. To filter the status data, locate the **CONTEXT** panel, then perform any of the following:
  - Select a requirement **Type**, such as a requirement specification revision or test specification, or select **All**.

Available requirement types are updated with the preference **REQ\_DashboardSearchTypes**. For help with updating this preference, contact your system administrator.

- Enter a **Filter** string.

Note:

Wildcards are not supported; however, you can enter partial strings.

The requirement results list updates automatically as you revise the filters.

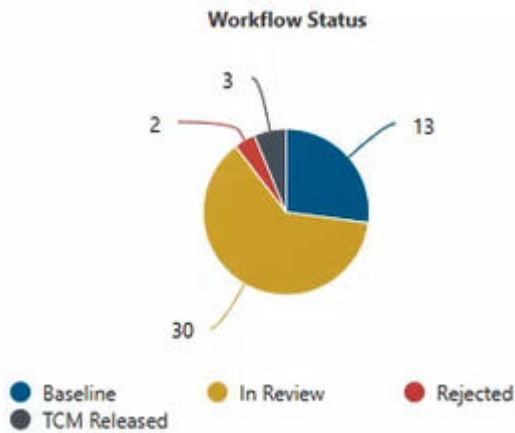
3. Select a requirement from the generated list to display its data.

The pie chart reports include the following:

- **Workflow Status:** Displays a requirement approval count grouped by status.
- **Test Coverage:** Displays a count of requirements with and without associated tests.
- **Comments:** Displays a comment count grouped by review status.

4. Perform any of the following actions:

- Click a pie chart piece to filter charts and tables with that criterion. The query filter appears at the top of the of the pie chart panels.



For example, click the **In Review** pie piece in the **Workflow Status** panel to filter and display only those requirements that are in the review phase.

- Click a pie chart label to toggle its display in the pie chart. For example, click **Baseline** in the previous **Workflow Status** graphic to toggle the baseline data display.

## HTML Templates

### Create an HTML template for requirements

You can create a user template to:

- Define the formatting for HTML-format specifications, requirements, and paragraphs that you view.
- Add standard HTML tags and document properties such as the date the requirement was last edited and who edited the requirement last. For a current list of supported document properties, contact your system administrator.
- Hide any or all three object headers, the object ID, paragraph number, and the object header name or title in the **Documentation** tab or Word document after export. Your administrator can create these templates. The HTML template header can only be edited from rich application client.

Note:

Each object type can be defined in only *one* user template.

For example, you create a template named **Template1** that defines the formatting for the **Requirement** and **Paragraph** object types. If you then attempt to create a second template, **Template2**, that defines the formatting for the **Specification** and **Requirement** object types,

Active Workspace displays an error message. This is because you have already defined the **Requirement** object type formatting in **Template1**. To resolve this error, you must either remove the **Requirement** object type from **Template1** or delete the template entirely.

Your user template overrides any other templates defined by your system administrator.

To create a user HTML template:

1. Navigate to the folder where you want to create the template, and go to **More commands ... > New ✨ > Add**.

The **Add** panel appears.

2. In the **Type** field, find and select **HTML Object Template**.
3. Enter a template name. The **ID** and **Revision** fields are assigned, but you can change them.
4. In the **Associated With** field, select one or more object types, such as **Requirement** or **Paragraph**, to apply the template to.
5. (Optional) Add the template to a project by clicking **Add Project** ⊕, selecting a project from the list, and clicking **Assign**.
6. Click **Add**.

The template appears in the structure pane, and a preview of the template appears in the **Overview** tab.

7. Click the **Documentation** tab, and then click **Edit** ✎.

8. Do one of the following:

- Click **Summary** to edit the template.
- Click **Check Out** to limit the edits to the user performing the checkout.

Note:

You must click **Start Edit** to make edits, and then check in the template for other users to view the edits.

- Click **Briefcase Check Out** to allow edits at another site.

For information on using Briefcase files, see the Active Workspace Fundamentals documentation.


- Click **Delete** to delete the template.

9. Click **Edit**  > **Save Edits**.

## Create a new HTML template for requirement specifications

You can create a user template that already has several details in place, saving time and effort in the process.


To create a user HTML template:

1. Open the **Home** folder, or the folder where you want to create the template.
2. Click **More commands** **...** > **New**  > **Add**.

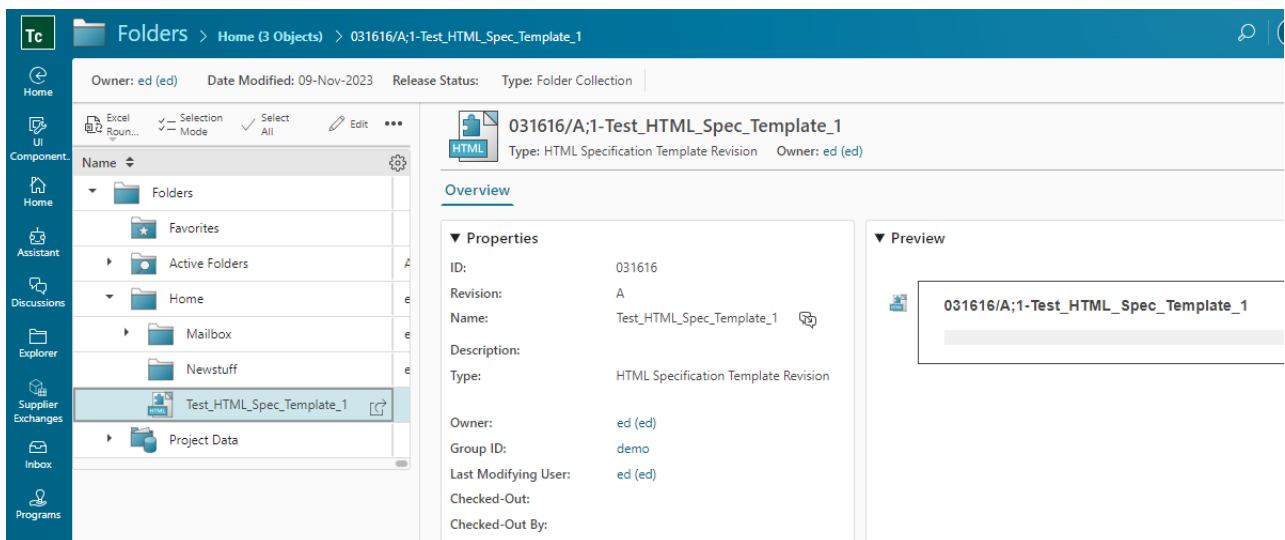
The **Add** panel is displayed.


3. In the **Type** field, find and select **HTML Specification Template**.

The **HTML Specification Template** fields appear.

4. Enter a name and description for the template.
5. (Optional) Add the template to a project by clicking **Add Project** , selecting a project from the list, and clicking **Assign**.
6. Click **Add**.

The template appears in the structure pane, and a preview of the template appears in the Overview tab.



7. (Optional) To edit the properties of the template from the **Overview** tab, click **Edit** , and do one of the following:

- Click **Summary** to edit the template.
- Click **Check Out** to limit edits to the user performing the checkout.

You must click **Summary** to make edits and then check in the template for other users to view the edits.

- Click **Briefcase Check Out** to allow edits at another site.



For information on using Briefcase files, see the Active Workspace Fundamentals documentation.

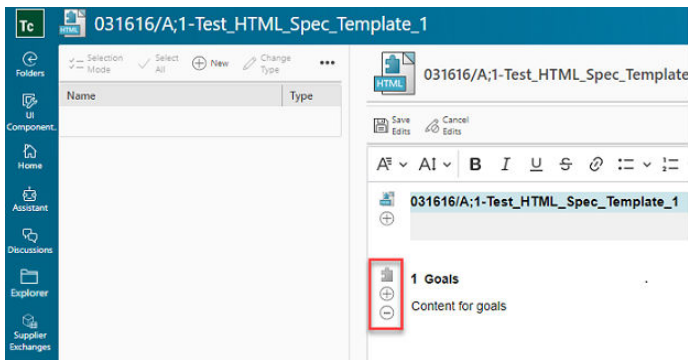
- Click **Delete** to delete the template.

## Editing the HTML template

1. **Locate the template** that you want to edit in the structure pane, and then click **Open** .



The template appears in the work area.

2. Click **Add**  in the template to create objects, or click **Remove**  to delete objects.




The changes are reflected in the tree view on the left side of the screen.

### Note:

Click **Move Up**  or **Move Down**  to move the selected requirement up or down but within the same parent.

You can use promotion and demotion to modify the requirement specification structure by changing the parent-sibling-child relationships. Use **Promote** and **Demote** in conjunction with the **Move up** and **Move down** features to adjust the structure.

3. Make any other edits to the text or objects, and then click **Save Edits**  on the template.

## Create a requirement specification from an HTML template

Requirement specifications are comprised of requirements, paragraphs, and even other specifications. You can create the requirement components, including requirement specifications, requirements, and paragraphs using an HTML template. Consider the following:

- The template supports commands that are used to edit the specification template.
- Header, cover page, and footer with contents and images are supported. You can create these in a specification export template. See [Create a requirement specification export template](#) for information on creating these items.
- You can add images or a logo to the cover page.

The following graphic shows a sample HTML specification template.

**Header**

**{%object\_name}**

**Cover Page**

Teamcenter-ID: {%item\_id}/{%item\_revision\_id}

**SIEMENS**  
*Ingenuity for life*

**{%object\_desc}**

Status: {%release\_statuses}


**Footer**

**{%page\_number} or {%total\_page}**

1. Display the folder to which you want to add the requirement specification, for example, your **Home** folder or other folder designated by your administrator to hold new specifications.
2. Click **More commands ...** > **New** ✨ > **Create Specification**.

The **Create Specification** panel is displayed.

- In the **Create Specification** panel, enter a **Name** and **Description**.
- Select the **HTML Specification Template** from the list. The templates defined for your organization appear.
- Click **Add**.

The specification opens in the **Documentation** tab, where you can enter your content and add objects. You must click **Save Edits**  or use the keyboard shortcut Ctrl + S to save any changes.

## Importing and exporting requirement content

### Working With Microsoft Excel files

#### Create a Microsoft Excel template for import

The file used for importing Microsoft Excel files must follow a specific format. You can create your own file.

The following graphic shows a simple one-sheet Microsoft Excel file that is properly formatted.

	A	B	C	D	E	F
1	<b>Tc_Level</b>	<b>Tc_ObjectType</b>	<b>Title</b>	<b>Information</b>	<b>Component Description</b>	<b>Access Level</b>
2	0	RequirementSpec	Vehicle Requirements	Vehicle Requirement	In principle, a complete vehicle control system involves which controls all aspects of the vehicle.	secret
3	1	Requirement	Complete Vehicle System	<b>REQ-10-Complete Vehicle System</b>		secret
4	2	Requirement	Vehicle System Specification	<b>REQ-11-Vehicle System Specification</b>	defines the set of vehicle signals	secret
5	2	Requirement	Reserved	<b>REQ-12-Engine Compartment Installation</b>	The engine compartment wiring harness W32 connects the governor	secret
6	3	Requirement	Engine Compartment Installation	<b>REQ-121-Noise - Exterior</b>	Dealing with exterior noise is an exceedingly common noise issue.	secret
7	1	Paragraph	Fuel Economy	<b>REQ-20-Body System</b>	The motor system is the part of the central nervous system that is involved with move	secret
8	2	Paragraph	Noise - Exterior	<b>REQ-21-Body Structure Subsystem</b>	This is part of motor system	secret
9	3	Paragraph	Noise - Interior	<b>REQ-211-Body Closure Subsystem</b>	Functional Subsystem etc	super-secret
10	4	Paragraph	Control Subsystem	<b>REQ-2111-Rear Side Doors</b>	A vehicle typically has doors: front doors and rear doors.	super-secret
11	4	Requirement	Control Subsystem	<b>REQ-212-Suspension System</b>	Suspension is the system of tires, tire air, springs, shock absorbers and linkages that connects relative motion between the two.	super-secret
12	2	Paragraph	Type Approval & Def/Class	<b>REQ-30-Front Wheel Alignment Specs</b>		super-secret
13	3	Paragraph	MATLs & Restricted Substances	<b>REQ-31-Front Springs</b>		super-secret
14	2	Requirement	Vehicle System Specification	<b>REQ-40-Vehicle System Specification</b>		super-secret
15	2	Requirement	Reserved	<b>REQ-50-Engine Compartment Installation</b>	The engine compartment wiring harness W32 connects the governor	secret
16	<endtag>					

- Open a blank Microsoft Excel spreadsheet.
- Cell A1** must contain the header **Tc\_Level**, indicating the hierarchy within the specification. 0 is the specification level, level 1 is the requirement level, level 2 is the paragraph level, and so forth. Cell A1 is highlighted in the graphic above.
- Cell B1** must contain the header **Tc\_ObjectType**. **Object Types** include the highest level such as **RequirementsSpecification**, subtype levels such as **Requirement**, **Target** and **Paragraph**. Cell B1 is highlighted in the graphic above.
- Add other columns as necessary. All other fields represent what is imported into Active Workspace.

5. At the end of the file, **<endtag>** is required in column A, as shown in the graphic above.

Note:

- Multiple worksheets can be imported from any Microsoft Excel file to Active Workspace. For more information about importing multiple worksheets, see Import Microsoft Excel with multiple BOM.
- String and numeric properties are the only supported properties for Microsoft Excel import.
- During an export to Microsoft Excel, the columns that have a configured runtime property on the BOMline are exported. However, there is no support for the export of properties on **AwB0Element**; therefore, only those columns that are also on BOMline are considered for export. Any visible column if present on the **Awb0Element** is skipped during the export.

### Create a Microsoft Excel template for export

Active Workspace includes a default Microsoft Excel template that you can use to create a customized template for exporting workspace objects to Microsoft Excel. Siemens Digital Industries Software recommends that you do not make changes to the default template, but rather make a copy and modify the copy for your use.




### Restrictions and limitations

Only a user with the DBA role can create a Microsoft Excel template for exporting workspace objects.

In the Excel file, you must leave one empty column between the properties table on the left and the rules table on the right.

G	H	I	J	K	L
					.
Data Modified	CO		Level	Type	
{%last_mod_date}	{%checked_out}		<rule>		

### Procedure

1. From the **Home** screen, click **Explorer**  icon from the left panel and navigate to the folder where you want to create the template, but do not select any objects within the folder.
2. Click **More commands**  > **New**  > **Add**.

The **Add** panel appears.

3. In the **Type** field, filter and select **Excel Template**.
4. Type a name for the template, and then click **Add**.

The template opens in the **Overview** tab.

5. Open the template from the structure, and then download the Microsoft Excel template file from the **Attachments** tab.
6. To modify the template, open the Microsoft Excel file, and choose any of the following actions:
  - Insert columns in the left-hand table to add properties.  
Refer to the BMIDE for information on property names.
  - Set rules for the object types in the right-hand table.
7. Save the Microsoft Excel file to keep all changes.
8. On the **Attachments** tab, click **Add to** ⊕.

The **Add** panel appears.

9. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**.
10. In the **Type** field, filter and select **MS ExcelX**.
11. Click **Add**.

The modified template file appears in the **Attachments** tab.

Note:

You can delete the original template file from the **Attachments** tab, retaining only the modified copy.

### Create an object in an exported Excel file


After exporting an Excel file using the Excel Round-trip functionality, you can create a new object in the file.

## Procedure

1. After exporting a file such as a specification into Excel, open the spreadsheet, right-click a row, and click **Insert...** to add the new row.
2. Enter the hierarchy level number in the **Tc\_Level** column, where level 0 is the specification level, level 1 is the requirement level, level 2 is the paragraph level, and so forth.

Note:

You must use a sequential number. For example, you cannot put level 4 under a level 2. It will be considered as level 3 under level 2 when you import changes to Active Workspace.

3. Enter the object type to be created in the **TC\_Object\_Type** column, and leave the **TC\_ObjectID** column blank.
4. Save the changes to the Excel file.
5. Return to Active Workspace and click **Excel Round-trip** , and select **Import Changes**.

Tip:

Remove the new object by using the delete feature in Excel.

## Import Microsoft Excel documents into requirements specifications

You can create specifications by importing Microsoft Excel documents into Active Workspace. The import process uses mapping functionality to create a specification structure based on header information in the Excel document.

For more information about supported editing formats, see About editing content in the Documentation tab or Microsoft Word.


## Restrictions and limitations

- Ensure the Excel document is formatted correctly; otherwise, Active Workspace displays an error message and stops the import process. See Create a template file for information.
- Close the Excel document before importing; otherwise, Active Workspace displays an error message and stops the import process.
- Do not select any objects in the folder where you are importing the document. Otherwise, the **Import Structure** option will not be available.

## Prerequisites

You must create a template file using the correct format. See [Create a template file](#) for information on template formatting.

## Procedure

1. [Create a template file](#).
2. From the **Home** screen, click **Explorer**  icon from the left panel, and navigate to the folder where you want to import the requirement specification, but do not select any objects within the folder.
3. Click **More commands ... > Import/Export > Import Structure**.

The **Import Structure** panel displays.

4. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**.


The **Map Properties** fields appear in the **Import Structure** panel.


Note:

It can take several seconds for Active Workspace to verify that the spreadsheet structure is valid.

5. By default, all the required excel headers are mapped to the respective attributes or objects.
6. (Optional) Do one of the following:
  - To use a saved mapping:
    - a. Click the **Saved Mappings** drop-down list and select a mapping.
    - b. Select the properties to match from each drop-down list.
  - To create and save a mapping:
    - a. Click the not mapped excel header drop-down list and select **Add New** to add properties.  
The **Add Properties** panel appears.
    - b. Click the **Subtypes** drop-down list and select a subtype.  
A list of properties associated with the selected subtype appears.

- c. From the list of properties, select the desired properties check boxes and click **Add**.
  - d. Select the added properties in each of the drop-down lists.
  - e. Enter a unique name in the **Saved Mappings** field.
7. (Optional) Select the **Run in Background** check box. If you select this option, you receive a notification when the operation completes.
  8. (Optional) Click **Preview**, to check if all the BOM structures in excel appears correctly in the left pane.
  9. Click **Import Structure**.

A message displays notifying that the excel import is in progress and you will be notified in alerts  when the import process completes.

10. Click the alert  and then select the **Requirement Manager** notification.
11. Click the **Import Specification** notification, and then click the **Related Objects** link to your imported specification.

## Results

Active Workspace displays the requirement specification in the Table view.

### Export and import Excel requirements using round-trip

You can export selected (or all) workspace objects to Excel, and then make changes, including additions, and deletions, to the Excel sheet, and re-import it into Teamcenter.

The Excel Round-trip process is different from the Export to Excel process from Teamcenter. There are features supported in Export to Excel that are not supported in Excel Round-trip. The following table lists a comparison of the supported features.

Feature Supported	In Excel Round-trip	In Export to Excel
Export using packing on template	No	Yes
Export of properties on relation	No	Yes
Export of tracelinks on occurrences	No	Yes
Export the TraceView Data to Excel	No	Yes

See *Active Workspace Fundamentals* for information on the Export to Excel functionality.


## Restrictions and limitations

- Dynamic compound properties (DCP) are not supported and are skipped if they appear in the table during export.
- Adding an existing object as a new object is not supported.
- Excel Round-trip cannot modify the requirement content by design as it exports the BodyClearText which is a read-only property.
- Excel Round-trip does not support required properties during the creation of new object.
- Table attributes are not supported.

## Prerequisites

- The Export to Excel button must be enabled. See [Export to Excel button functionality](#) for more information.
- You must create a template file using the correct format. See [Create a template file](#) for more information.

## Procedure


1. [Locate the requirement component, such as requirement specification, requirement, or paragraph](#), that you want to export, and then click **Open** .

Tip:

Press the **Ctrl** or the **Shift** key to select multiple requirements.

2. Click **Excel Round-trip**  > **Export to Excel** .

Note:

If you open a requirement specification but you want to export only one of its requirements, select the requirement in the content pane and click **Export to Excel** .

The **Export to Excel** panel displays.

3. (Optional) In the **Settings** section, select any of the following check boxes:
  - **Allow Structure Changes in Excel:** Selecting this option allows external users to modify the structure in Excel.

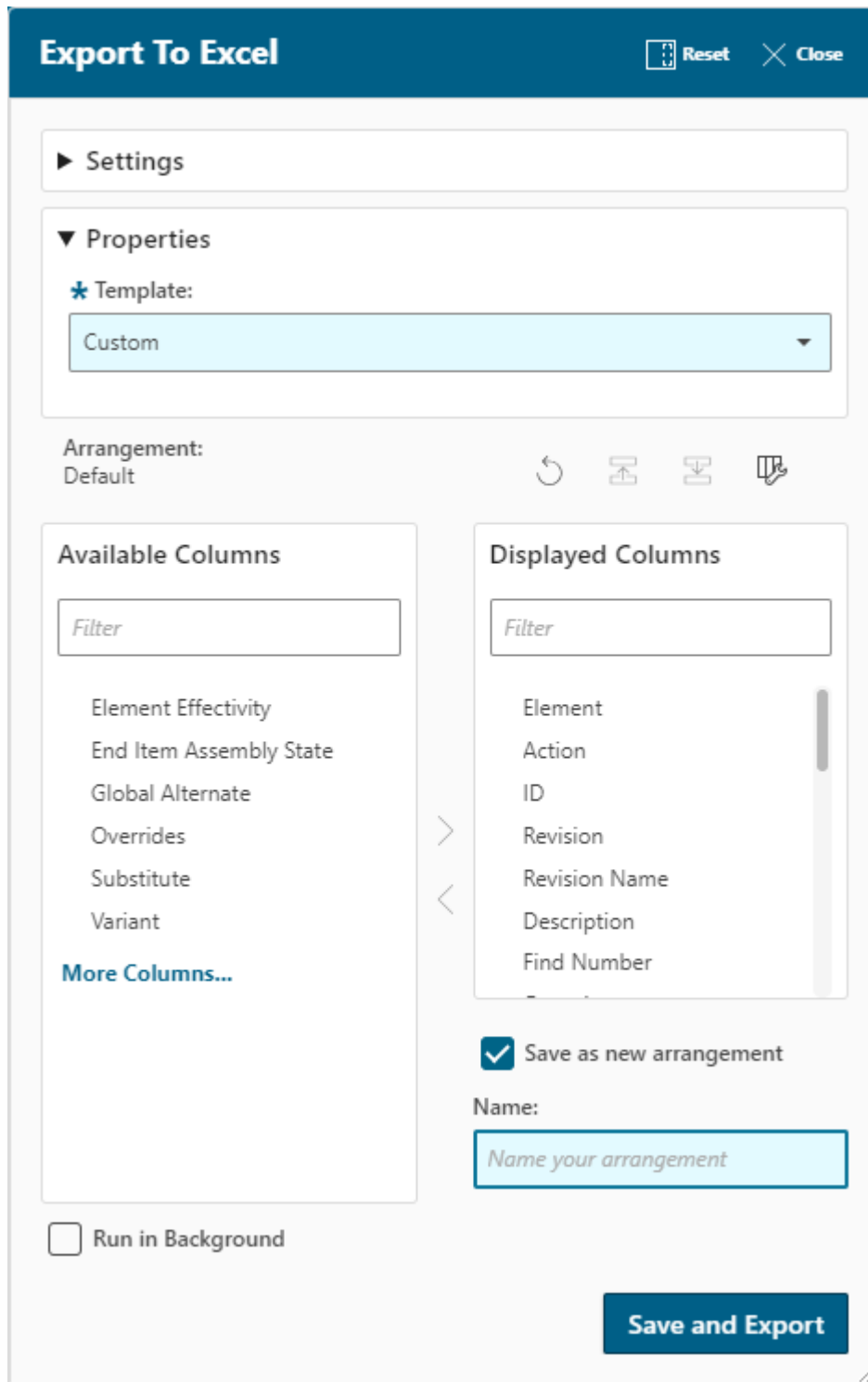
- **Include ID as Hyperlink:** The **Object ID** column, as well as the link, to that object is exported to Excel. You can click the link and open the object in Active Workspace.
  - **Include Outline Numbers:** The paragraph number for the requirement is exported to Excel.
4. In the **Properties** section, select a template from the existing template list, or select **Custom** to create a new template.
- If you select **Template**, you can export the requirement component into a specified template format. You must select a template from the list. The templates available depend on the requirement component that you have selected. See [Create a Microsoft Excel template for export](#) for information on creating a template format.

Note:



To export an interdependent List of Values (LOVs), also known as cascading LOVs, select the template that you have created for exporting interdependent LOVs.

If no template is available, contact your administrator.

- If you select **Custom**, you can select the properties or the columns that are exported to the Excel spreadsheet.





- a. To add or remove column headings, select a name in the appropriate column and click **Add** > or **Remove** < .

- b. (Optional) In **Displayed Columns**, select the property that you want to reorder and click **Move Up**  or **Move Down** .

If the selected column arrangement matches the primary column arrangement and either it contains DCP properties or is modified by the user, it will always be saved as a new arrangement.

- c. Enter a unique name and click **Save and Export**.

You can use a previously created column arrangement by clicking **Column Arrangements**  and selecting a saved arrangement from the list.

To remove a custom column arrangement, highlight it and click **Delete** .

**Note:**

During the export process, only the properties on the Revision and properties on the BOM Line which are mapped on Awb0ConditionalElement for the given property's Awb0BOMToOccurrence Property Constant are exported. There is no support for the export of properties on Awb0Element. Therefore, any visible column present on the Awb0Element is skipped during the export.

5. Click **Export**.

The export process begins, and a scrolling bar at the top of the screen indicates the process is still running.

6. Open the exported spreadsheet and make any edits to the properties listed, and then save the spreadsheet.

**Note:**

- If no objects match the criteria in the Excel template, a blank Excel sheet appears.
- Do not alter the following columns in the Excel sheet: **TC\_ObjectType**, **TC\_ObjectID**, and **ID**. One exception to altering **TC\_ObjectType** is for creating a new object in the structure.
- If LOVs or configured objects or are exported:
  - The exported spreadsheet displays properties as columns and the available list of values as drop-down lists for the columns.
  - Comma-separated values are supported.

- If there are interdependent properties, the values in the drop-down list change depending on the value selected in the driving property column.
- You cannot manually add a value for interdependent properties or cascading LOVs.
- Any updates made in the exported spreadsheet will appear in Active Workspace when the Excel spreadsheet is re-imported.
- When using the round-trip functionality for Excel, the text fields are read-only and are intended for displaying a preview of the requirement or paragraph. Any updates made will not appear when the document is re-imported. You can edit text information only on the **Documentation** tab.

7. (Optional) To create a new object in the structure, do the following:

- a. Open the spreadsheet, right-click a row, and then click **Insert...** to add a new row.
- b. Enter the hierarchy level number in the **Tc\_Level** column, where level 0 is the specification level, level 1 is the requirement level, level 2 is the paragraph level, and so forth.

Note:


You must use a sequential number. For example, you cannot put level 4 under a level 2. It will be considered as level 3 under level 2 when you import changes to Active Workspace.

- c. Enter the object type to be created in the **TC\_Object\_Type** column, and leave the **TC\_ObjectID** column blank.

Note:

When you add a child as an intermediate node under a leaf node in the structure, it appears at the end of the list of pre-existing children of that parent, no matter where it is added in the structure within the Excel spreadsheet.

- d. Save the changes to the Excel file.

8. Click **Excel Round-trip** , and then select **Import Changes**.
9. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**. Select the **Overwrite Conflicts** check box to ensure that your changes are retained in the event of a conflict.
10. Click **Import**.

## Use Excel Round-trip to edit table properties


You can make changes, including addition, and deletion, of table properties in Active Workspace and export them to Excel, and then modify the exported table properties in Excel and re-import it into Active Workspace.

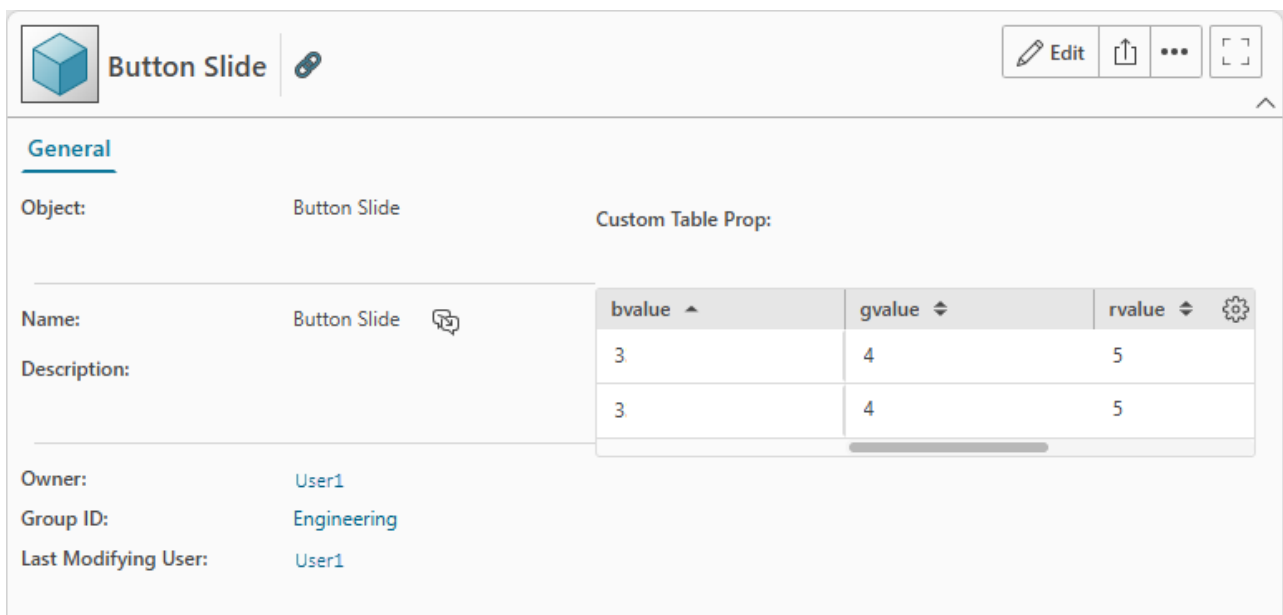
For information about table properties, see *BMIDE for Data Model Design* documentation in the Teamcenter help documentation.






## Prerequisites

You must create a template file using the correct format. See [Create a template file](#) for information on template formatting.

## Procedure

1. From the **Home** screen, click **Explorer**  icon from the left panel, and navigate to the requirement or requirement specification with table properties.



2. (Optional) To add a table property to the existing list of table properties:
  - a. Click **Edit** .
  - b. Click **Add**  and enter the values for the table properties.
  - c. Click **Save** .
3. Click **Excel Round-trip**  > **Export to Excel** .

The **Export to Excel** panel appears.

4. Within **Settings**, ensure all check boxes are clear.
5. Within **Properties**, select the template that you have created to export the table properties.

See [Create a Microsoft Excel template for export](#) for information about creating a template format.


6. Click **Export**.

The export begins, and the progress bar appears.

7. Open the exported spreadsheet, make any edits to the table properties, and then save the spreadsheet.

**Note:**

- In the exported spreadsheet, the primary items are displayed in red, the secondary items related through trace links are displayed in green, and the table properties are listed below the secondary items.
- The table properties can have multiple rows.
- The table properties support LOVs.
- You can not add or delete table properties, these actions will be ignored.

8. Click **Excel Round-trip** , and then select **Import Changes**.
9. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**. Select the **Overwrite Conflicts** check box to ensure that your changes are retained in the event of a conflict.
10. Click **Import**.

#### Import multiple BOMs or structures using Excel

You can import multiple Bills of Material (BOMs) or structures with multiple object types by importing Microsoft Excel documents into Active Workspace. The structures in the Excel can have different object types, such as item, function structure, part, and requirement specification.

#### Restrictions and limitations


- Ensure the Excel document is formatted correctly; otherwise, Active Workspace displays an error message and stops the import process. See [Create a template file](#) for information.

- Close the Excel document before importing; otherwise, Active Workspace displays an error message and stops the import process.
- Do not select any objects in the folder where you are importing the document. Otherwise, the **Import Structure** option will not be available.

## Prerequisites

You must create a template file using the correct format. See [Create a template file](#) for information about template formatting.

## Procedure

1. From **Home**, click **Explorer**  icon, and navigate to the folder where you want to import the multiple BOMs or structures, but do not select any objects within the folder.
2. Click **More commands ... > Import/Export > Import Structure**.

The **Import Structure** panel appears.

3. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**.

The **Map Properties** section appears in the **Import Structure** panel.

**Note:**

It can take several seconds for Active Workspace to verify that the spreadsheet structure is valid.


4. By default, all the required Excel headers are mapped to the respective attributes or objects.
5. (Optional) Do one of the following:
  - To use a saved mapping:
    - a. Click the **Saved Mappings** drop-down list, and select a mapping.
    - b. Select the properties to match from each drop-down list.
  - To create and save a mapping:
    - a. Click the unmapped Excel header drop-down list and select **Add New** to add properties.  
The **Add Properties** panel appears.
    - b. Click the **Subtypes** drop-down list, and select a subtype.  
A list of properties associated with the selected subtype appears.

- c. From the list of properties, select the desired properties check boxes and click **Add**.
  - d. Select the added properties in each of the drop-down lists.
  - e. Enter a unique name in the **Saved Mappings** field.
6. Select the **Run in Background** check box. By selecting this option, you receive a notification when the operation completes.

**Tip:**



It is recommended to select this check box.

7. (Optional) Click **Preview** to check if all the BOM structures in Excel appears correctly in the left pane.
8. Click **Import Structure**.




A message displays notifying that the Excel import is in progress and you will be notified in Alerts  when the import completes.

## Results

Depending on the selection of the **Run in Background** check box:

- If the check box is selected, all the successfully imported BOMs or structures will appear in the alert .
  1. Click **BOM Import Successful**, and a list of all imported structures appears with an **Overview** tab on the right panel.
  2. Select an imported structure and click the **Related Objects** link to view the imported BOM or structure.
  3. Repeat the previous two steps for the rest of the imported structures.
- If the check box is not selected, only the first BOM structure in the first sheet of the Excel will open and the remaining BOM structures will be saved in the destination folder. All the imported BOM structures from the excel do not open simultaneously.
  1. From the destination folder, click an imported structure and click **Open** .

**Tip:**




You can also select an imported structure and click **Open**  > **Open in New Window**  or **Open in New Tab**  from the primary toolbar.

2. Repeat the previous step for the remaining imported structures.


#### Export and import Excel with multiple BOMs using round-trip

You can export multiple types of structures or Bills of Material (BOMs) to Excel and then make changes, including additions and deletions, to the Excel spreadsheet, and re-import it to Active Workspace.

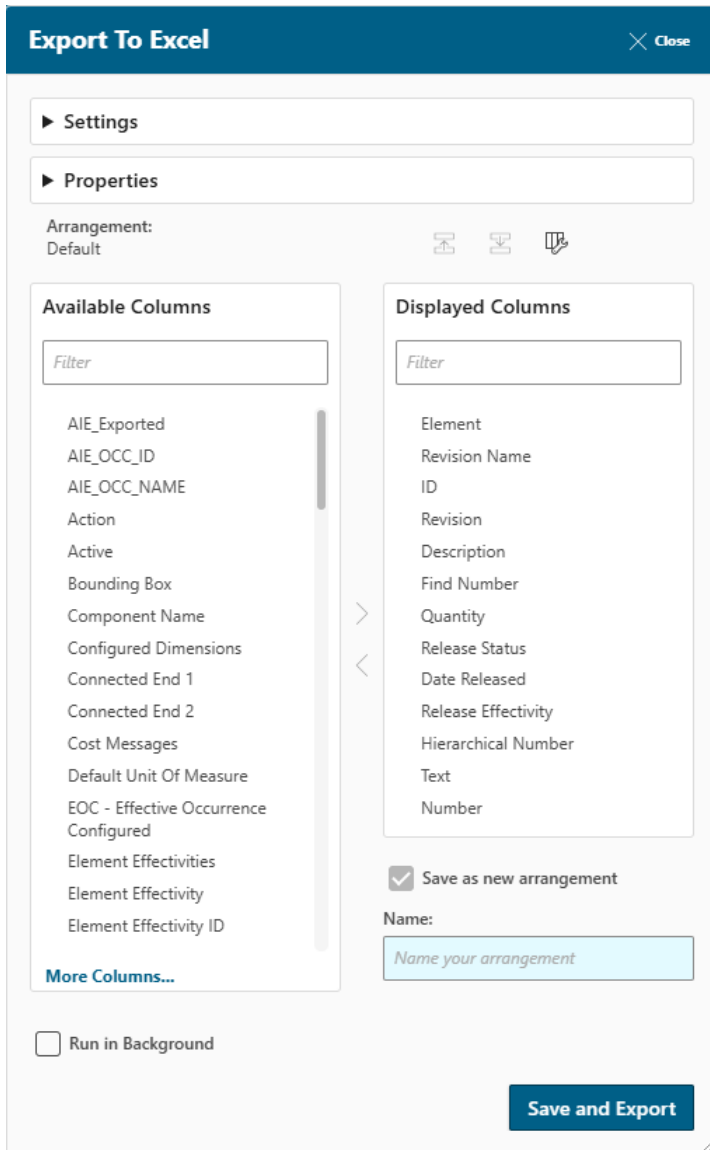
#### Procedure

1. From **Home**, click **Explorer** , and navigate to the folder where the structures or BOMs you want to export are available, and select them.
2. Click **Excel Round-trip**  > **Export to Excel** .

The **Export to Excel** panel appears.

3. In the **Settings** section, select the following check boxes:
  - **Export As Structure:** Selecting this option exports the selected structures or BOMs as structures in Excel.
  - (Optional) **Allow Structure Changes in Excel:** Selecting this option allows external users to modify the structure in Excel.
  - (Optional) **Include ID as Hyperlink:** The **Object ID** column, as well as the link to that object, is exported to Excel. You can click the link and open the object in Active Workspace.
4. In **Properties**, select **Custom** to create a column arrangement that you can use in the future when exporting multiple structures. Then, click **Column Arrangements**  and select a saved arrangement from the list.


The list of default columns that appear in **Displayed Columns** is taken from any one of the selected structures and not from all the selected structures or BOM.



- To add or remove column headings, select a name in the appropriate column and click **Add** > or **Remove** < .
- (Optional) In **Displayed Columns**, select the property that you want to reorder and click **Move Up** ⬆️ or **Move Down** ⬇️.
- Select the **Save as new arrangement** check box and enter a unique name.
- Click **Save and Export**.

The export process begins, and a progress bar appears.

The selected structures are exported to an Excel file, with each spreadsheet representing a structure.

5. Open the exported Excel file, make edits to the properties listed in any of the spreadsheets, and then save the excel file.
6. In Active Workspace, navigate to the folder where the structures that you exported are available, and select one or more structures that you exported.
7. Click **Excel Round-trip** , and then select **Import Changes**.

The **Import Changes** panel appears.

8. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**. Select the **Overwrite Conflicts** check box to ensure that your changes are retained in the event of a conflict.
9. Click **Import**.





A message displays notifying that the Excel import is in progress and that you will be notified in **Alerts** when the import completes.

#### Export and import Excel solution items using round-trip

You can export solution items associated with a change notice (CN) to Excel, and then make changes, including additions and deletions, in the Excel spreadsheet, and re-import it into Active Workspace.

For more information about **Affected Items** and **Solution Items**, see *Change Management on Active Workspace — Usage* in the Teamcenter help documentation.

#### Procedure

1. From **Home**, click the **Changes** tile, or click **Changes**  from the left panel.
2. Search for the change notice you want to work with, select the change notice, and click **Open** .
3. Click **Affected Items** tab.
4. In the **Solution Items** section, select items that you want to export and click **Excel Round-trip**  **Export to Excel** .

The **Export to Excel** panel displays.

5. (Optional) In the **Settings** section, select the **Include ID as Hyperlink** check box to export the **Object ID** column, as well as the link, to that object, to Excel. You can click the link and open the object in Active Workspace.
6. In the **Properties** section, select **Custom** as the template.

Export To Excel
✕ Close

▼ **Settings**

Include ID as Hyperlink

▼ **Properties**

✦ **Template:**

Custom ▼

Arrangement: Default

Filter

↺
⏪
⏩
↻

**Available Columns**

Filter

**More Columns...**

**Displayed Columns**

Filter

- Object
- Lineage
- Type
- Proposed Design Owning Site
- Release Status
- Cancelled
- Date Released
- Owner
- speed
- Mass
- Breaktime
- velocity


Run in Background


Save as new arrangement

Export

- a. To add or remove column headings, select a name in the appropriate column and click **Add** ➤ or **Remove** ➤.
- b. (Optional) In **Displayed Columns**, select the property that you want to reorder and click **Move Up** ⏪ or **Move Down** ⏩.


- c. Enter a unique name and click **Save and Export**.

You can use a previously created column arrangement by clicking **Column Arrangements**  and selecting a saved arrangement from the list.

To remove a custom column arrangement, highlight it and click **Delete** .

7. Click **Export**.

The export process begins, and a progress bar appears.

8. Open the exported spreadsheet and make any edits to the properties listed, and then save the spreadsheet.
9. In the **Solution Items** section, click **Excel Round-trip** , and then select **Import Changes**.
10. Click **Choose File**, navigate to the Microsoft Excel file location, and then click **Open**. Select the **Overwrite Conflicts** check box to ensure that your changes are retained in the event of a conflict.
11. Click **Import**.

## Working With Microsoft Word files

### Preview Microsoft Word content before import

You can preview content from Microsoft Word before importing that content into Active Workspace

When previewing content, you can hover the cursor on a heading to view the Microsoft Word style applied to the heading. This information can be used to define import rules which has a style is specified in the Condition field. You can also add rules before previewing the content. See [Add import rules](#) for information on adding import rules.

You can also reorder the sections of the specification to change the structure and change object types in the structure before the content is imported. See [Manage structure when previewing or importing content](#) for information.

You can also add the imported file as an attachment to the object.

### Restrictions and limitations

- If the requirement specification you are importing skips section levels, the import fills these gaps with empty requirements. An example of this is when your specification has sections 3, 3.11, and 3.12, but does not contain section 3.1, which should appear before 3.11 and 3.12. The import then creates an empty requirement for section 3.1 in the preview.

- Embedded OLE objects such as Microsoft Office files (Word, Excel, and so on) appear as icons in the preview. You can click the icon to open and view these OLE objects, but you cannot edit them. Embedded OLE graphic objects (PNG, JPG, and so on) appear fully rendered in the preview.


## Prerequisites

- You must have the Requirements microservices installed.

See [Configuring Requirements microservices on .NET-based machines](#) for information on configuring Requirements microservices.

- You must also have the REQ\_Microservice\_Installed preference set to true. See [Set preferences for requirements](#) for information on setting preferences.

## Procedure

1. From the **Home** screen, click **Explorer**  from the left panel and navigate to the folder where you want to import the requirement specification, but do not select any objects within the folder.

Tip:


You can also click the **IMPORT SPECIFICATION**  tile on your home page.

2. Click **More commands ...** > **Import/Export**  > **Import Specification**.

The **Import Specification** panel appears.

3. Click **Choose File**, navigate to the Microsoft Word file location, and then click **Open**.

Additional fields appear in the **Import Specification** panel.

4. (Optional) Select the **Add the imported file as an attachment** check box.
5. (Optional) Do any of the following:
  - Select the **Retain Paragraph numbers** check box to keep paragraph numbers.
  - Select the **Create default requirement type items for missing sections** check box to add items for missing sections.
  - Click **Add Rules** , and then complete the fields in the **Add Rules** panel to add parameters such as key words or headers. For more information, see [Add import rules](#).
6. Click **Preview**.

The specification structure appears in the left pane, the preview appears in the middle pane, and the **Import Specification** panel appears in the right pane.

7. (Optional) Reorder sections in the preview using the promote/demote or move up/down actions in the left-hand pane.
8. (Optional) Change object types in the preview pane by clicking the side bar icon.
9. Click **Update Preview** to view your changes prior to importing them.

For more information about importing, see [Import Microsoft Excel requirements specifications](#).


#### Import Microsoft Word requirement specifications

You can import Microsoft Word documents using heading styles or keywords into Active Workspace. During import, the system looks for keyword matches or style matches in paragraphs, and then imports such paragraphs (where matches are found) into separate requirements. If there are no matches found among the given rules, then the paragraph is imported as the default type specified on the import panel.

For more information about supported editing formats, see [About editing content in the Documentation tab or Microsoft Word](#).

##### Note:

- Internal links within a Word document and external URL links are supported.
- BMP and EMF files are not supported when importing a Microsoft Word document.
- OLE objects can be imported as long as the **REQ\_Microservice\_installed** preference is set to **true**. They must be supported by the dataset, and are not set to **Fulltext** in the business object constant **Fnd0DatasetFileExtensionRestrict**. File types restricted by this constant can be viewed in the preview, but will not be imported.
- You can add the imported file as an attachment to the specification object.

1. From the **Home** screen, click **Explorer**  icon from the left panel and navigate to the folder where you want to import the requirement specification, but do not select any objects within the folder.

##### Tip:

You can also click the **IMPORT SPECIFICATION**  tile on your home page and then skip to Step 4.

2. Click **More commands ... > Import/Export > Import Specification**.

The **Import Specification** panel appears.

3. Click **Choose File**, navigate to the Microsoft Word file location, and then click **Open**.

Additional fields appear in the **Import Specification** panel.

**Import Specifi...** Pin Panel Close

To

Home

File: ?

Choose File Test\_... (0.013MB) X

Import Requirement types only

Retain Paragraph numbers

Create default Requirement type items for missing sections

Add the imported file as an attachment

Specification Type:

Requirement Specification

Default Requirement Type:

Requirement

Rules


Run in Background


Preview Import

4. (Optional) Do any of the following:
  - Select the **Import Requirement Types only** check box to import only requirement types.
  - Select the **Retain Paragraph numbers** check box to keep paragraph numbers.

Note:


If you select the **Import Requirement Types only** check box, the **Retain Paragraph numbers** check box is unavailable.

- Select the **Create default requirement type items for missing sections** check box to add items for missing sections.
  - Select the **Add the imported file as an attachment** to add the imported file as an attachment to the specification object.
5. From the **Specification Type** drop-down list, select **Requirement Specification**, and from the **Default Requirement Type** drop-down list, select **Requirement**, which is the default type when there is no match found for the **supplied rules**.
  6. (Optional) To load **saved rules**, click the **Saved Rules** drop-down list, and select an item.
  7. (Optional) To **create rules**, click **Add Rules** ⊕, and then complete the fields in the **Add Rules** panel.
  8. (Optional) Perform any of the following:
    - To add another rule, repeat Step 7.
    - To save the rule, click **Save Rule** , and name the rule. Select **Global Rule** for the rule to apply to all users. Then, click **Save**.
  9. (Optional) Select the **Run in Background** check box. If you select this option, you receive a notification when the operation completes.
  10. (Optional) Click **Preview** to view the requirement as it will appear in Active Workspace. If you add a rule, you can **Update Preview**.
  11. Click **Import**.

The import process begins and a scrolling bar at the top of the screen indicates the object ID and that the process is still running. Active Workspace sends you an alert  when the import process completes.

Note:

If the requirements are already loaded, then there is no way to determine if the import is complete. Manually refresh the page to determine if the import is complete.

12. Click the alert  and then select the **Requirement Manager** notification.

- Click the specification name in the notification.

The specification opens in the **Table** view.



### Import Microsoft Word requirements content not in a specification

You can import Microsoft Word documents that are not part of a specification. You select the import location in either a folder or within your existing specification.


For more information about supported editing formats, see [About editing content in the Documentation tab or Microsoft Word](#).

Note:

- BMP and EMF files are not supported when importing a Microsoft Word document.
- OLE objects can be imported as long as they are supported by the dataset, and are not set to **Fulltext** in the business object constant **Fnd0DatasetFileExtensionRestrict**. File types restricted by this constant can be viewed in the preview, but will not be imported.

- (Optional) [Activate Requirement Manager workspace mode](#).
  - Do one of the following:
    - From the **Home** screen, click **Explorer**  icon from the left panel and navigate to the folder to import the requirements.
    - [Open the requirement specification](#) that you want to import into. Select the **Table**, **Table with Summary**, **Tree**, or **Tree with Summary** view, and then select a requirement.
  - Click **More commands ... > Import/Export > Import Specification**.
- The **Import Specification** panel opens.
- Click **Choose File**, navigate to the Microsoft Word document location, and then click **Open**.
  - Select to import the specification **As Child Requirement**, **As Child Specification**, or **Changes to Specification**.
  - (Optional) Select the **Create default requirement type items for missing sections** check box.
  - Select the **Add the imported file as an attachment** check box.
  - Select the **Default Requirement Type**.
  - (Optional) To [create rules](#) click **Add Rules**  and then complete the fields in the **Add Rules** panel.

#### 10. Click **Import**.

The import process begins and a scrolling bar at the top of the screen indicates the object ID and that the process is still running. Active Workspace sends you an alert  when the import process completes.

#### Note:

If the requirements are already loaded, manually refresh the page to determine if the import is complete.

### Export and import Word requirements using round-trip

You can export requirements content to Microsoft Word in read-only or edit mode (without comments), make changes to the document, and then re-import the content to Active Workspace.

When performing the export process, you can:

- Select a specification export template that applies formatting to the entire exported file. This formatting may include text color, background color, and other text formatting options. See [Create a requirement specification export template](#) for information on creating a specification export template.
- Select an HTML object template that will override the formatting provided by the specification export template. In this instance, the formatting specified in the HTML object template is applied to the exported file, rather than the formatting specified in the specification export template.
- Select a variant rule that filters the requirement specifications or requirement objects in Active Workspace. In this instance, only the requirements and parameters that meet the applied variant


rule are exported to Word document. See [Configuring parameters with variants](#) for information about creating and applying parameter variant values.

- Choose to include any documents linked to the requirements content. These documents may include related test cases, test results, and test coverage.

Note:

Released objects cannot be edited in Word.

## Export the requirements from Active Workspace to Microsoft Word

1. **Locate the requirement component (requirement specifications, requirements, or paragraphs)** that you want to export, and then click **Open** .

Tip:

Press the **Ctrl** or the **Shift** key to select multiple requirements.

2. (Optional) In the **Variant** field, select a variant option from the existing list of variants applied to the requirement. The default value is **No Variant Rule**.

Active Workspace filters the requirement specifications or requirement objects according to the selected variant.

3. Click **Documentation > Word Round-Trip**  **> Export to Word**.

The **Export to Word** panel appears.

4. Select **Read-only** or **Editable** on the **Export to Word** panel.
  - If you select **Read-only**, the document is locked for editing.
  - If you select **Editable**, you can make changes to the requirement content.
5. (Optional) Choose any of the following actions:
  - Select **Include comments** to include any comments made in the document.

Note:

Comments can be created on requirements body text only.

- Select **Include paragraph numbering** to include the original paragraph numbering.
- Select **Include Table of Contents** to include a table of contents.

- Select **Include Parameters** to include the associated parameters.
- Select **Include Variant** to include the applied configuration rule or variant rule information.

Note:

The **Include Variant** option appears only if a variant rule is selected.

- Select **Include Table of Figures** to add a table of figures in the Table of Contents section of the exported file.
- Select **Include List of Tables** to add a list of tables in the Table of Contents section of the exported file.
- Select **Add Watermark** to add a watermark to the exported file. You can define the label text and specify whether the watermark appears in a horizontal or diagonal format.
- Select **Lock Requirement Content** to lock the content for editing; however, the requirement properties can be edited.

Note:

This option appears only if the **Editable** button is selected.

- Select **Edit Structure** to add, delete, or edit objects in the exported file and then reimport those changes.

Note:

This option appears only if the **Editable** button is selected.

6. Select a **Specification Export Template** to include a header or footer or apply text formatting to the exported file.
7. In the Properties section, do one of the following:
  - Select **As Shown** to export default properties.
  - Select **Selected Properties** to choose which object type and properties to export.
  - Select **Override HTML Object Template** to apply a selected HTML object template to a specific object type within the document.
8. Click **Export**.

The export process begins, and a progress bar is displayed. The requirement content is exported to a Word document.

If there are any parameters and variants associated with or applied to the requirement specifications or requirement objects:

- A detailed parameter properties table appears below its respective requirement.
- The variant rule information appears in tabular form at the end of the exported Word document.

**Caution:**

- When a specification is exported, the Track Changes feature is on by default. If it is turned off, any changes made to the file are not imported into Active Workspace. Reimport only updates the changes that are tracked by Word.
- Changed graphics are not tracked by Word, and not imported as part of the re-import. Alternately, you can delete the graphic and replace it in the new version of the document.

## Edit the requirements in Microsoft Word

1. Open the exported document.

The Word document appears with dropdown lists in the object types column.

2. Do any of the following:

- To add a new item, copy and paste an item in the appropriate location. You can then edit the text or item type as necessary.
- To delete an item, highlight the item and delete it.

**Note:**

You cannot modify parameter properties and variant rules in the exported word document.

3. Save the document.

## Import the requirements back to Active Workspace

1. Click **Documentation > Word Round-Trip**  **> Import Changes**.

The **Import Changes** panel appears.

2. Click **Choose File**, navigate to the Microsoft Word file location, and then click **Open**.

The **Options** fields appear.

3. Click **Overwrite Conflicts** to ensure that your changes are retained in the event of a conflict, or click **Import Comments** to import any review comments.
4. Click **Import**.

Note:

After exporting and reimporting an edited Word document:

- The edited content (either modified or newly added items) will adhere to the HTML object template associated with the edited content object type.
- The **History** tab may record more changes than were actually made. This is because the conversion utility converts Word to HTML and also HTML back to Word. For any of these conversions, the conversion utility updates the formatting information and these updates are recorded in the **History** tab, in addition to the changes made in the Word document itself.

### Import a specification from within another requirement structure

You can import specifications into Active Workspace and also import specifications from within the requirements structure, including as child requirements, changes to a specification, or as a child specification.

1. (Optional) Activate Requirement Manager workspace mode.
2. **Import the requirement** you want to work with, if it is not open.

You can also click the **IMPORT SPECIFICATION**  tile on your home page.

3. Navigate to the folder containing the requirement you want to import into, and select **Tree with Summary** from the left pane.
4. Click **More commands ... > Import/Export > Import Specification**.

The **Import Specification** panel appears.

5. Click **Choose File**, navigate to the location of the file to import, and click **Open**.

Additional fields appear on the **Import Specification** panel.

6. From the pane on the left, select the location in the open specification for the new specification.
7. Select the type of import:

- **As Child Requirement:** Imports the selected specification as a child of the requirement at the level you selected.
  - **Changes to Specification:** If you choose **Compare**, the differences between the proposed and current requirement are displayed, and you can choose to **Revise** the original specification based on these differences.
  - **As Child Specification:** Imports the selected specification as a child specification at the level you selected.
8. (Optional) To create rules, click **Add Rules** ⊕ and then complete the fields in the **Add Rules** panel.
- **Import As:** Select **Requirement**, **Paragraph**, **Target**, or **Verification Requirement** from the list.
  - **Condition:** Select this option to search the Word document for matching partial or exact strings, or a designated style.

Complete the following fields:

- a. Select **Words Contain Exact Match** to search the Microsoft Word document for words exactly as you enter them or select **Words Contain Partial Match** to search for the entered words as substrings. For example if you enter **light**, and **electric**, Active Workspace returns only those documents that contain those words exactly as entered. However if you select **Partial Match**, Active Workspace returns documents that contain words such as brakelight, lightning, electrical, and electrics. In both cases, the object is imported as the type selected in the **Import As** list.

Note:

Consider the following when importing content with rules:

- Search words are case-insensitive.
- Paragraphs do not split during import, even if there are multiple keywords matching within the paragraph. The paragraph acts as one object.
- If your content contains a paragraph that follows with an image or an OLE object, then the paragraph and the image or OLE object import as one object.
- Each table is considered a single paragraph.
- Rules run in the order that you enter them.

- b. Enter the words to match, separated each word by a comma (,). The comma acts as an OR operator.
- c. Click **Add**.
- **Has Style:** Select this option to search the Word document for text that matches a Microsoft Word heading (**Heading 1 to Heading 9**):

- a. Select a heading name from the drop-down.
  - b. Select the import format to apply: **Paragraph**, **Requirement**, or **Target**.
  - c. Click **Add**.
- **Property Rules:** Check this box to include property rules as part of the import process.

Note:

**Property Rules** are available when Requirements Management microservices are installed.

- a. **Set Property:** Choose the property from the list.
  - b. Select the value for the property.
  - c. Choose the parameters to follow regarding the conditions: **Always**, **If any of the following conditions are met**, or **if all of the following conditions are met**. If you select any/all conditions, you are prompted to enter those conditions.
9. Click **Import**.

#### Manage structure when previewing or importing content

You can reorder the sections of the specification to change the structure and change object types in the structure before the content is imported.

#### Restrictions and limitations

- When changing object types, you cannot select multiple objects in the preview pane.


#### Prerequisites

- You must have the Requirements microservices installed.

See [Configuring Requirements microservices on .NET-based machines](#) for information on configuring Requirements microservices.

- You must also have the REQ\_Microservice\_Installed preference set to true. See [Set preferences for requirements](#) for information on setting preferences.

#### Procedure

1. From the **Home** screen, click **Explorer**  icon from the left panel and navigate to the folder where you want to import the requirement specification, but do not select any objects within the folder.

**Tip:**

You can also click the **IMPORT SPECIFICATION**  tile on your home page.


2. Click **More commands ... > Import/Export > Import Specification**.

The **Import Specification** panel appears.

3. Click **Choose File**, navigate to the Microsoft Word file location, and then click **Open**.

Additional fields appear in the **Import Specification** panel.

4. (Optional) Do any of the following:


- Select the **Retain Paragraph numbers** check box to keep paragraph numbers.
- Select the **Create default requirement type items for missing sections** check box to add items for missing sections.
- Click **Add Rules** , and then complete the fields in the **Add Rules** panel to add parameters such as key words or headers. For more information, see [Add import rules](#).

5. Click **Preview** or **Import**.

The specification structure appears in the left pane and the preview appears in the main work area. The **Import Specification** panel appears in the right pane only if you clicked **Preview**.

6. (Optional) Reorder sections in the structure using the **Promote/Demote** or **Move Up/Move Down** actions.

7. (Optional) Do one of the following to change object types:


- Select one or more objects in the structure. Click **Change Type**  and then select a type.
- Select an object in the preview pane. Click the side bar icon and select an icon.

8. Click **Update Preview** to view your changes prior to importing them.

For more information about importing, see [Import Microsoft Word requirement specifications](#).

## Add import rules

You can add a variety of rules when you import a Microsoft Word file.

1. (Optional) To create rules, click **Add Rules**  and then complete the fields in the **Add Rules** panel.

**Import Specificati...** Pin Panel Close

← Add Rules

\* Import As:  
Requirement

\* Criteria:  
if any of the following conditions are met

Condition: -

\* When:  
Words Contain Exact Match

Required

+ Property Rules

+ Add Proper...

- **Import As:** Select **Requirement**, **Paragraph**, **Target**, **Verification Requirement**, or **Test Case** from the list.
- **Criteria:** Select if any or all conditions must be met.
- **Condition:** Select this option to search the Word document for matching partial or exact strings, or a designated style.

Complete the following fields:

- **Words Contain Exact Match** to search the Microsoft Word document for words exactly as you enter them.
- **Words Contain Partial Match** to search for the entered words as substrings. For example if you enter **light**, and **electric**, Active Workspace returns only those documents

that contain those words exactly as entered. However if you select **Partial Match**, Active Workspace returns documents that contain words such as brakelight, lightning, electrical, and electric. In both cases, the object is imported as the type selected in the **Import As** list.

- **Does Not Contain Word** to search that the text does not contain the entered word.
- **Has Style** to search for a particular Microsoft Word style.
- **Sentence Begin on New Line** to search for a carriage return/new paragraph.

Note:

Consider the following when importing content with rules:

- Search words are case-insensitive.
- Paragraphs do not split during import, even if there are multiple keywords matching within the paragraph. The paragraph acts as one object.
- If your content contains a paragraph that follows with an image or an OLE object, then the paragraph and the image or OLE object import as one object.
- Each table is considered a single paragraph.
- Rules run in the order that you enter them.

- b. **Property Rules:** Check this box to include property rules as part of the import process.

Note:

**Property Rules** are available when Requirements Management microservices are installed.

- A. **Set Property:** Choose the property from the list.
- B. Select the value for the property.
- C. Choose the parameters to follow regarding the conditions: **Always, If any of the following conditions are met, or if all of the following conditions are met.** If you select any/all conditions, you are prompted to enter those conditions.
  - c. Enter the words to match, separated each word by a comma (,). The comma acts as an OR operator.
  - d. Click **Add**.
- **Has Style:** Select this option to search the Word document for text that matches a Microsoft Word heading (**Heading 1 to Heading 9**):
  - a. Select a heading name from the drop-down.
  - b. Select the import format to apply: **Paragraph, Requirement, or Target**.
  - c. Click **Add**.


2. Click **Add**.

### Word Export Configuration Options

Several options are available for exporting requirements to Microsoft Word. You can include headers and footers, export properties to Word for a round-trip import, and export the requirement as it is seen in the **Documentation** tab.

Note:

- Including headers and footers requires the export microservice.
- Exporting properties to Word for a round-trip import and exporting the view as seen in the **Documentation** tab require both the export and import microservices.

1. **Locate the requirement content** that you want to export, and then click **Open** .
2. Click **Word Round-trip > Export to Word** .
3. Choose **Read-only** or **Editable** for the exported file.
4. Under **Options**, you can choose to include comments, paragraph numbers, a table of contents and a specification cover template.

**Export To Word** ✕ Close

Read-only  
 Editable

▼ **Options**

Include comments  
 Include paragraph numbering  
 Include Table of Contents  
 Include Parameters  
 Include Variants  
 Include Table of Figures  
 Include List of Tables  
 Add Watermark

Specification Export Template:

**Properties:**

As Shown  
 Selected Properties  
 Override HTML Object Template

**Export**

- (Optional) If you want to include a cover page, select a **Specification Export Template**. The specification export template defines the header, footer, and cover page. When a template is revised multiple times, only the latest revision of the template is displayed along with other associated templates in the drop-down list.

6. (Optional) If you choose **Selected Properties**, choose the properties you want to export.
7. Click **Export**.

### Create a requirement specification export template

You can store standard items like a preface, legal text, or table of contents in a template for the Word import/export function, reducing the need for post-processing to the exported Word document.

1. In the **HOME**  folder, click **More commands ...** > **New**  > **Add**.

The **Add** panel appears.

2. In the **Type** field, filter and select **Specification Export Template**.

The **Specification Export Template** fields appear.

3. Enter a name and description for the template. You can change the **ID** or **Revision** if desired.
4. (Optional) Click **Is Global** to apply this template globally.
5. Click **Add**.

The template appears in the **Overview** tab, showing the **Properties** and **Preview**.

6. (Optional) To edit the properties, do one of the following:
  - Click **Summary** to edit the template.
  - Click **Check Out** to limit edits to the user performing the checkout.

You must click **Start Edit** to make edits and then check in the template for other users to view the edits.

- Click **Briefcase Check Out** to allow edits at another site.

For information on using Briefcase files, see the Active Workspace Fundamentals documentation.

- Click **Delete** to delete the template.
7. (Optional) To edit the template, click **Documentation**, and then click in the **Header**, **Cover Page**, or **Footer** area.

You can use the toolbar to make changes to the font, add links, tables, lists, images, or equations.

8. Click **Overview** to return to the overview and preview, and save the template when prompted.

### Edit a requirements specification export template

You can make edits to a specification export template using the **Documentation** tab.

1. From the **Home** folder, open the specification export template.

The template appears in the **Overview** tab.

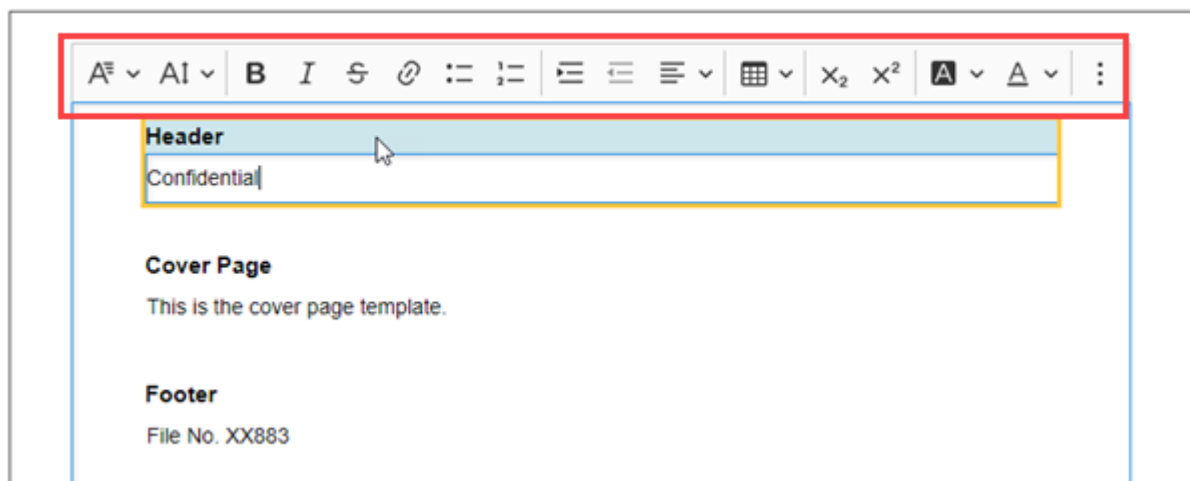
2. Click **Documentation**.

The specification export template appears.

3. Click in the specification export template.

The template opens for editing.

4. Use the toolbar to make changes to the font, add links, tables, lists, images, or equations.



5. Click **Overview** to return to the overview and preview, and save the template when prompted.


### Working With PDF files

#### Import a PDF

You can import PDF documents using keywords into Active Workspace. During import, the system looks for keyword matches or style matches in paragraphs and imports such paragraphs (where matches are found) into separate requirements. If there are no matches found among the given rules, then the paragraph is imported as the default type specified on the import panel.

**Note:**

- You must create bookmarks during the Word export PDF creation process. Manually-created bookmarks within the PDF files are not recognized during import.
- If any page in the PDF has text in a mix of single and multi-column formats, the text may not appear in the correct sequence.
- Header and footer text is skipped in the output.
- Equations do not appear in the correct format.

1. From the **Home** screen, click **Explorer**  from the left panel and navigate to the folder where you want to import the requirement specification, but do not select any objects within the folder.

**Tip:**

You can also click the **IMPORT SPECIFICATION**  tile on your home page.

2. Click **More commands ... > Import/Export > Import Specification**.


The **Import Specification** panel opens.

3. Click **Choose File**, navigate to the PDF file location, and click **Open**.

Additional fields appear on the **Import Specification** panel.

4. (Optional) Select any of the following check boxes:

- **Import Requirement Types only** to select a subtype to import from the drop-down list.
- **Retain Paragraph numbers** to keep paragraph numbers. This option does not appear when the **Import Requirement Types only** check box is selected.
- **Create default requirement type items for missing sections** to create items for sections that are missing.

5. From the **Specification Type** drop-down list, select **Requirement Specification** and from the **Default Requirement Type** drop-down list, select **Requirement**, which is the default type when there is no match found for the supplied rules.
6. (Optional) To load saved rules, click the **Saved Rules** drop-down list.
7. (Optional) To create rules, click  next to **Rules** and complete the fields in the **Add Rules** panel:

- **Import As:** Select what to import the PDF file as, such as a requirement, paragraph, target, or verification requirement.
- Choose the parameters to follow regarding the conditions: **If any of the following conditions are met**, or **if all of the following conditions are met**. If you select any/all conditions, you are prompted to enter those conditions.
- **Condition:** Select this option to search the document for matching partial or exact strings.

Complete the following fields:

- a. Select **Words Contain Exact Match** to search the document for words exactly as you enter them or select **Words Contain Partial Match** to search for the entered words as substrings. For example if you enter **light**, and **electric**, Active Workspace returns only those documents that contain those words exactly as entered. However if you select **Partial Match**, Active Workspace returns documents that contain words such as brake-light, lightning, electrical, and electrics. In both cases, the object is imported as the type selected in the **Import As** list.

Note:

Consider the following when importing content with rules:

- Search words are case-insensitive.
- Paragraphs do not split during import, even if there are multiple keywords matching within the paragraph. The paragraph acts as one object.
- If your content contains a paragraph that follows with an image or an OLE object, then the paragraph and the image or OLE object import as one object.
- Embedded objects import as images.
- Each table is considered a single paragraph.
- Rules run in the order that you enter them.


- b. Enter the words to match, separated each word by a comma (,). The comma acts as an OR operator.
  - c. Click **Add**.
- **Property Rules:** Check this box to include property rules as part of the import process.

Note:

**Property Rules** are available when Requirements Management microservices are installed.


- a. **Set Property:** Choose the property from the list.
- b. Select the value for the property.

- c. Choose the parameters to follow regarding the conditions: **Always, If any of the following conditions are met**, or **if all of the following conditions are met**. If you select any/all conditions, you are prompted to enter those conditions.
8. (Optional) Perform any of the following:
    - To add another rule, repeat Step 7.
    - To save the rule, click **Add**, and name the rule. Select **Global Rule** for the rule to apply to all users.
  9. (Optional) Select the **Run in Background** check box to run the import process in the background.
  10. Click **Preview** to view the document prior to importing, or click **Import** to start the import process.

The import process begins and a scrolling bar at the top of the screen indicates the object ID and that the process is still running. Active Workspace sends you an alert  when the import process completes.

**Note:**

If the requirements are already loaded, then there is no way to determine if the import is complete. Manually refresh the page to determine if the import is complete.

11. Click the alert  and then select the **Requirement Manager** notification.
12. Click the specification name in the notification.

The specification opens in the **Table** view.

## Working with ReqIF Files

### ReqIF Overview

ReqIF is an XML schema used to exchange requirements, along with associated metadata, between tools from various vendors. This format also defines a workflow for transmitting the status of requirements between partners. While developed for the automotive industry, ReqIF is suitable for lossless requirement exchange in any industry.

Teamcenter supports the exchange or interchange of requirements specifications with external customers or suppliers using ReqIF technology. You can import ReqIF files to create a new structure in Teamcenter, or export the structure to a new ReqIF file.

Teamcenter currently supports ReqIF version 1.2 and the following file types:

- .reqif - plain ReqIF files without embedded objects
- .reqifz - archive that contains one ReqIF file with embedded objects

#### Import ReqIF Files

ReqIF is an XML file format used to exchange requirements and metadata between multiple software tools. You can import and export reqIF content to share requirements between partners and suppliers.

#### Import ReqIF file using a saved mapping template

1. From the **Home** tile, click **IMPORT SPECIFICATION**.
2. Click **ReqIF**.
3. Choose the file to import and click **Open**.
4. Choose a saved mapping from the list.
5. Click **Import**.

If the ReqIF file has multiple specifications, all specifications appear in the specified folder.

#### Import ReqIF files using run-time mapping


1. From the **Home** tile, click **IMPORT SPECIFICATION**.
2. Choose the file to import and click **Open**.
3. Do one of the following:
  - Select a **Saved Mapping** from the list.
  - Select a corresponding map type for the imported file from the list.

**Map Types** shows the mappings from the original requirement.

The screenshot shows the 'Import' dialog box with the 'ReqIF' tab selected. The 'MAP TYPES' section is expanded, showing the following mappings:

- Heading: SVItem
- ReqIF.ChapterName: Revision Independe...
- ReqIF.ForeignCreat: Revision Independe...
- ReqIF.ForeignID: Digital Signature Sta...
- Requirement: Requirement1
- ReqIF.ChapterName: Revision Independe...
- ReqIF.ForeignCreat: Date Modified
- ReqIF.ForeignID: Government Classifi...
- Status Of The Substance Declaration: Workflow
- Approved: (empty dropdown)
- Unapproved: (empty dropdown)
- Rejected: (empty dropdown)
- Under Review: (empty dropdown)
- Test Case: (empty dropdown)
- User Story: (empty dropdown)


An 'Import' button is located at the bottom of the dialog.

- (Optional) To save the mapping, click **Save Mapping** , and enter a name for the mapping. You can also click **Global Rule** to save this mapping for all users.
- (Optional) Click **Run in Background** to run the import process in the background.
- Click **Import**. The **Import** button appears after the mapping is complete for all types and required properties.

If the ReqIF file has multiple specifications, all specifications appear in the specified folder.

## Export ReqIF files

### Export ReqIF files using a run-time mapping

- Locate the requirement content** that you want to export, and then click **Open** .

The requirement content appears in the **Documentation** tab editor. The following graphic shows a sample requirement specification structure.

The screenshot displays a requirements management interface. On the left, a table lists requirements under the 'PRO-X Refrigerator Market Requirements' element. The table has columns for 'Element Name', 'ID', 'Revision', and 'Revision Name'. The 'Goals and Objectives' requirement (REQ-000785) is highlighted. On the right, a detailed view of this requirement is shown, including a title, a description, and a list of sub-requirements.

Element Name	ID	Revision	Revision Name
PRO-X Refrigerator Market Requirements	027175	A	PRO-X Refrigerator Market Requirements
Strategy and Overview	REQ-000784	A	Strategy and Overview
Goals and Objectives	REQ-000785	A	Goals and Objectives
Strategic Road Map	REQ-000786	A	Strategic Road Map
Customer Categories (User Profiles or Personas)	REQ-000787	A	Customer Categories (User Profiles or Personas)
Competitive Strengths and Weaknesses	REQ-000788	A	Competitive Strengths and Weaknesses
External Positioning	REQ-000789	A	External Positioning
Business Model	REQ-000790	A	Business Model
Affected Groups	REQ-000797	A	Affected Groups

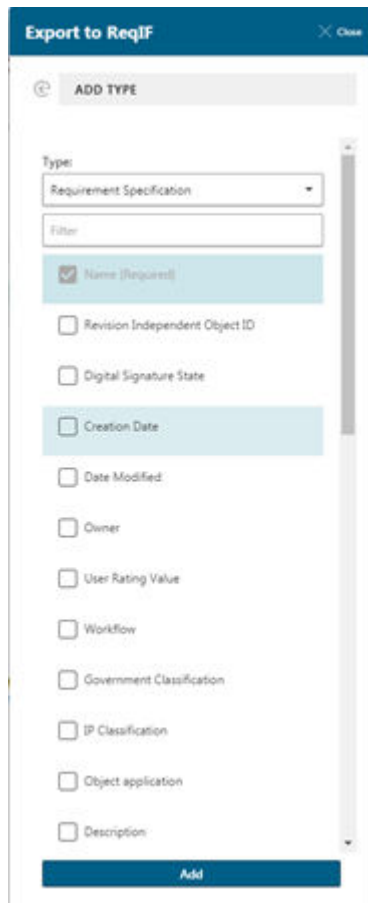
**027175-PRO- X Refrigerator Market Requirements**


**1 REQ-000784- Strategy and Overview**  
askkdjfas.l .alkj.ik a.lskdjirpuoihh. sllkdpoihj.ikj.ikThieq uick brown fox jumped over the lazy dog.

**1.1 REQ-000785- Goals and Objectives**  
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna seddf pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.  
Nunc viverra imperdiet enim. Fusce est. Vivamus a tellus. test test test fridge!  
Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin pharetra nonummy pede. Mauris et orci.  
Aenean nec lorem. In porttitor. Donec laoreet nonummy augue.  
Suspendisse dul purus, scelerisque at, vulputate vitae, pretium mattis, nunc. Mauris eget neque at sem venenatis eleifend. Ut nonummy.


2. Select **Export to ReqIF** from the top of the screen.
3. Type a name for the new ReqIF file.
4. Do one of the following:
  - Select a **Saved Configuration**.
  - Add a type.

To add a type, make your selections from the list and click **Add**.

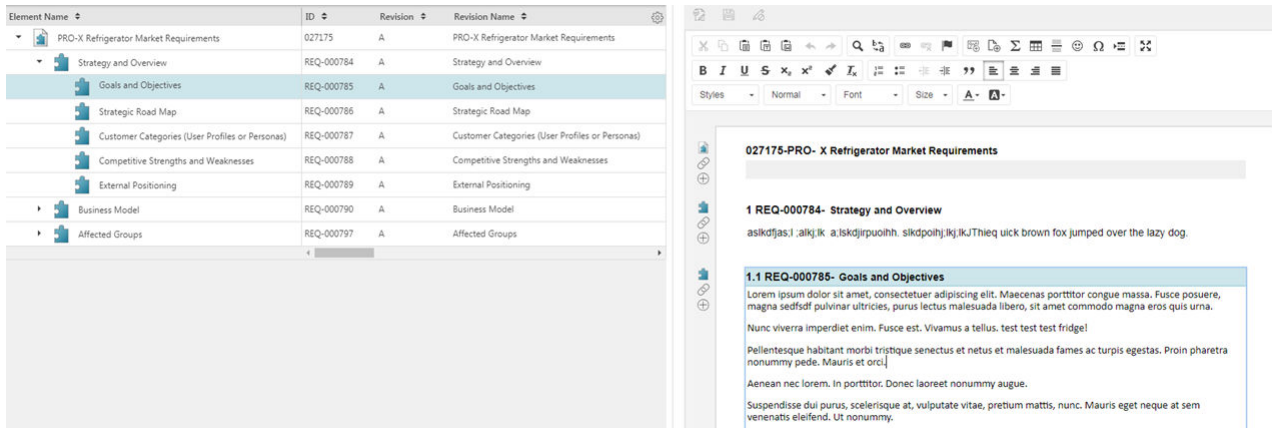


5. Follow the same process as step 4 to create a mapping for trace links.
6. (Optional) To save the mapping, click **Export Configuration** , and enter a name for the mapping. You can also click **Global Rule** to save this mapping for all users.
7. (Optional) Click **Run in Background** to run the import process in the background.
8. Click **Export**. The **Export** button appears once all of the Teamcenter types and their corresponding properties are selected.

### Export ReqIF files using a saved mapping template

1. **Locate the requirement content** that you want to export, and click **Open** .

The requirement content appears in the **Documentation** tab editor. The following graphic shows a sample requirement specification structure.



2. Select **Export to ReqIF** from the menu at the top of the screen.
3. Select a saved mapping from the **Saved Configurations** list.
4. (Optional) Add trace links by making selections from the list, and click **Add**.
5. Click **Export**.

### Note:

ReqIF export supports only editable numeric, string, Boolean and date properties.

### Export ReqIF files from search results

1. Using **Search**, locate the requirement revision.

### Note:

You must search for *requirement* revision, not specification revision.

2. Select the requirements to export.
3. Click **Share > Export to ReqIF**.
4. Do one of the following:
  - Select a **Saved Configuration**.
  - Add a type.

To add a type, make your selections from the list and click **Add**.

5. Click **Export**.

Note:

ReqIF export supports only editable numeric, string, Boolean and date properties.


## Authoring requirement and paragraph content

### Create requirement specifications

You can create the requirement components, including requirement specifications, requirements, and paragraphs. Requirement specifications are comprised of requirements, paragraphs, and even other specifications. You can add content such as text or graphics to the requirement using either a plain text editor, rich text editor, or Microsoft Word, depending on how your administrator has configured the environment. This procedure uses the term *requirement components* generically to refer to requirement specifications, requirements, or paragraphs. The procedure guides you through the most common process to create a requirement specification and then add requirements and paragraphs to the specification. However, you can perform the process in any order.

**Note:**

The custom notes functionality is currently not available in Active Workspace.

1. Display the folder to which you want to add the requirement specification, for example, your **Home** folder or other folder designated by your administrator to hold new specifications.
2. Click **More commands ...** > **New**  > **Create Specification**.

You can also click the **CREATE SPECIFICATION**  tile on your home page.

The **Create Specification** panel appears.

3. In the **Specification Type** field, select **Requirement Specification**.
4. Enter the properties for the new requirement specification, and then click **Add**.

The system creates the new requirement specification and opens it in the **Documentation** tab editor.

## About editing content in the Documentation tab or Microsoft Word

### High contrast accessibility option

When the *High Contrast* accessibility feature provided by Microsoft Windows operating systems is turned on, the editor removes all highlights, and the font color is changed to the default. The editor offers support for the High Contrast feature, but it is limited to Mozilla Firefox because other browsers do not support this feature. When this feature is enabled, the editor is rendered as clear lines and the toolbar buttons present text labels, instead of icon images.

### Editing options

Use the following table to determine your requirement editing options based on your requirement format:

If your requirement specification or requirements are in the following format	And you want to	Then do the following	Notes
Microsoft Word	Edit the content in the <b>Documentation</b> tab	Edit Microsoft Word content in the <b>Documentation</b> tab.	If you want to edit in HTML, contact your application administrator to convert

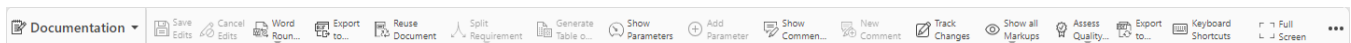
If your requirement specification or requirements are in the following format	And you want to	Then do the following	Notes
Microsoft Word	Edit the content in Microsoft Word	<b>Export and import the Microsoft Word requirements using the round-trip option.</b>	<p>the Word content to HTML permanently.</p> <p>The requirement content appears in a <b>Documentation</b> tab, depending on how the content was last edited and how your administrator has configured the editor. If the requirement content was last edited in Word or Excel, a message appears indicating you must edit the content in that application.</p>
Microsoft Word	Edit the content in the <b>Documentation</b> tab	Edit the content in the <b>Documentation</b> tab.	<p>The requirement content appears in a <b>Documentation</b> tab, depending on how the content was last edited and how your administrator has configured the editor. If the requirement content was last edited in Word or Excel, a message appears indicating you must edit the content in that application.</p>
Hybrid Microsoft Word and HTML	Edit the content in the <b>Documentation</b> tab	<b>Export and import the Microsoft Word requirements using the round-trip option.</b>	<p>If you want to edit in HTML, contact your application administrator to convert the Word content to HTML permanently.</p>
Microsoft Excel	Edit the content in the <b>Documentation</b> tab.	You can do the following:	<p>The requirement content appears in a <b>Documentation</b> tab,</p>

If your requirement specification or requirements are in the following format	And you want to	Then do the following	Notes
Microsoft Excel, round trip option		<ul style="list-style-type: none"> <li>• <b>Import the Microsoft Excel content into Active Workspace.</b></li> <li>• Paste the Microsoft Excel content into the <b>Documentation</b> tab editor and retain the Excel formatting.</li> </ul>	<p>depending on how the content was last edited and how your administrator has configured the editor. If the requirement content was last edited in Word or Excel, a message appears indicating you must edit the content in that application.</p>
PDF	Edit the content in the <b>Documentation</b> tab.	<p><b>Perform a Microsoft Excel round trip export and import.</b></p> <p><b>Import from PDF</b> and then edit in the <b>Documentation</b> tab.</p>	<p>The requirement content appears in a <b>Documentation</b> tab, depending on how the content was last edited and how your administrator has configured the editor. If the requirement content was last edited in Word or Excel, a message appears indicating you must edit the content in that application.</p>

## Documentation tab toolbars

### Documentation tab toolbar: Manage requirement content, parameters, and comments












Use the following toolbar commands to perform file actions such as save, export, and reuse. You can use the keyboard shortcuts Ctrl + Alt + T to access this toolbar in the **Documentation** tab.



The following commands are of special note:

**Note:**

Depending on the size of your browser window and display resolution, **More commands ...** command may be displayed on some of the results panel toolbar. Here, you can view additional commands that are not displayed on the toolbar due to size constraints.

- Use **Word Round-trip**  to export the content to Word for editing, and then reimport the content.
- Use **Export to ReqIF**  to export the content to the Requirements Interchange Format (ReqIF) XML schema.
- Use **Reuse Document**  to copy or derive a specification.
- Use **Split Requirement**  to divide the requirement content into separate requirements.
- Use **Generate Table of Contents**  to generate and manage a table of contents for the requirement specification.
- Use **Show Parameters**  to view and manage associated parameters in a table.
- Use **Show Comments**  or **New Comment**  to view, filter, and manage all associated comment threads.
- Use **Track Changes** to view markups of **edits to requirement text**.
- Use **Assess Quality Compliance**  to generate a quality matrix.
- Use **Export to PDF**  to save the requirement content to a PDF file.
- Use **Keyboard Shortcuts** to view a list of available keyboard shortcut keys.
- Use **Cross Reference**  to cross-reference information in a particular requirement to another part of a requirement specification without creating a trace link.

### Text Formatting toolbar: Format text and insert URL links

Use the following toolbar commands to author your requirements content. You can use the keyboard shortcuts Ctrl + Alt + E to access this toolbar in the **Documentation** tab.



- Use **Font Family** to apply a specific font to the selected text.
- Use **Font Size** to apply a specific font size to the selected text.

- Use **Bold** or use the keyboard shortcut Ctrl + B to bold or unbold on the selected text.
- Use **Italic** or use the keyboard shortcut Ctrl + I to italicize or remove italics from the selected text.
- Use **Underline** or use the keyboard shortcut Ctrl + U to underline or remove the underline from the selected text.
- Use **Strikethrough** or use the keyboard shortcut Ctrl + Shift + X to strike through the selected text.
- Use **Link** or use the keyboard shortcut Ctrl + K to add a URL to the selected text. The URL format must begin with **http://** or **https://**. Click an existing link to edit or delete the link.
- Use **Bullet List** to convert the selected paragraphs to a bulleted list, where each paragraph is a bullet point.
- Use **Numbered List** to convert the selected paragraphs to a numerical list, where each paragraph is a consecutive numeral.
- Use **Increase indent** to indent the selected text.
- Use **Decrease indent** to remove indentation from the selected text.
- Use **Font Background Color/Font Color** to select a color and apply it to or remove it from the selected text's background or the font itself. These colors appear in the resulting file if the object is exported to Microsoft Word.

Note:




You can also create a specification export template to apply background or font colors in the resulting file when the object is exported to Microsoft Word. See [Create a requirement specification export template](#) for information on creating a specification export template. See [Export and import Word requirements using round-trip](#) for information on applying a specification export template when exporting to Microsoft Word.

- Use **Text alignment** to align the selected paragraphs to the left margin, right margin, or center or to make it justified.
- Use **Insert table** to draw the table dimensions (rows and columns) and insert into the selected location.
- Use **Superscript/Subscript** to toggle the selected text to either of these formats.
- Use **Undo/Redo** or use the keyboard shortcuts Ctrl + Z to undo or Ctrl + Y to redo one or more actions.
- Use **Find and replace** or use the keyboard shortcut Ctrl + F to find and replace text. Enter text and click **Find** on the **Find and replace** dialog box to locate text in the document. If you want to replace the located text, enter the replacement, and then click **Replace** or **Replace All**.

## Text Formatting toolbar: Insert graphics, OLE objects, and equations

Use the following section of the toolbar commands to add or insert images and equations.



- Use **Insert Image**  to select a graphic file in a common format. Click an image in the editor to edit the alt text (tooltip text that is displayed when hovering the mouse over the image), adjust the alignment, or add a caption.
- Use **Insert OLE**  to select any Object Linking and Embedding (OLE) file, such as a Microsoft Excel or PowerPoint file, which appears as an icon. Double-click the icon to download the file.
- Use **Insert Equation**  or use the keyboard shortcut Ctrl + M to enter an equation (sometimes referred to as formulas).

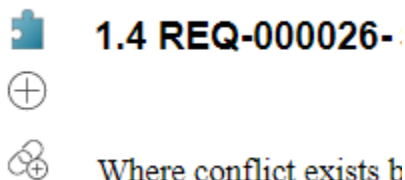
You cannot copy/cut and paste equations from a Microsoft Word document into the editor; however, you can import and export equations from Microsoft Word.





Note:

You can only edit an equation in the authoring tool in which the equation was created.

## Documentation tab toolbar: Add requirements and trace links

Use the following toolbar commands (located within the editor on the left-hand side) to manage requirements directly in the editor.





- Use the **Requirement Revision**  to perform operations on the selected requirement, such as **remove**, **copy**, **move**, and **freeze**.
- Use **Add**  or use the keyboard shortcuts Ctrl + Enter to create a sibling or Shift + Enter to create a child requirement directly in the editor.
- Use **Trace Link**  or use the keyboard shortcut Alt + L to create trace link from a requirement or **Trace Link**  to manage an existing trace link.

#### Keyboard navigation in the Documentation tab

You can navigate the **Documentation** tab user interface using specific keyboard shortcuts. As you navigate through the page, the area highlights to show your location.

**Keyboard Shortcuts**

 Reset
  Close

**▼ Navigation**

Move to Documentation:	Ctrl + Shift + E
Text Formatting Toolbar:	Ctrl + Alt + E (Arrows to move)
Documentation Toolbar:	Ctrl + Alt + T (Tab to move)
Next Requirement:	Tab
Previous Requirement:	Shift + Tab
Next Line:	Arrow keys (up/down)

**▼ Authoring**

Add Sibling:	Ctrl + Enter
Add Child:	Shift + Enter
Change Requirement Type:	Alt + T
Create Trace link:	Alt + L
Insert URL:	Ctrl + K
Insert Equation:	Ctrl + M

**▼ General Editing**

Save:	Ctrl + S
Copy Text:	Ctrl + C
Paste Text:	Ctrl + V
Bold:	Ctrl + B
Italic:	Ctrl + I
Underline:	Ctrl + U
Strikethrough:	Ctrl + Shift + X
Undo:	Ctrl + Z
Redo:	Ctrl + Y
Find and Replace:	Ctrl + F

You can perform certain specific requirement functions, using the following keyboard shortcuts:

To	Use this
<p>Navigate or focus on the first visible requirement object title in the <b>Documentation</b> tab editing area.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><b>Note:</b></p> <p>This shortcut works only if the focus is outside the <b>Documentation</b> tab editing area.</p> </div>	Ctrl + Shift + E
Access or navigate to the text formatting toolbar of <b>Documentation</b> tab from any location on the user interface.	Ctrl + Alt + E
Access or navigate to the <b>Documentation</b> tab toolbar from any location on the user interface.	Ctrl + Alt + T
Navigate between lines in a requirement.	The up arrow and the down arrow keys.
Create a sibling.	Ctrl + Enter
Create a child requirement.	Shift + Enter
<p>Modify or change the requirement type.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p><b>Note:</b></p> <p>This shortcut works only when the focus is on a new requirement object.</p> </div>	Alt + T. This displays a list of relevant new requirement object types. Press Tab key to select the desired object type.
Create a trace link.	Alt + L
Navigate between requirements.	<p>Do any of the following:</p> <ul style="list-style-type: none"> <li>• Tab. This enables you to move to the next requirement.</li> </ul>

To	Use this
	<ul style="list-style-type: none"> <li>• Shift + Tab. This allows you to move to the previous requirement.</li> </ul>
Access or navigate to the drop-down list for a command. For example, navigate to the desired command in the <b>More commands</b> ... list.	<ol style="list-style-type: none"> <li>1. Enter or the Spacebar. This displays the drop-down list.</li> <li>2. The up and down arrow keys to move to the desired option in the list.</li> <li>3. Enter. This enables you to access the desired command or option.</li> </ol>

## Edit requirement content in the Documentation tab

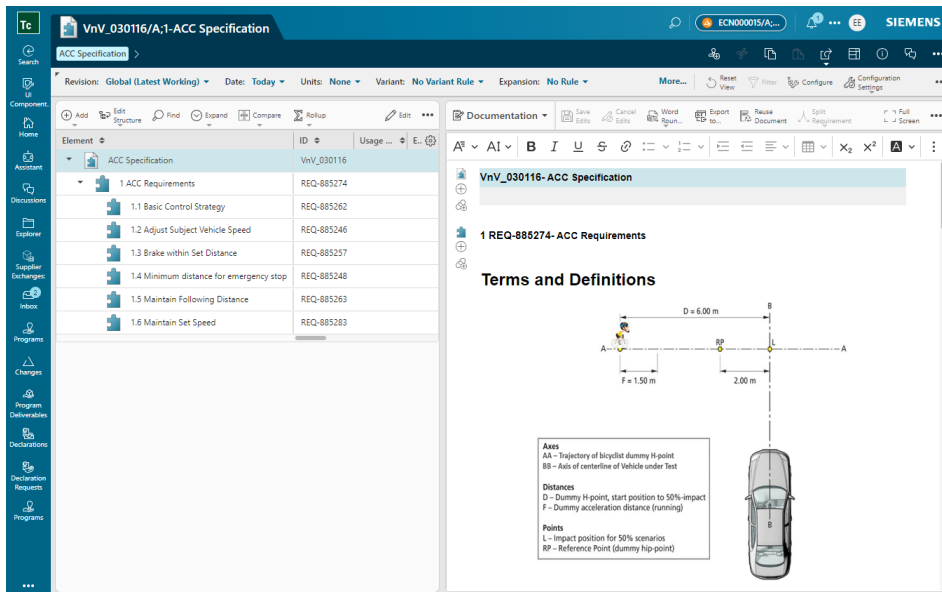
You can edit requirement content in the **Documentation** tab. The requirement source format can be plain text, rich text (not created in Microsoft Word), and HTML. For information about Microsoft Word support, see [Edit requirements in Microsoft Word](#).

### Note:

Control characters are not supported on the Documentation tab for non-UTF8 environments. Setting the environment variable POM\_STRIP\_CTRL\_CHARS enables the application to remove the control characters and save the data to the database. This is only applicable for non-UTF8 environments. For more information, see Teamcenter Environment Variables.

1. **Locate the requirement content (requirement specifications, requirements, or paragraphs)** that you want to edit, and then click **Open** .

The requirement content appears in the **Documentation** tab editor. The following graphic shows a sample requirement specification structure.



**Note:**

Use the tree controls in either the requirements list or the **Documentation** tab editor to expand or collapse the requirement structure.

## 2. Edit the requirement content.




Consider the following situations when editing content:

- The requirement content appears in a rich text editor, depending on how the content was last edited and how your administrator has configured the editor.
- Teamcenter does not accept some special characters in requirement text. One example is the ₪ symbol for the Israeli New Shekel. When these characters are used in requirement text, the specification content and documentation tabs appear blank.
- If the selected requirement (or requirement specification) has children, the children are also entered into edit mode.
- When adding new requirements or trace links, you can promote/demote or move items up and down. Changes you make in the **Documentation** tab are reflected in the tree view.
- Selecting an object in the tree also selects it in the **Documentation** tab, and vice-versa.
- If you attempt to update a requirement that another user has updated, Active Workspace displays a warning. You can save the changes locally and apply them later.

- You can undo changes up to the last saved version of the requirements document, using the keyboard shortcuts Ctrl + Z to undo or Ctrl + Y to redo (up to 20 actions) until you exit the document. You can also use the toolbar buttons. The undo/redo actions are valid until you exit the requirement.

**Note:**

If you added a new requirement object using the **Add** panel, that action cannot be undone.

- You can move sections up or down by right-clicking on the sidebar icons in the tree view.
- Clicking **trace link**  in the **Documentation** tab or using the keyboard shortcut Alt+L launches the **Trace Link** panel. The object where you clicked the trace link is the starting point. You can drag the end point from the tree structure, or click  next to the end object in the documentation tab.
- A bold **trace link**  indicates the requirement has an associated trace link. Mouse over the trace link icon to see the content the requirement is linked to.

**Note:**

If any content is not editable, which is indicated by highlighted text, hover over the text to display a tooltip that explains the reason.

The following graphic shows read-only and editable content:


**SubRequirement A**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

**SubRequirement B**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, s

In the example above, shaded content is read-only, and unshaded content is editable.

- Click **Save Edits**  or use the keyboard shortcut Ctrl + S.

If you edit requirement content and then attempt to change pages, the editor prompts you to save or discard the changes.

## Track changes to requirement text

You can use the **Track Changes** option to view markups of changes to requirement text. You can view any modified or reformatted textual content directly within the Documentation tab as well as tracking each change inline with the specific requirement. Multiple users can collaborate on every requirement

to determine consistency and accuracy according to company guidelines. In addition, when using comments along with the track changes, global collaboration can be easily coordinated.


The **Track Changes** option marks edits as they are made directly within the rich content of the requirement and the requirements specification. As you type or edit the existing text, a red color appears to indicate a deletion, while a green color indicates an addition. Formatting changes are indicated by a bounding box with blue lines above and below the text. This is because these changes may include many different formatting styles.

The **Comments and Changes** panel that appears when **Show Comments** is selected updates dynamically to indicate all inline edits. You can click **Tracked Changes** on the **Comments and Changes** panel to display a list of edits, and then accept or discard the edits as necessary. As changes are accepted, they are updated directly within the selected requirement and are saved to the specification.

#### Prerequisites

For track changes functionality, you must have Teamcenter 14.2 or higher installed.

#### Procedure

1. From the **Home** screen, click **Explorer**  icon from the left panel and open the specification where you want to edit text.
2. Click **Track Changes** on the **Documentation** tab.
3. Click **Show Comments and Changes**.

The **Comments and Changes** panel appears.




4. Edit the requirement text as necessary.

The changes appear in the text and in the **Comments and Changes** panel.

5. (Optional) Do one of the following:
  - Click **Accept Suggestion** to accept the specified change.
  - Click **Discard Suggestion** to discard the specified change.
  - Click **Review** and then accept or reject all changes at once or only the selected changes.





#### Edit requirements in the Summary Table tab

The **Summary Table** tab provides an overview of the requirement structure by paragraph for the specification in a tabular format. You can view and edit content such as trace links, comments, parameters, revision name, and so on.

1. **Activate Requirement Manager workspace mode.**
2. **Locate the requirement content (requirement specifications, requirements, or paragraphs)** that you want to edit, and then click **Open** .
3. Click the **Summary Table** tab.
4. Notable tasks include the following:
  - Click in editable fields such as **Revision Name** to directly change content.
  - Mouse-over a **Text** field, and then click **Edit**  to open the **Documentation** dialog to edit the requirement in a simplified version of the **Documentation**.
  - Click a **Trace Link**  icon to create a trace link. The number next to the trace link indicates the number of trace links associated with the paragraph. Click the number to open the **Existing Trace Link** panel to manage the trace links associated with the paragraph.

Note:

You can use the keyboard shortcut Alt + L to create a trace link.

- Click a comment  to manage comments associated with the paragraph in the **Comments panel**.
- Click an attachment  to manage attachments associated with the paragraph in the **Attachments** panel.
- Click a parameter  to display the parameter in the **PARAMETERS** table below the summary table.
- Click a suspect link  to open the **Review Suspect notification in your Inbox**.

## Edit requirement properties in the Information panel

You can edit requirement summary information in the **Information** panel.

1. **Locate the requirement content (requirement specifications, requirements, or paragraphs)** that you want to edit, and then click **Open** .
2. Click **Information** .

The requirement properties appear in the **Information** panel.

An *occurrence* is the repeated use of a requirement using different occurrence names. For example, you could have a requirement for "Temperature - High" and then create copies that are occurrences that represent "Temperature - Medium" and "Temperature - Low."


3. Click **Edit** and make any changes to the **Revision Name** or **Description**.
4. Click **Save**.

#### Edit the properties of requirement content in the Tree view



You can edit the properties of requirement specifications, requirements, and paragraphs, in the **Tree** view or the **Tree with Summary** view.

1. **Locate the requirement content (requirement specifications, requirements, or paragraphs)** that you want to view, and then click **Open** .

The requirement content appears in the **Documentation** tab editor.


2. Select either the **Tree** or **Tree with Summary** view.
3. Click **Edit** .

Editable properties become active.

4. Double-click an editable field to modify the value.
5. Do one of the following:
  - To save your changes, click **Save Edits**  or use the keyboard shortcut Ctrl + S.
  - To cancel your changes, click **Cancel Edits** .

#### Create a requirement baseline

A requirement *baseline* is a snapshot of a requirement. You often create baselines to preserve the state of a requirements structure at a particular checkpoint. You can audit an updated requirement by **comparing the requirement** to its baseline.

1. **Locate the requirement** that you want to create a baseline for, and then click **Open** .


The requirement content appears in the **Documentation** tab editor.

2. Click **More commands**  > **New**  > **Create Revision Baseline**.

The **Create Revision Baseline** panel opens.

3. Select a baseline template from the drop-down menu.
4. (Optional) Select the **Precise Baseline** check box.
5. Click **Create**.

Your saved requirement baseline is now available in your **Home folder** and you receive a notification in the **Alerts** panel.

6. Click **Alerts**  and click the baselined requirement in the **Target Object** section.

**Warning:**

You must have the AsyncService translator installed, configured, and running for the baseline feature to operate properly.

**Note:**

Once a baseline is created, you cannot create or delete tracelinks on the baseline version, if the defining end is the baseline. You can use the constant **Fnd0PreventTraceLinkCreate** to control creation and deletion:

**If the constant value is NONE:**

Source/ Defining Object	Target/Complying Object	Create Trace Link?	Delete Trace Link?
Baselined item	Non-baselined item	Yes	Yes
Baselined item	Baselined item	Yes	Yes
Non-baselined item	Non-baselined item	Yes	Yes
Non-baselined item	Baselined item	Yes	Yes

**If the constant value is BOTH:**

Source/ Defining Object	Target/Complying Object	Create Trace Link?	Delete Trace Link?
Baselined item	Non-baselined item	Yes	Yes
Baselined item	Baselined item	No	No

Source/ Defining Object	Target/Complying Object	Create Trace Link?	Delete Trace Link?
Non-baselined item	Non-baselined item	Yes	Yes
Non-baselined item	Baselined item	Yes	Yes

If the constant value is **PRIMARY**:

Source/ Defining Object	Target/Complying Object	Create Trace Link?	Delete Trace Link?
Baselined item	Non-baselined item	No	No
Baselined item	Baselined item	No	No
Non-baselined item	Non-baselined item	Yes	Yes
Non-baselined item	Baselined item	Yes	Yes

If the constant value is **SECONDARY**:


Source/ Defining Object	Target/Complying Object	Create Trace Link?	Delete Trace Link?
Baselined item	Non-baselined item	Yes	Yes
Baselined item	Baselined item	Yes	Yes
Non-baselined item	Non-baselined item	No	No
Non-baselined item	Baselined item	No	No

### Generate and manage a table of contents


You can insert, update, or delete a table of contents within any requirement or paragraph under the following conditions:

- The top-level object of the structure must be a requirements specification.
- The cursor must be placed in the content field for the requirement or paragraph.
- You can have only one table of contents per requirements specification.

When generating a table of contents, you can also include a list of tables or a list of figures.

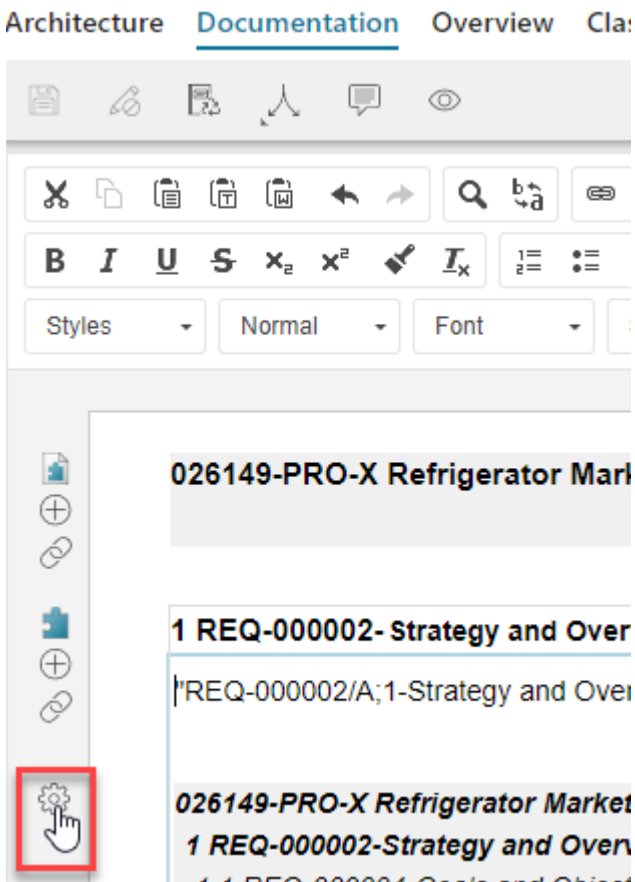
1. **Locate the requirement or paragraph** that you want to create a table of contents for, and then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Move the cursor to an area in the requirement or paragraph where you want to insert the table of contents.
3. Click **Generate Table of Contents** .

The table of contents appears.


4. (Optional) To edit or remove the table of contents, click **Table of Contents** and then select either **Update** or **Delete**.

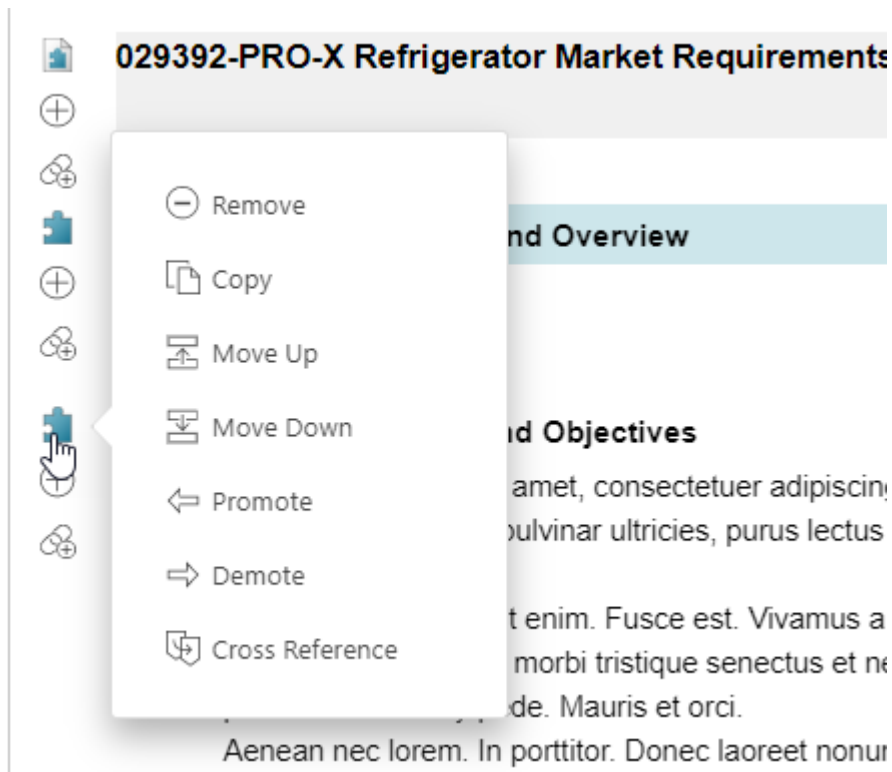


5. Save the requirement to retain the table of contents.

## Create a cross reference within a requirement

You can cross-reference information in a particular requirement to another part of a requirement specification without creating a trace link, allowing you to link to information in other parts of the document.

1. **Locate the requirement specification** that you want to view, then click **Open** .
2. Right-click the object type icon in the left side bar of the **Documentation** tab and select **Cross Reference**.





3. Navigate to another area in the same specification, or open another requirement specification.
4. Navigate to the location for the cross-reference, right-click, and paste the copied information into the content.

The **Show Cross-reference as:** dialog appears.

5. Select how you want the cross reference to appear and then click **Paste**.

## Show Cross-reference as:

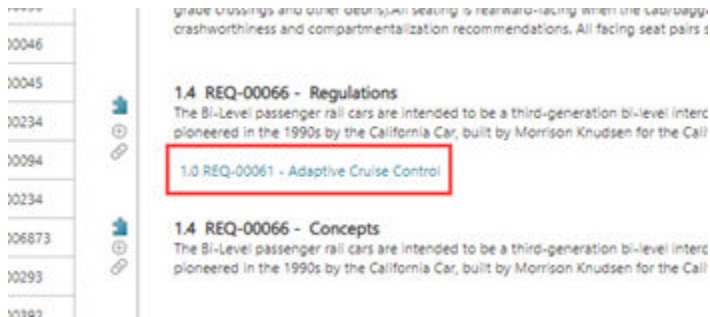
- 1.3.1 : Goals and Objectives
- 1.3.1 REQ-000132 Goals and Objectives
-  1.3.1 : Goals and Objectives
-  1.3.1 REQ-000132 Goals and Objectives

Custom

Enter String value in Textbox:

Default String

Paste



1.4 REQ-00066 - Regulations  
The Bi-Level passenger rail cars are intended to be a third-generation bi-level intercity pioneered in the 1990s by the California Car, built by Morrison Knudsen for the CalP

1.0 REQ-00061 - Adaptive Cruise Control

1.4 REQ-00066 - Concepts  
The Bi-Level passenger rail cars are intended to be a third-generation bi-level intercity pioneered in the 1990s by the California Car, built by Morrison Knudsen for the CalP

The cross-referenced information appears in the specification in the chosen style.

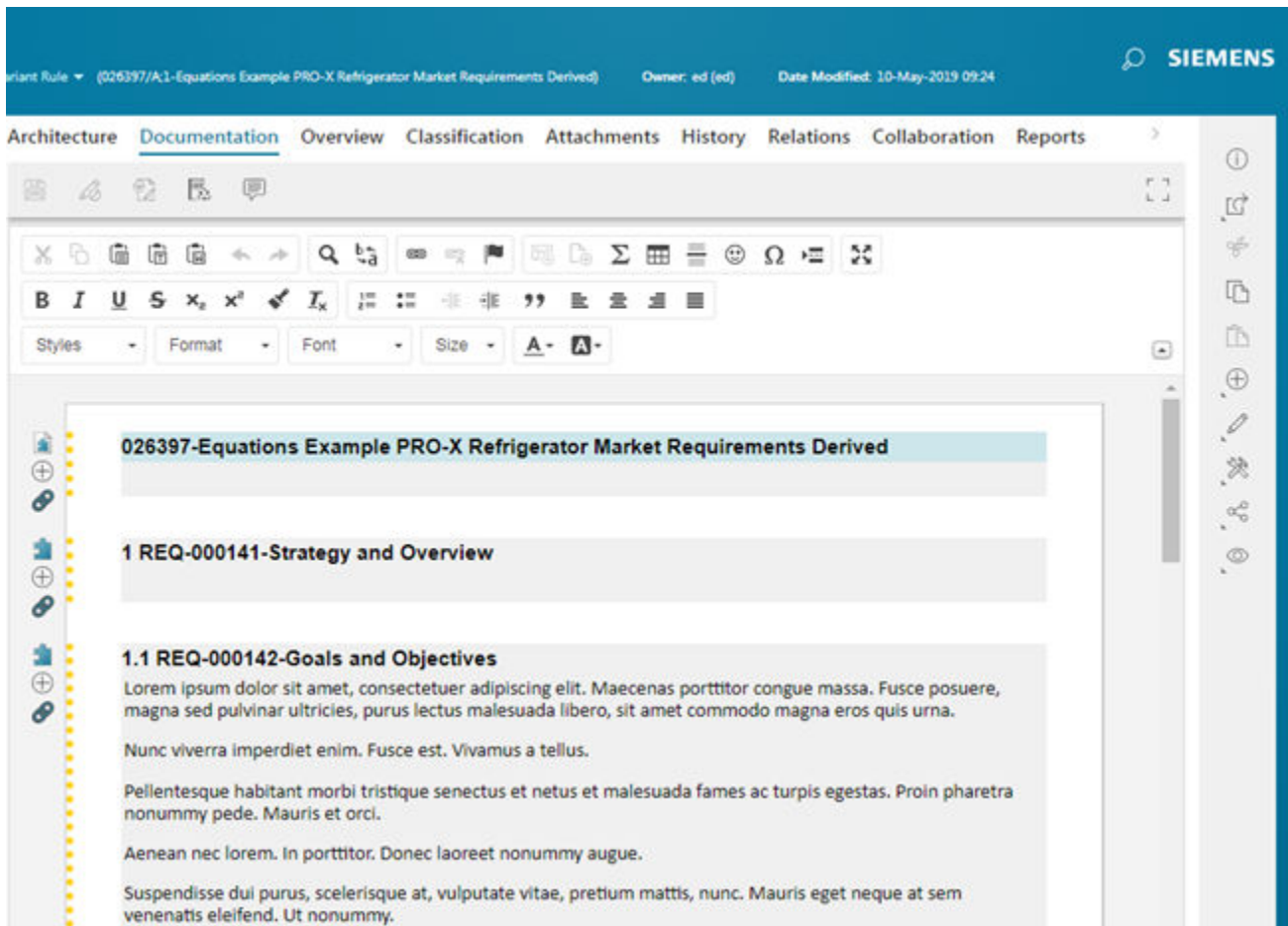
## Copying and deriving requirements

### Differences between copied and derived specifications

You can choose to copy or to derive specifications. Standalone specifications allow you to make changes to the document, while derived specifications offer limited editing capabilities. The following table shows the differences:

Capability	Standalone Specification Copy	Derived Specification
Creates new document and work item underneath.	Creates both a new requirement specification and the objects underneath.	Creates both a new requirement specification and the objects underneath it in non-editable mode and with derived relationship.
Copy links from parent.	Yes.	No. Existing links remain the same.
Additional links created.	Optional, although additional links can be created.	No. Existing and new requirements are linked with a derived-from relation.
Read-only or editable.	Both the new and existing requirement specification, and their objects, are editable.	New requirement specification and the objects underneath are read-only.
Auto-sync with source document.	N/A.	Yes.
Freeze/unfreeze/overwrite individual object.	N/A.	Yes.



A derived specification appears as follows in the **Documentation** tab:



The title contains the word **Derived**, and the yellow dots along the side indicate referenced content.

## Copy or derive a specification

You can choose to copy or derive specifications. Standalone specifications allow you to make changes to the document, while derived specifications offer limited editing capabilities.

1. **Locate the requirement content (requirement specifications, requirements, or paragraphs)** that you want to copy, and click **Open** .
2. In the **Documentation** tab, click **Reuse Document** .

The **Reuse Document** panel appears.

3. Select **Create a Copy** or **Create a Derived Document**. See **Differences between copied and derived specifications** to understand the differences between each document type.
4. Enter a name for the copy or derived document.

5. (Optional) Select the **Run in Background** check box. If you select this option, you receive a notification when the operation completes.
6. Click **Reuse**.


The new document appears in the **Documentation** tab.

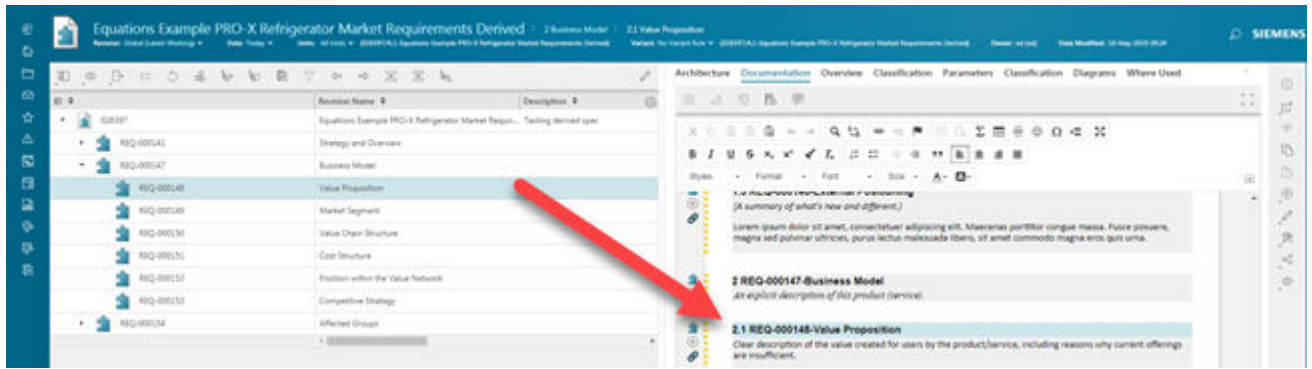
### Freezing, unfreezing, and overwriting derived requirement objects

Occasionally, you may find it necessary to freeze, unfreeze, or overwrite objects in a derived requirement specification.

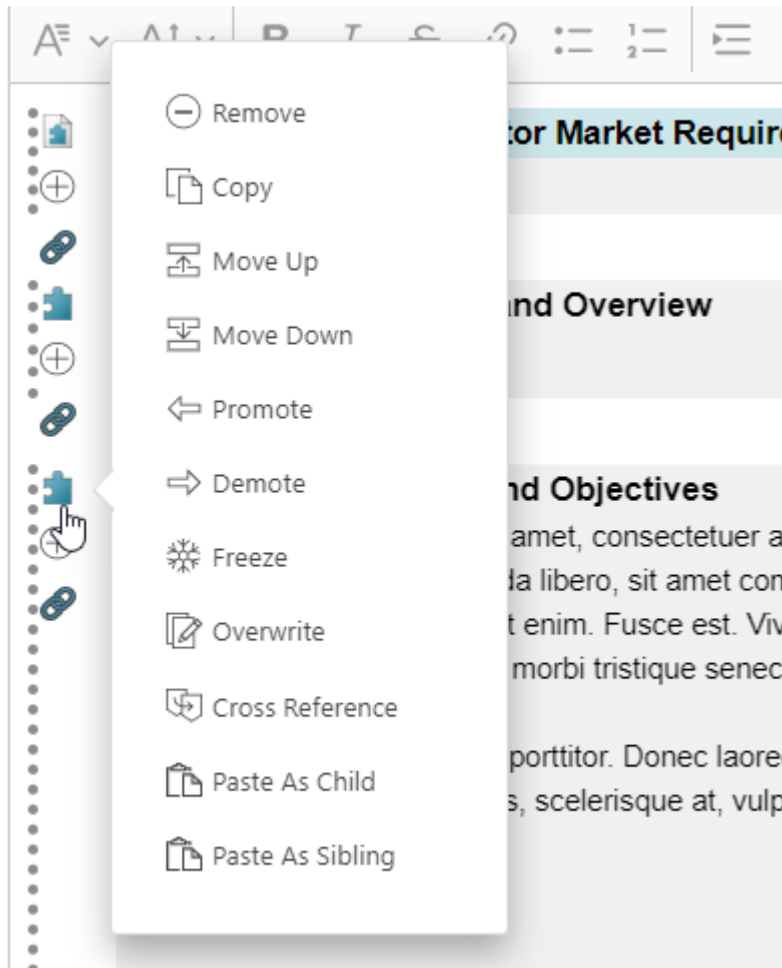
These capabilities are:

Capability	Description	Bar state and color
<b>Freeze</b>	Select <b>Freeze</b> to the <b>latest revision</b> of a source requirement object. Once frozen, changes to the source requirement content are not reflected in the derived specification.	Blue dotted
<b>Freeze</b>	Select <b>Freeze</b> to an <b>earlier revision</b> of a source requirement object. Once frozen, changes to the source requirement content are not reflected in the derived specification.	Orange dotted
<b>Unfreeze</b>	When an object is unfrozen, any details updated in the source document after that point are retained.	Grey dotted
<b>Overwrite</b>	Selecting <b>Overwrite</b> breaks the relation between the source and derived objects. Objects in both specifications can be edited and are independent of each other.	Solid yellow

1. **Locate the derived requirement content** that you want to edit, and then click **Open** .
2. From the left side of the screen, select one or more objects to change. It is highlighted in the **Documentation** tab.



3. Right-click **Requirement Revision**  and make a selection.



4. Select one of the following:
  - To freeze an item, select the revision you want, and click **Freeze**.

**Freeze**
✕ Close

Revision

Name	Revision	Version	Last Modified User	Date Modified	
REQ-000135-Strategic Road M...	A	2	ed (ed)	13-May-2021 07:06	

Preview

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas porttitor congue massa. Fusce posuere, magna sed pulvinar ultricies, purus lectus malesuada libero, sit amet commodo magna eros quis urna.

- To unfreeze or change the frozen content to another revision, do one of the following:
  - Select **Freeze** and then select another revision to freeze to.
  - Select **Unfreeze** and the dialog automatically changes to the latest revision.
- If you select **Overwrite**, a solid yellow line appears to the left and the content can now be edited.

**2.1 REQ-000148- Value Proposition**

Clear description of the value created for users by the product/service, including reasons why current offerings are insufficient. Adding new content after overwrite.

5. **Save** any changes made.

### Merging a derived requirement with a master requirement

In some instances, you may want to merge a derived specification back into the original specification. When a specification is derived, items can be added or overwritten, and occasionally you may want to merge these changes back into the original specification.

Note:

Consider the following when working with merged requirements:


- Properties that you update in the master requirement are not updated in the derived requirement.
- Structure updates that you make in the derived or master specification cannot be compared or merged.
- You can update properties for frozen requirements in a derived requirement; however, you cannot edit the requirement text itself.
- Any requirement content having track changes either in master or derived specification is not allowed to merge.

1. Locate and open the derived specification.

In the **Documentation** tab, the specification appears with the following color coding:

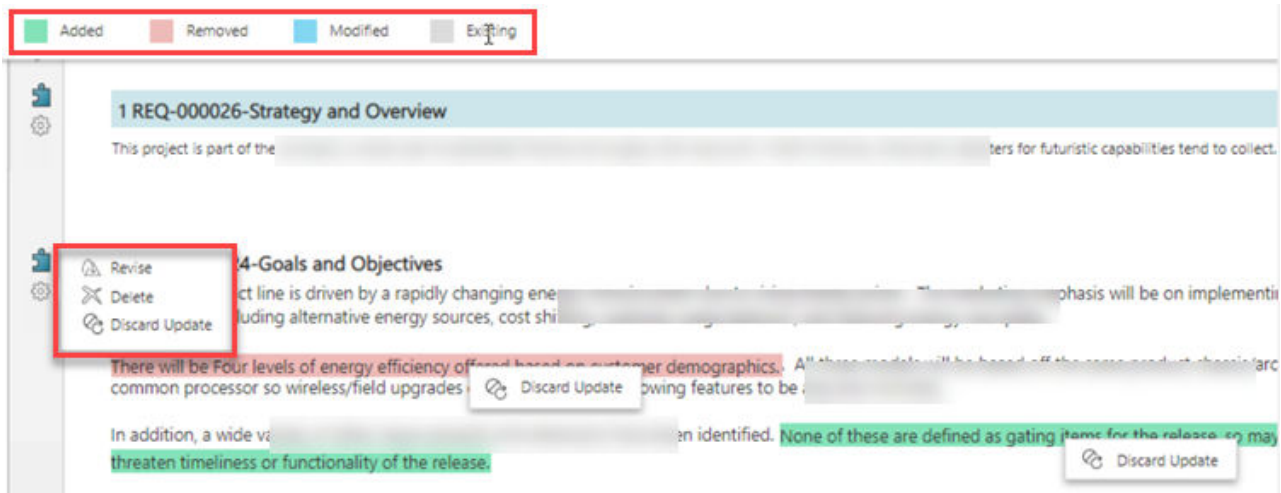
- Green: Content added.
- Red: Content removed.
- Blue: Content modified.
- Black: Existing content.

Hovering over the highlighted area shows a tooltip that indicates if the content is frozen to the latest or non-latest revision.

2. Choose **Merge**  to merge the updates with the base document.

The merge preview launches, where you can discard or accept updates to merge with the base document.

3. Right-click in the context commands to discard or accept updates.



- Discarding removed or added text: If you select **Discard Update** on a removed or modified line of text, the highlight changes to grey and a message appears to accept the update. You cannot discard modified text (highlighted in blue).
  - Discarding added text: If you choose to discard an update containing added text, the highlight turns red to show it is not part of the base document. Right-click to accept the change.
4. Click **Merge** to merge the accepted changes to the base document, or **Revise** to revise the base document with the new updates.

## Comparing requirement properties

When requirements are merged, you can view or change the merged text and properties. You can also accept or discard changes to these properties before the requirement is finalized.

1. Locate and open the merged specification.

In the **Documentation** tab, the specification appears with the following color coding:

- Green: Text was added.
- Red with strike-through: Text that was removed.
- Blue: Text that was modified.
- Black: Current existing text.

The screenshot shows a requirement specification interface. At the top, there is a legend with four colored squares: green for 'Added', red for 'Removed', blue for 'Modified', and grey for 'Existing'. Below the legend, the requirement is titled '1 REQ-000026-Strategy and Overview'. The main text describes the project's goal to penetrate financial and supply chain accounts in North America. Below the text is a 'PROPERTIES' table with the following data:

Property Name	Property Value
Description	Goals and Objectives
IP Classification	Secret
Priority	2
Technical Capability	In Progress

Below the table, there is a sub-section titled '1.1 REQ-000024-Goals and Objectives'. The text in this section is color-coded: some parts are green (added), some are red with a strike-through (removed), and some are blue (modified). The text describes energy efficiency levels and improvements.

### Note:

The **PROPERTIES** table values are set by your system administrator in the **REQ\_compare\_properties\_for\_merge** preference. To view complete properties, click **Information** ⓘ on the right-hand menu.

2. Perform any of the following actions:

- To revise, delete, or update an entire requirement object, click , and then select the following:

- **Revise** to update the master (source) document with changes to this requirement.
- **Delete** to remove this requirement from the master (source) document.
- **Discard Update** to remove any changes made to this requirement from the master (source document).
- **Update** to undo a **Discard Update**.

Note:

The **Action** column in the results panel indicates the action taken.

- To discard a specific change from the master (source) document, right-click highlighted (green, red, or blue) content and select **Discard Update**.
- To apply a specific change to the master (source) document, right-click grey (**Existing**) content and select **Accept Update**.

The base document property is shown, and you are prompted to **Accept Update** to revert back to the original value.

3. Click **Merge** to accept changes to the existing base document or **Revise** the base document with the updates.

## Modify requirements structure

### Create and add requirements or paragraphs to requirement specifications

You can create and add requirements and paragraphs to existing requirement specifications.

Note:

The custom notes functionality is currently not available in Active Workspace.

1. **Locate the requirement specification** that you want to view, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Click **Add** , and then select **Sibling** or **Child**, if you have an existing requirement selected.

Note:


You can use the keyboard shortcuts Ctrl + Shift + Enter to add a child or Ctrl + Enter to add a sibling.

3. Click **Save Edits** or use the keyboard shortcut Ctrl + S.


Active Workspace creates the new requirement or paragraph and adds it to the view on the left.

## Split requirements

You can divide requirements content into separate requirements:

1. **Locate the requirement specification or requirement** that you want to split, and then click **Open** .



The requirement content appears in the **Documentation** tab editor.

2. Select the content that you want to split.
3. Click **Split**  and then select **Child** or **Sibling**.

Active Workspace creates the child or sibling with the selected content.


## Update the requirements structure using paragraph numbers

You can update the requirement structure by adjusting the related paragraph numbers for the requirement content. You can move topics within a parent, make a parent-child relationship, or move a child from one parent to another. The paragraph numbers appear in both the requirement structure tree in the left pane and in the **Documentation** tab editor.


1. **Locate the requirement specification (or other requirement)** that you want to update, then click **Open** .
2. Select the **Tree view**.
3. Click **Reorder** .

The **Paragraph Number** fields become editable.

### Note:

If you do not see the **Paragraph Number** column, click  and then click **Arrange** to ensure the column is selected to appear. You can also scroll your view to the right to locate the column and move it to a more convenient location.

Element Name	ID	Paragraph Number
▼ Strategy and Overview	REQ-000021	1
Goals and Objectives	REQ-000022	1.1
Strategic Road Map	REQ-000023	1.2
Customer Categories (User Profiles or Personas)	REQ-000024	1.3
Competitive Strengths and Weaknesses	REQ-000025	1.4
External Positioning	REQ-000026	1.5
▼ Business Model	REQ-000027	2
Value Proposition	REQ-000028	2.1
Market Research	REQ-000029	2.2

4. Update the paragraph numbers to change the content order, and then click **Edit**  > **Save Edits** or use the keyboard shortcut Ctrl + S.

Element Name	ID	Paragraph Number
▼ Strategy and Overview	REQ-000021	1
Goals and Objectives	REQ-000022	1.1
Strategic Road Map	REQ-000023	1.3
Customer Categories (User Profiles or Personas)	REQ-000024	1.2
Competitive Strengths and Weaknesses	REQ-000025	1.4
External Positioning	REQ-000026	1.5

**Note:**

It can take several seconds for the structure to refresh with the updated content order.

Element Name	ID	Paragraph Number
▼ Strategy and Overview	REQ-000021	1
Goals and Objectives	REQ-000022	1.1
Customer Categories (User Profiles or Personas)	REQ-000024	1.2
Strategic Road Map	REQ-000023	1.3
Competitive Strengths and Weaknesses	REQ-000025	1.4

Consider the following:

- To make a topic a child, add another decimal value. In the following example, you want to make **Competitive Strengths and Weaknesses (1.4)** a child of **Strategic Road Map (1.3)**.

Element Name	ID	Paragraph Number
▼ Strategy and Overview	REQ-000021	1
Goals and Objectives	REQ-000022	1.1
Customer Categories (User Profiles or Personas)	REQ-000024	1.2
Strategic Road Map	REQ-000023	1.3
Competitive Strengths and Weaknesses	REQ-000025	1.4
External Positioning	REQ-000026	1.5

You edit the order and update **Competitive Strengths and Weaknesses** from **1.4** to **1.3.1** and save.

Strategic Road Map	REQ-000023	1.3
Competitive Strengths and Weaknesses	REQ-000025	1.3.1

The structure refreshes showing the new parent-child relationship:

▼ Strategic Road Map	REQ-000023	1.3
Competitive Strengths and Weaknesses	REQ-000025	1.3.1

- To move a child topic from one parent to another, change the topic numbering to match the new parent. In the following example, you want to make **Goals and Objectives (1.1)** a child of **Business Model (2)**.

Element Name	ID	Paragraph Number
▼ Strategy and Overview	REQ-000021	1
Goals and Objectives	REQ-000022	1.1
Customer Categories (User Profiles or Personas)	REQ-000024	1.2
Strategic Road Map	REQ-000023	1.3
External Positioning	REQ-000026	1.4
▼ Business Model	REQ-000027	2
Value Proposition	REQ-000028	2.1
Market Segment	REQ-000029	2.2
Value Chain Structure	REQ-000030	2.3

You edit the order and update **Goals and Objectives** from **1.1** to **2.1** and save.

▼ Strategy and Overview	REQ-000021	1
Goals and Objectives	REQ-000022	2.1
Customer Categories (User Profiles or Personas)	REQ-000024	1.2

The structure refreshes showing the new parent-child relationship:


Business Model	REQ-000027	2
Goals and Objectives	REQ-000022	2.1
Value Proposition	REQ-000028	2.2
Market Segment	REQ-000029	2.3

## Create requirements directly in the Documentation tab


You can insert a new requirement directly in the **Documentation** tab editor.

**Note:**

Control characters are not supported on the Documentation tab for non-UTF8 environments. Setting the environment variable POM\_STRIP\_CTRL\_CHARS enables the application to remove the control characters and save the data to the database. This is only applicable for non-UTF8 environments. For more information, see Teamcenter Environment Variables.

1. **Locate the requirement content** that you want to edit, and then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. To the left of the requirement under which you want to create the new requirement, click **Add**  and select **Sibling** or **Child** if the options are available.

**Note:**

You can use the keyboard shortcuts Shift + Enter to add a child or Ctrl + Enter to add a sibling.

A new requirement entry appears and the paragraph numbers update automatically.

3. Edit the requirement title and content text.

**Note:**

Do not enter a requirement number in the title. The number is generated automatically.

4. Click **Save Edits** or use the keyboard shortcut Ctrl + S.

The updated requirement structure saves.

## Create or paste sibling or child requirements or paragraphs

### Create Siblings or Children


You can create and add sibling or child requirement specifications, requirements, and paragraphs to create a requirement specification structure. You can also **add requirements and paragraphs directly in the Documentation tab editor**.

Note:

The custom notes functionality is currently not available in Active Workspace.

1. **Locate the requirement specification** that you want to update, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Select the **Tree view**.
3. Click a requirement or paragraph in the table.
4. Click **Add** , and then select **Sibling** or **Child**.

Tip:

You can use the keyboard shortcuts Ctrl + Shift + Enter to add a child or Ctrl + Enter to add a sibling.


The new item appears.

### Copy and Paste Siblings or Children

You can copy and paste content to add sibling or child requirement specifications, requirements, and paragraphs.

1. **Locate the requirement specification** that you want to update, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Select the **Tree view** or **Table** view.
3. Select a requirement or paragraph, and then click **Copy** .
4. Select the target requirement or paragraph row.

**Note:**

If you do not select a row, the pasted requirement or paragraph appears at the bottom of the content panel table or tree.

5. Click **Paste**  > **Child** or **Sibling**.

## Create duplicate requirements from existing requirements

Use duplication to reuse requirements from one or more specifications in a different specification while maintaining independence. When using duplication, the selected requirements form the building blocks of the new specification but are not related to the old specification.

New IDs are generated for duplicated requirements. Therefore, any changes made to an original requirement do not affect its duplicate, and any changes to a duplicate do not affect the original.


### Restrictions and limitations

The commands **Move Up**, **Move Down**, **Promote/Demote/Paste as Child** or **Promote/Demote/Paste as Sibling** are not supported for multi-selection.

### Prerequisites

You must have a specification where you can insert the existing requirements before proceeding with the procedure.

### Procedure

1. Open the specification where the existing requirements will be inserted.
2. Click **Add** , and then click **Insert from Structure**.

The **Insert from Structure** panel appears.



3. Select the specification that contains the requirements you want to reuse, and click **Open in View**.

The screen opens in split view, showing the empty new requirement specification on the left and the original requirement structure on the right.

4. Copy the requirements in one of these ways:
  - To use drag-and-drop functionality:
    - a. Drag a requirement from the structure on the right and drop it in the structure on the left.

- b. Click **Paste as Copy** to create the duplicate requirement.

The **Paste** panel appears.

- To use copy-paste functionality:
  - a. Select the requirement you want to reuse in the structure on the right and click **Copy**  on the primary toolbar.
  - b. Select the specification in the structure on the left and click **Paste**  > **Child As Copy** on the primary toolbar.

The **Paste** panel appears.


5. Choose an option for the naming rule:

- Select **Default IDs** to use the default naming rule for the new requirement.
- Select **ID Naming Rule** to define a naming rule for the new requirement.

6. Click **Paste**.

The selected requirement structure appears in the new specification.

## Postrequisites



You can select the new requirement and then click **Information**  to view information about the original requirement.

## Move requirements up or down

As a system analyst with edit permissions, use **Move up** and **Move down** to modify the order of sibling requirements. If you want to move a requirement to a different parent, use the **Demote** feature to move the requirement to the same level as the target parent before using **Move up** and **Move down**.


1. **Locate the requirement specification (or other requirement)** that you want to update, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Select the **Tree view**.
3. Select the sibling requirements that you want to move up or move down.
4. Click **Move Up**  or **Move Down**  to move the selected requirement up or down, but within the same parent.

## Promote or demote requirements

You can use promotion and demotion to modify the requirement specification structure by changing the parent-sibling-child relationships. Use **Promote** and **Demote** in conjunction with the **Move up** and **Move down** features to adjust the structure.



1. **Locate the requirement specification (or other requirement)** that you want to update, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Select the **Tree view**.
3. Select the requirement that you want to promote or demote.

Note:

Demoting a requirement moves that require under the requirement directly above it.

4. Click **Promote**  or **Demote**  to move the selected requirement right or left in the structure.


Tip:

You can also drag multiple requirement components on to a parent component to create children quickly.

## Delete or move requirements with Remove, Copy, and Palette





You can update the requirement structure in the **Tree** view using the following commands:


- **Remove** command to remove a requirement from the structure.
- **Remove** command and the **Palette**, which operates similar to a cut and paste. This updates the occurrences, trace links, and all other data associated with the requirement specification.
- **Copy** command and the **Palette**, which operates similar to a copy and paste. This updates the occurrences, trace links, and all other data associated with the requirement specification.

1. **Locate the requirement specification (or other requirement)** that you want to update, and then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Select the **Tree view**.

3. Select the requirement that you want to move or press the Ctrl or Shift key to select multiple requirements.
4. Perform one of the following:
  - Click **Edit**  > **Remove** and confirm the removal. The selected element copies to the Palette. If you do not want to move the element, you are done.
  - Click **Copy** . The requirement copies to the Palette.
5. Select the requirement under which you want to add the requirements from the **Palette**.
6. Select the location in the table where you want to move the requirements to.
7. Do one of the following:
  - Click **Add**  > **Child, Sibling, or Replacement** to place the requirement a level below the selected requirement, at the same level as the selected requirement, or to replace the selected requirement respectively.
  - Click **Edit**  > **Replacement**.

Click  > **Child, Sibling or Replacement** to place the requirement a level below the selected requirement, at the same level as the selected requirement, or to replace the selected requirement respectively.

Active Workspace displays the **Add** panel.

The requirements that you removed are displayed in the **Clipboard** section.

8. From here, select the requirements you want to move to the selected location and click **Add**.

Active Workspace adds the requirements to the selected location.

## Assessing requirement quality

### Assessing requirement compliance

You can analyze the quality of a selected requirement and generate a quality matrix with a single command. This lets you check for quality and consistency in requirements, and aids in the creation of correct and complete requirements.

**Note:**

To use this feature, the preference **AWC\_REQ\_Reuse\_URL** must be implemented. Contact your system administrator for more information.

1. **Locate the requirement specification** that you want to view, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. From the **Documentation** menu, click **Assess Quality Compliance > Assess Quality Compliance**.
3. Select the requirement to evaluate.
4. At the bottom of the page, select the **Pattern Group** and **Pattern**. A *pattern* is a group of a certain set of quality rules.
5. Click any of the choices to view the results: **Correctness**, **Pattern Examples**, **Matching Pattern Structure**, and **Matching Pattern State**.
6. Make any corrections to the requirement, and the quality report is updated.

More information can be found on **The Reuse Company** website.

## Using patterns to create requirements


You can use predefined patterns to create requirements, ensuring quality and consistency. Automated quality rules assist in guided requirement creation, automatically mapping the requirement to the parameter in the creation process and ensuring correct formatting and correlation with the language, measurements, and other engineering components used by your organization.

**Note:**

- The third-party application from The Reuse Company must be installed.
- To use this feature, the preference **AWC\_REQ\_Reuse\_URL** must be implemented. Contact your system administrator for more information.

1. **Locate the requirement specification** that you want to edit, then click **Open** .

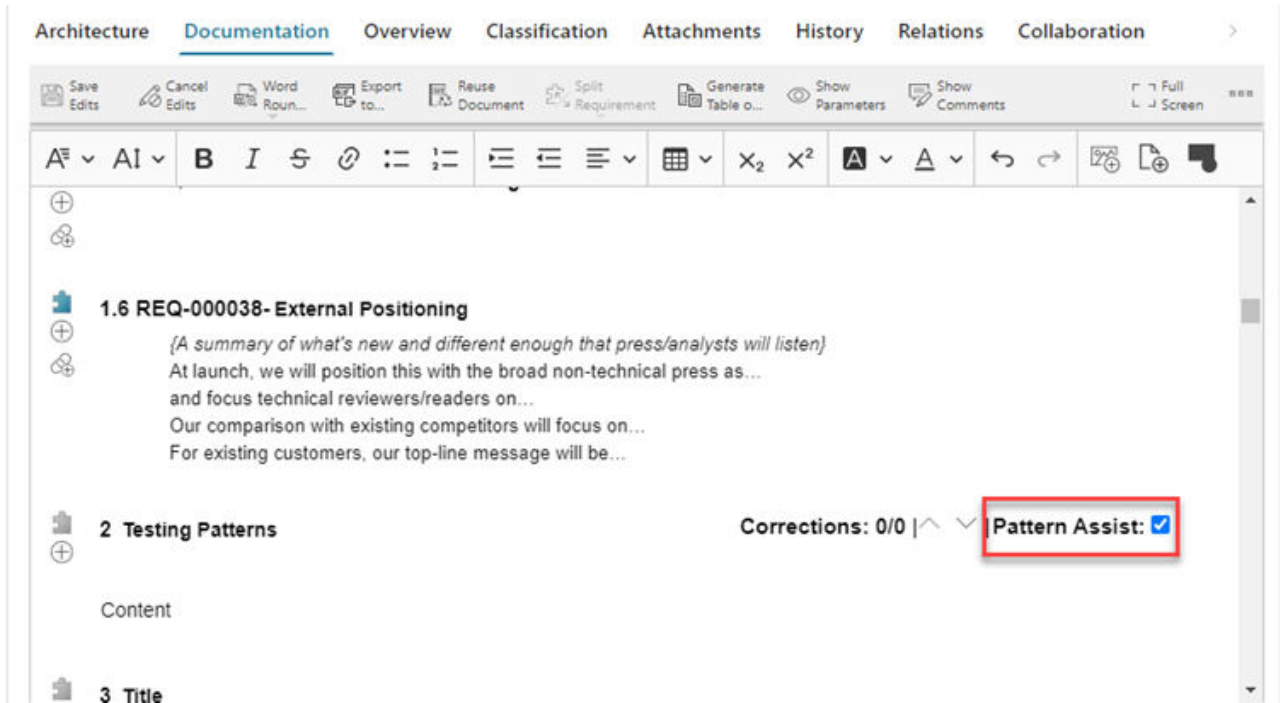
The requirement content appears in the **Documentation** tab editor.

2. Click **Add** , and then select **Sibling** or **Child**, to the requirement you want to work with by clicking

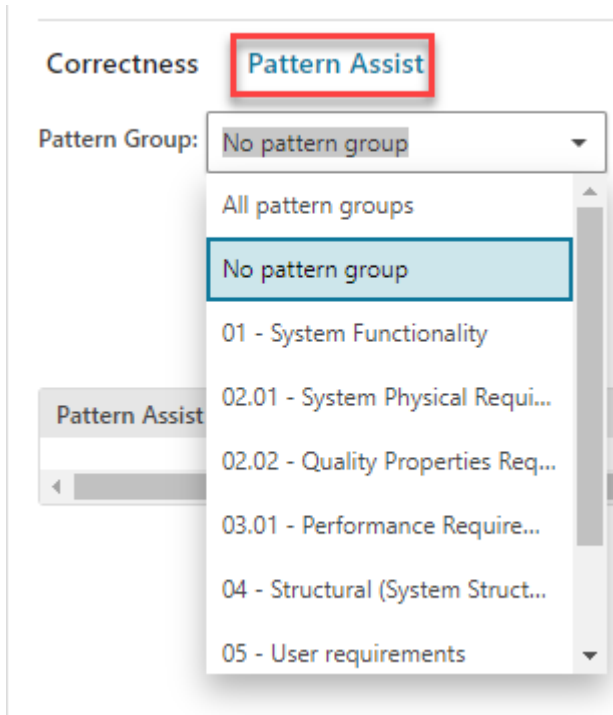
**Note:**

You can use the keyboard shortcuts Ctrl + Shift + Enter to add a child or Ctrl + Enter to add a sibling.

3. Click **Assess Quality Compliance** > **Assess Quality Compliance** from the **Documentation** menu.
4. In the new requirement, select **Pattern Assist** from the requirement header.



5. Click **Pattern Assist** in the panel under the requirement. From the **Pattern Group** at the bottom of the page, select the pattern group to use.



6. From the **Pattern Group**, select the **Pattern** to use. You can select from the list, or type ahead, and words or phrases are filtered.
7. Once you complete the selection process, click **Correctness** to assess the correctness of the phrase.

Metric	Correctness	Value	Summary	Mandatory
R02 Precision - TRC - Imperative mode (Enforce)	★ ☆ ☆	0	At least one imperative verb must be involved.	False
Wrong unit for a given property	★ ★ ★	0		False

8. (Optional) Make any edits to the phrase, and click **Correctness** to reevaluate it.

## Download a quality report

After you assess the quality of the requirement, you can use a single command to generate a downloadable report.

Note:

To use this feature, the preference **AWC\_REQ\_Reuse\_URL** must be implemented:

- **Preference description:**

Stores the URL to the Reuse Web API, which can be used to allow Requirement Quality Compliance from Active Workspace. The value is in the format `<Protocol>://<Domain>:<Port>/<Site> .`

- **Context name:** Teamcenter

- **Value:** `https://authoring.reusecompany.com:9095/v18.4_RC_Siemens`

Contact your system administrator for more information.

1. **Locate the requirement specification** that you want to view, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. After you **assess the requirement's quality**, click  **Assess Compliance Quality**, and then select **Download Quality Report**.

The report is generated and available for download.

More information can be found on **The Reuse Company** website.

## Associating requirements with system model blocks or workspace objects using trace links

### Manage trace links in the Documentation tab editor

The topics in this section provide information about working with trace links specifically in the **Documentation** tab editor.

For general information about working with trace links, see the Active Workspace Fundamentals documentation.

This procedure details trace link creation with requirements, but the procedure applies to the other supported objects such as system model blocks.

Note:

You can create a one-to-many or a many-to-one trace link relationship, but you cannot create a many-to-many trace link.

1. **Locate and open the requirement content (requirement specifications, requirements, or paragraphs).**

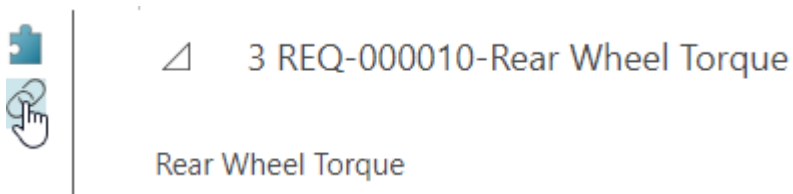
The requirement content appears in the **Documentation** tab editor.

2. In the **Documentation** tab editor, click **Trace Link**  to the left of the requirement component.



Note:

You can use the keyboard shortcut Alt + L to create a trace link.


The **Create Trace Link** panel appears.



Consider the following:

- If you want to create multiple trace links, click **Pin**  in the **Trace Link** panel. Pinning keeps the **Trace Link Panel** open after you create a trace link. You can then create another trace link by changing the **Start** or **End** objects.
- To remove requirements from the trace link, select the requirements in either the **Start** or **End** panel, and then click **Remove** .
- Click the **Existing** tab to show trace links already created.
- Click **Search** to search for workspace objects and select the **Search in-context only** check box to search within the selected configuration.

3. Select the **Type**, and then click **Create**.

The trace link icon turns bold  to indicate that a trace link is active. This icon also shows trace links to Polarion ALM objects.

## Create a trace link across browser tabs

You can create trace links between requirements and other workspace objects, across browsers.


This procedure details trace link creation with requirements, but the procedure applies to the other supported objects such as system model blocks.

Note:

You can create a one-to-many or a many-to-one trace link relationship, but you cannot create a many-to-many trace link.

1. **Locate the requirement content (requirement specifications, requirements, or paragraphs).**


The requirement content appears in the **Documentation** tab editor.

2. Select one or more requirements.
3. Click **Trace Link**  or use the keyboard shortcut Alt + L.


The **Create Trace Link** panel appears.

4. Drag the selected requirements to either the **Start** or **End** area on the **Trace Link** panel.

Tip:

To remove requirements from the trace link, select the requirements in either the **Start** or **End** panel, and then click **Remove** .

5. Using the **Search** function in Active Workspace, search for another object. The **Trace Link** panel remains open.
6. Drag and drop, or copy and paste, the object into the **End** area on the **Trace Link** panel.
7. Select the **Type**, and then click **Create**.


In the **Documentation** tab, the trace link icon turns bold  to indicate that a trace link is active.

### Trace link properties

You can easily view the properties for any trace link, including its name, direction (start or end), and owner.

1. **Locate the requirement content (requirement specifications, requirements, or paragraphs).**

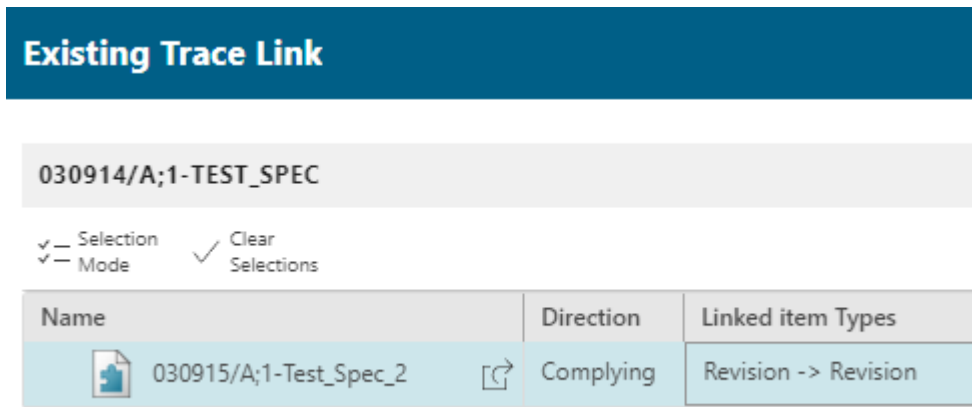
The requirement content appears in the **Documentation** tab editor.

2. Select the trace link whose properties you want to view by right-clicking **Trace Link** .

A list of the existing trace links appears.


3. Click **More**.

The **Existing Trace Link** panel appears, and displays trace link properties in a table format.



## Manage trace links with revisions and occurrences

An *occurrence* is the repeated use of a requirement using different occurrence names. For example, you could have a requirement for "Temperature-High" and then create copies that are occurrences that represent "Temperature-Medium" and "Temperature-Low." You can **edit the occurrence name in the Information panel**. You can create trace links in any relationship between revisions and occurrences, such as occurrence-to-revision, occurrence-to-occurrence, and so on.

1. **Locate the requirement content (revision or occurrence)** that you want to link, and then click **Open** .
2. **Create the trace link using any method.**

Revisions and occurrences are indicated with an **R** and **O** icon, respectively in the **Trace Link** panel. Click the icon to switch between revision and occurrence. When switching to the occurrence, the occurrence name displays.


3. Click **Create**.

## Remove a requirement from a trace link in the Documentation tab

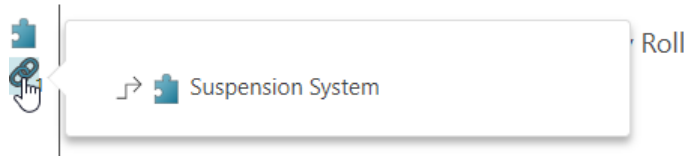
As a system analyst, you can remove a trace link from a requirement by removing the requirement as the defining or complying object. This procedure details trace link deletion with requirements, but the procedure applies to the other supported objects such as system model blocks.


1. **Locate the requirement specification, requirement, or paragraph** that you want to view, and then click **Open** .

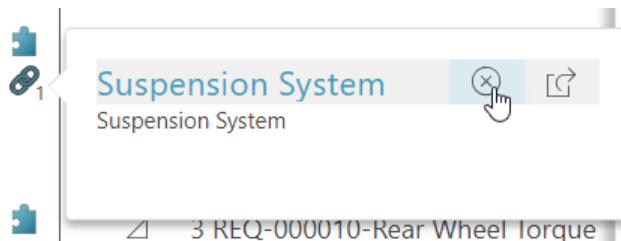
The requirement content appears in the **Documentation** tab editor.

2. Hover over a trace link icon .

A pop up displays the linked requirements. The following graphic shows the user hovering over an active trace links to show that it is linked to **Requirement\_A**.



3. Click the linked object in the pop up.
4. Click the linked object in the pop up and then click **Remove** .



The selected object is removed from the trace link.

### Working with suspect links

#### Caution:

Beginning with Active Workspace 2412, the async translator must be installed and the dispatcher environment must be running to work with suspect trace links. The **REQ\_suspicion\_mgmt** preference is deprecated.

#### Note:

- Occurrence trace links are not supported. Revision-to-revision trace links are supported.
- The **Tracelink\_Edit\_enabled** preference must be enabled for each user to ensure delivery of suspect tracelink notifications.

Trace links create relationships between dynamic objects that can include requirements, system models, and so on. These objects change throughout the product lifecycle. Object owners impacted require change notification. Active Workspace notifies impacted users when a change to a trace-linked object, such as a requirement, occurs. The impacted user can then choose to comment with approval or rejection. For example, a requirement has a trace link to a system model and the requirement owner

changes the requirement. Active Workspace sends a notification to the system model owner receives a notification that the requirement has changed.

The following process details a typical scenario where **defining\_object**, in this case, a requirement, has a downstream trace link to **complying\_object**, in this case, a system model. Each object has a unique user owner.

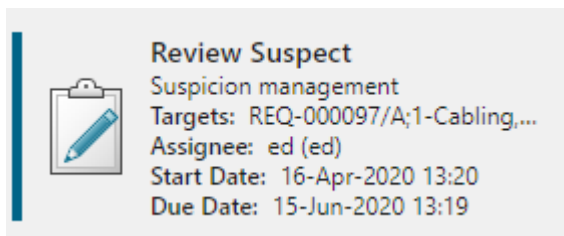
Consider the following:

- The suspect link indicator is available for any Active Workspace object.
- If the owner of the **defining\_object** and **complying\_object** have the same owner, Active Workspace does not send a suspect trace link notification.
- Due to workflow restrictions, trace link notifications are sent for checked-in objects only. Additionally, objects cannot be released or have a denied access state.

1. Open an object and click **More commands ... > Manage**  **> Mark as Suspect** .

Because the **defining\_object** has a trace link to a **complying\_object**, Active Workspace starts a signoff workflow where the object owner receives a notification in their **Inbox** that the object has changed.

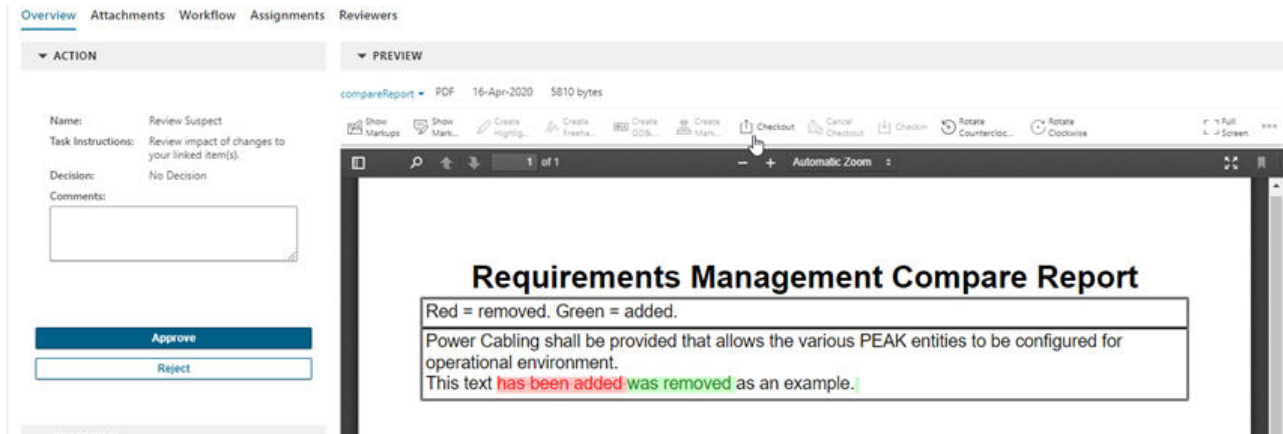
An Inbox notification looks similar to the following graphic:



2. The object owner opens the **Review Suspect** notification to view the requirement change report.

Note:

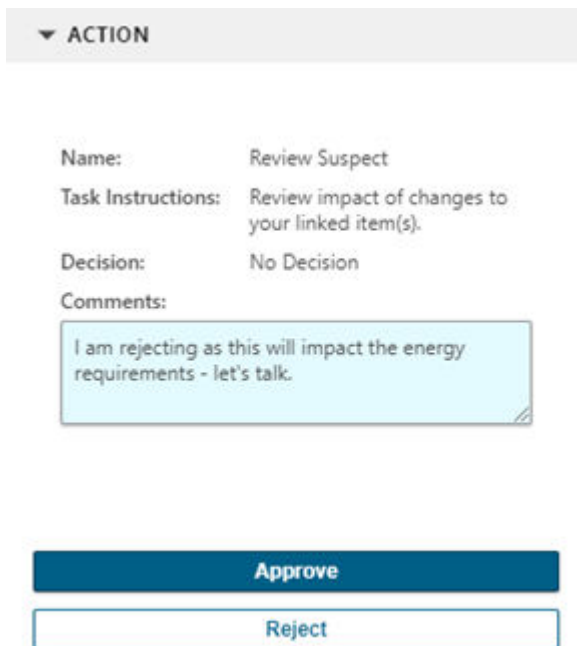
The change report is not available for content authored in Microsoft Word.



3. Perform the following:

- Add comments about the requirement change.
- Click **Approve** or **Reject** for the requirement change.

The following graphic shows the system model owner rejecting the change and commenting on the reason:



4. Either owner can click the **Workflow** tab to see the suspect signoff status.

Overview Attachments **Workflow** Assignments Reviewers

**▼ CURRENT AND COMPLETED TASKS**

Selection Mode Select All Export To... Paste ...

Task	Status	Performer	Due Date	End Date	Comments
New Review Task 1	Started	ed (ed)	15-Jun-2020 13:30		
New Review Task 1 : Signoff	No Decision	ed (ed)	15-Jun-2020 13:30		I am rejecting as this will ...
New Review Task 1 : Select Team	Completed	ed (ed)		16-Apr-2020	
Review Suspect	Completed	ed (ed)	15-Jun-2020 13:30	16-Apr-2020	

**► UPCOMING TASKS**

Apply Layout Full Screen

```

graph LR
    Start[Start] --> NRT1[New Review Task 1]
    NRT1 --> RS[Review Suspect]
    RS -- Reject --> NRT1
    RS -- Approve --> Finish[Finish]
  
```

#### Note:

The user who initiates the workflow does not receive workflow error notifications. Errors are indicated in the log files. For more information, contact your system administrator.

## Review a suspect link from the documentation tab

When a suspect is raised, the owner of the suspect trace link sees an indicator in the **Documentation** tab.

1. Open a requirement for review. Click **Documentation**.

If a suspect object exists, this icon appears:



2. Click to opens the **Inbox** to review notifications.
3. Proceed from **step 3** in the previous section.



## Set verification method between requirement and EBOM occurrence with a trace link

You can update a trace link between a requirement and an EBOM occurrence by updating the trace link properties between the items.

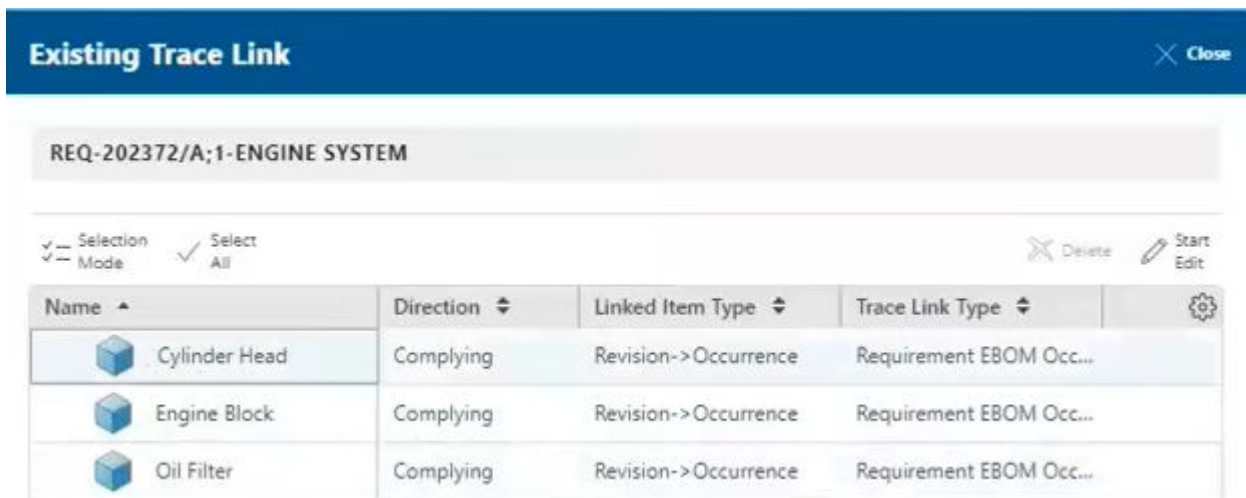
### Prerequisites

There must be an existing trace link between a requirement and EBOM occurrence.

### Procedure


1. Open a requirement specification and select a requirement in the tree.
2. Click **More commands** **...** > **View**  > **Existing Trace Link** .

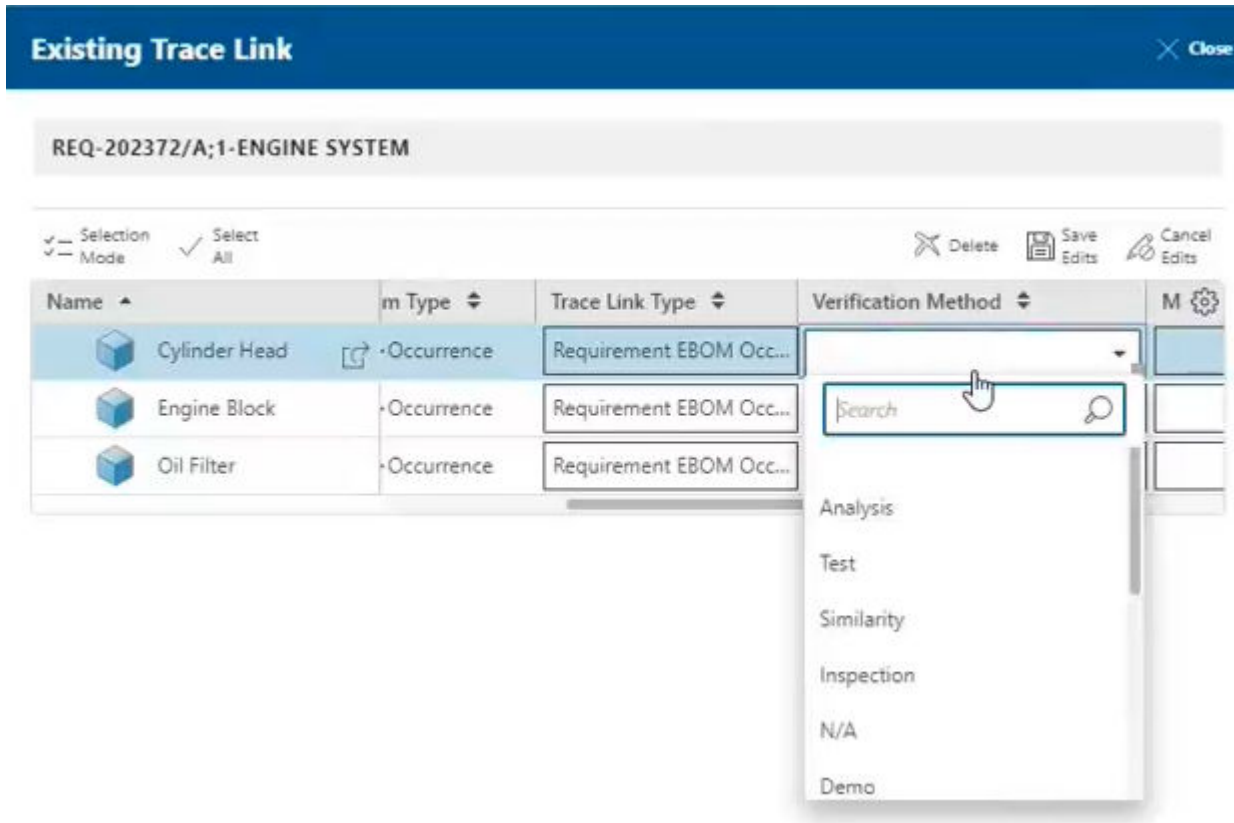
The **Existing Trace Link** dialog appears, showing the link properties. The **Verification Method** column does not appear in the properties table by default, so you need to add it.




#### Note:

When the **Existing Trace Link** panel is launched inside the matrix, no Occurrence to Occurrence trace link is displayed

3. Click **Arrange**  and add the **Verification Method** and **Method Description** columns to the trace link properties table.
4. Click **Arrange** to close the dialog box and update the trace link properties table.
5. Click the **Verification Method** field for a requirement, and then select a method from the dropdown.



- (Optional) Click the **Method Description** field and enter information about the verification method.
- Click **Save Edits** , and then close the dialog box.

## Results

Active Workspace updates the trace link properties with the verification method properties.

## Manage trace links with the Tracelink Matrix



### Generate a trace link matrix

The trace link matrix allows you to view and manage the relationships of trace-linked items within a structure. These items may include requirements, specifications, parameters, parameter sets, or parameter projects.


### Display options

When including requirements and specifications in a trace link matrix, you can choose whether to display all parameters associated with these items in the trace link matrix. Parameters are automatically displayed when parameter sets and parameter projects are included in a trace link matrix. See [Perform operations from the trace link matrix](#) for information on hiding parameters in the matrix.

If two different items are selected, you can both swap the rows and replace the column object prior to generating the matrix.

If only one item is selected, the **Matrix is symmetric** check box appears. This check box is selected by default and the **Swap**  and **Replace Column**  icons are not available. You can clear this check box and then replace the column prior to generating the matrix.

Note:

The **Swap**  option is always unavailable when only one item is selected for the matrix.

### Matrix size adjustment

In the **List** view, select a number from the **Items per page** drop-down, such as **50** to adjust the matrix to 50x50.

### Trace link count

Each cell provides a count of trace links for the row and column objects. Additionally, the summation ( $\Sigma$ ) column and row cells provide the trace link count for all trace links in the respective row or column. The calculation of the trace link summation depends on the matrix type.

### Matrix types

There are three matrix types.

- **Quick Matrix:** Shows the direct tracelinks between row and column objects.
- **Full-Rollup Matrix:** Shows tracelinks that include the children of row and column objects.
- **Dynamic Matrix:** Shows all tracelinks to which an object is connected.
- The available matrix types depend on the items selected to use as rows and columns. This table shows the matrix type availability.

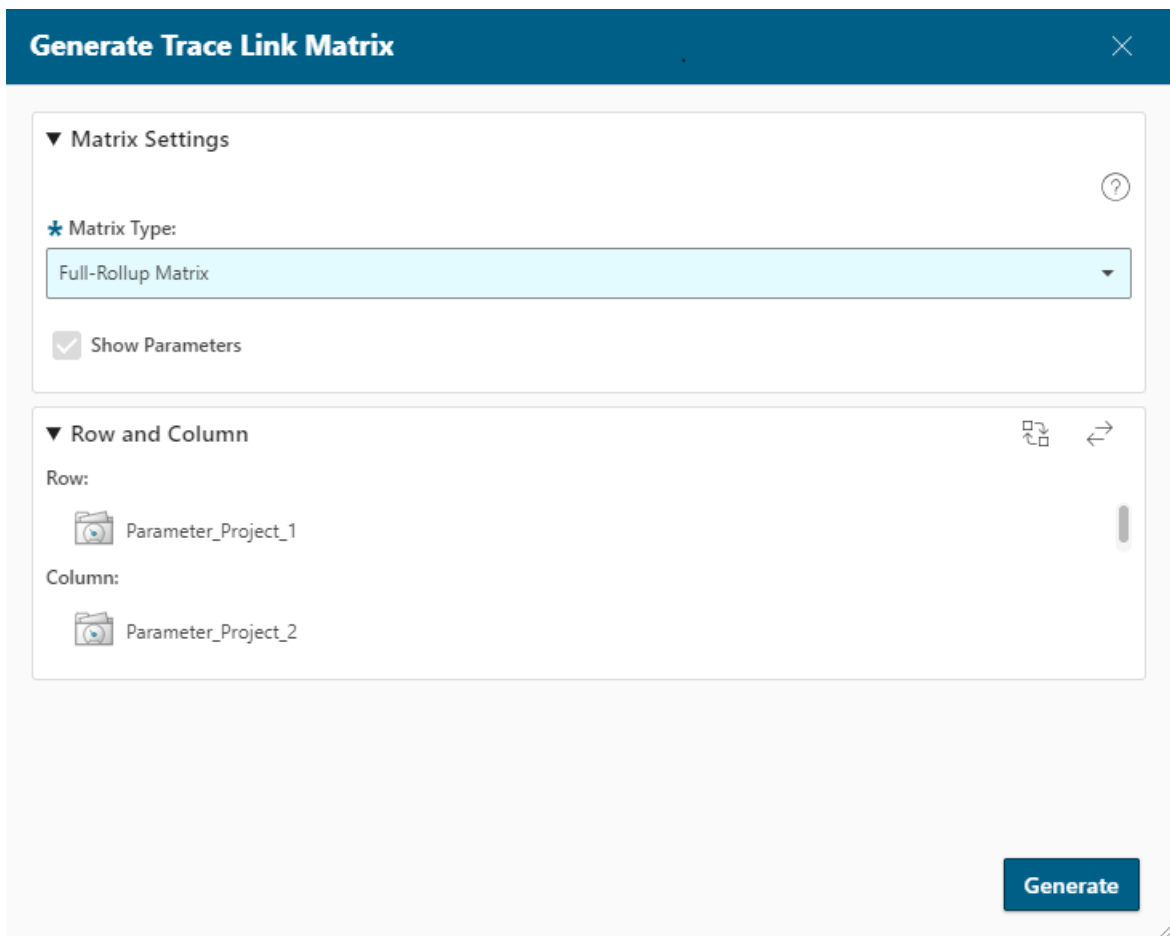
If you select	Quick Matrix	Full-Rollup	Dynamic Matrix
Single item	Yes	Yes	Yes
Two items	Yes	Yes	Yes
Single specification	Yes	Yes	Yes
Two specifications	Yes	Yes	Yes
Single folder	Yes	No	Yes
Two folders	Yes	No	Yes
More than two folders	No	No	Yes

If you select	Quick Matrix	Full-Rollup	Dynamic Matrix
One folder and one item revision	Yes	Yes	Yes
Single parameter project	Yes	Yes	Yes
Two parameter projects	Yes	Yes	Yes


## Procedure

1. **Select requirement specifications, items, or other structures** to use as the rows and columns of the matrix. If only one object is selected, the same structure is used for both rows and columns. For a dynamic matrix, you can select any number of objects.
2. Click **More commands...**, and then click **New > Generate Trace Link Matrix**.


The **Generate Trace Link Matrix** panel appears.



3. Select a **Matrix Type**.

4. (Optional) Clear the **Matrix is symmetric** check box to activate the **Replace Column** icon if only one item is selected for the matrix.
5. (Optional) If including requirements or specifications in the matrix, clear the **Show Parameters** check box to hide all associated parameters.
6. Do you want to manage the row or column settings?
  - To swap the row and column, click **Swap** .

The objects change places in the **Row and Column** box.

- To replace the column object, do the following:
  - a. Click **Replace** .
  - The **Replace Column** fields appear.
  - b. Enter a search term.
  - c. Select an object to replace the column.
  - d. Click **Replace**.

Note:

These options are available only for Quick and Full-Rollup Matrix types.

7. Click **Generate**.

## Results

The matrix appears in the primary work area. The trace link count for all collapsed rows appears in the summation column. You can expand the rows to expose the child trace link counts.

Note:

When the **Existing Trace Link** panel is launched inside the matrix, no Occurrence to Occurrence trace link is displayed.

The following graphic shows a sample **Quick Matrix** with parameters displayed.

Quick Matrix

Filter Expand View Hide/Unhide Reset Replace Save Refresh Trace... Settings Create Trac...

		Test_Spec_1a							
		Test_Spec_1a	No Title	param2	param1	Description*	Rationale*	Analysis	
	Σ	1	2	2	1		1	1	
Test_Spec_1a	Test_Spec_1a	1		1					
	No Title	2				1	1		
	param2	2	1		1				
	param1	1		1					
	Description*								
	Rationale*	1	1						

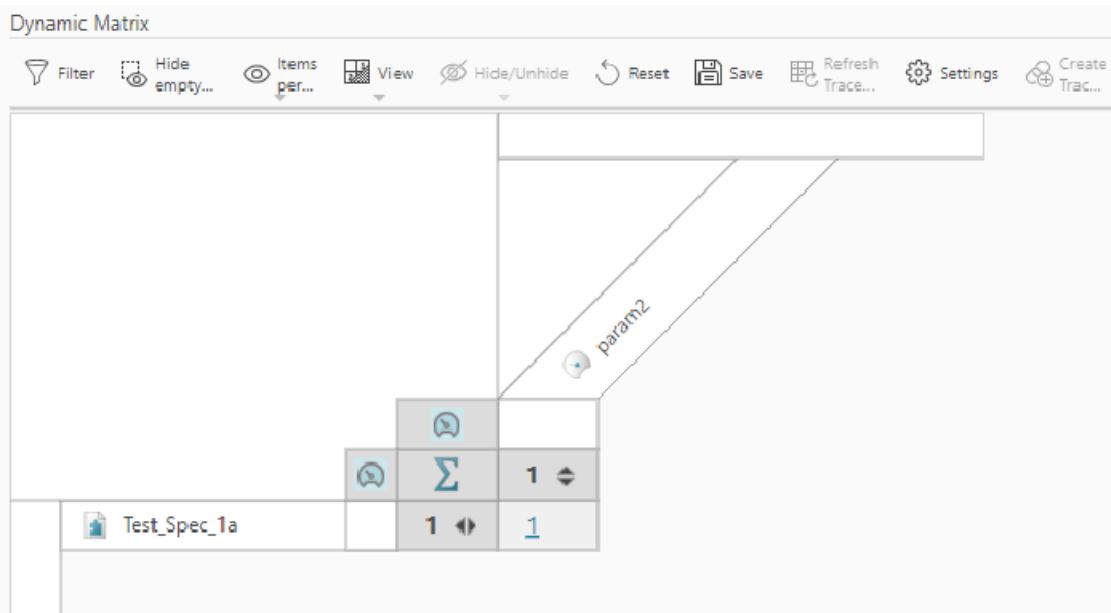
The following graphic shows a sample Full- Rollup Matrix with parameters displayed.

Type: Traceability Matrix Owner: ed (ed)

Filter Expand View Hide/Unhide Reset Save Refresh Trace... Settings Create Trac...

		Test_Spec_1a							
		Test_Spec_1a	No Title	param2	param1	Description*	Rationale*	Analysis	
	Σ		4				1	1	
Test_Spec_1a	Test_Spec_1a								
	No Title	4	2			1	1		
	param2								
	param1								
	Description*								
	Rationale*	1	1						

The following graphic shows a sample **Dynamic Matrix** with parameters displayed.




### Perform operations from the trace link matrix


You can perform several operations from the trace link matrix.

#### Create a trace link

You can create a trace link directly in the matrix.


1. Do one of the following:
  - Select a cell in the matrix and click the **Create Trace Link**  button in the toolbar or use the keyboard shortcut Alt + L.

Note:

The **Create Trace Link**  button becomes visible in the toolbar when a cell is selected.

- Double-click a cell in the matrix.

The **Create Trace Link** panel appears with the matrix cell row as the **START** and the cell column as the **END** targets.

2. (Optional) Click **Swap**  if you want to switch the start and end objects for the trace link.
3. (Optional) Select a trace link **Type** from the drop-down list.

4. Click **Create**.

The matrix is updated to include the new trace link in the cell count.

**Note:**

The trace link matrix is a static snapshot of the available trace links between row and column objects. Once it is generated, you can navigate to another link in the matrix, and click the back button to return to the matrix page. In this case, however, any trace links created by you or a different user *after* the matrix is generated, do not appear when you click the back button.

### Create a one-to-many tracelink

You can create trace links quickly between requirements, parameters, and other workspace objects, including adding one-to-many tracelinks. You can select multiple cells in a row or column, and choose to create a one-to-many or many-to-one tracelink.

1. Click **Ctrl** and select several cells in a row or column. You must select cells in the same row or column.

You cannot select cells in a different row or column.

2. Click **Create Trace Link** .

The **Create Trace Link** panel appears, showing the starting and ending objects.

3. Click **Create**.

The matrix is updated to show the new trace links in the selected cells.

### View trace link details and delete trace links

You can view information about the trace links and delete one or more trace links directly from the matrix.

1. Click a cell with a trace numerical value.

A table appears at the bottom of the screen showing trace links associated with the selected cell. The table provides information about each trace link such as type, direction, and defining or complying (start and end) object names.

2. Select one or more trace links, and then click **Delete**.

The matrix is updated with the new trace link values.

## View trace link density with the heatmap

A heatmap shows the number of trace links through a color indicator. The darker the color, the higher the number of trace links.

To activate, click **View**  > **Heatmap** .

## Update the trace link matrix settings

You can update the settings for trace link matrix viewing.

1. Click **Settings** .

The **Matrix Settings** panel appears.

2. For row and column heading display names, do one of the following:
  - Select the **Name** check box to display the object name.
  - Select the **ID** check box to display the ID number.
  - Select the **Owner** check box to display the object owner's name.
  - Select all check boxes to display the all information.
3. Select the **Show Parameters** check box to hide parameters in the matrix.
4. Select **Count** or **Heatmap** in the **Grid View** field.
5. Select **Show arrows** to display the trace link direction arrows.
6. Click **Apply**.

## Hide or unhide parameters by row or column

Click the **(icon)** to the left of a row or the top of a column to hide or unhide any parameters in that area.

## Hide or unhide rows or columns

Note:

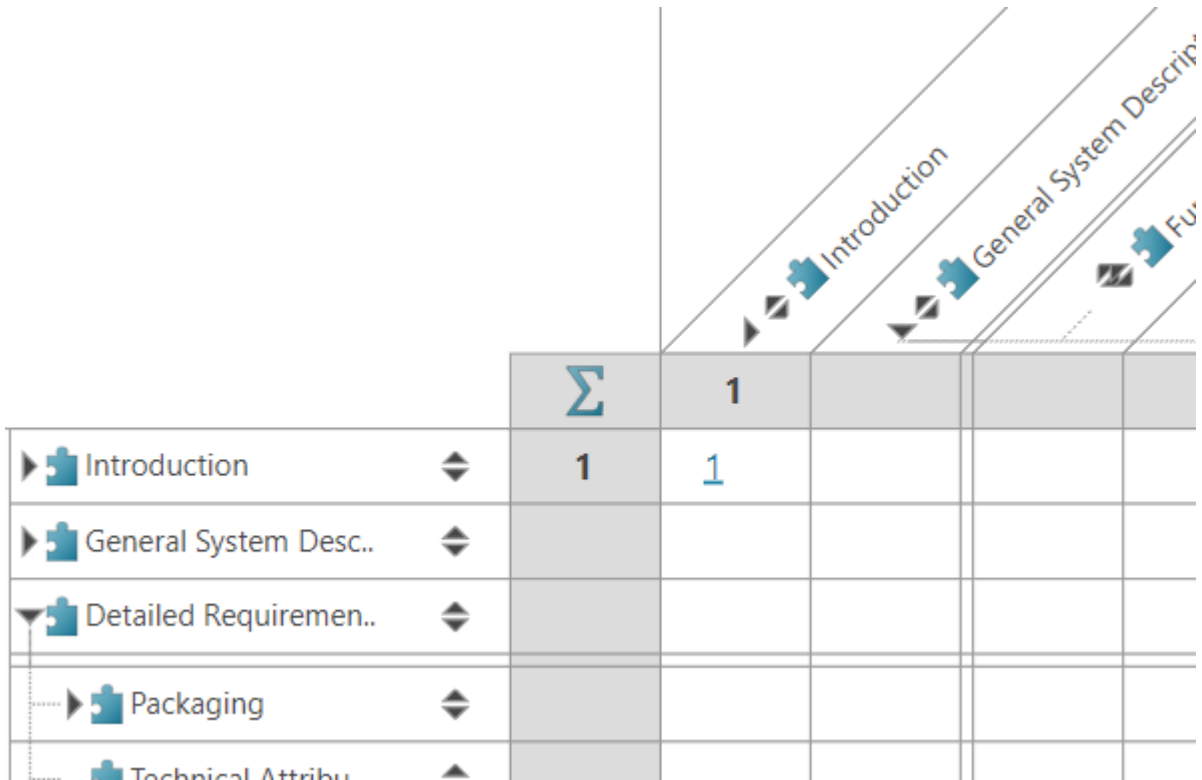
This feature is available in list view only.

1. Select one or more rows or columns.


- Click **Hide/Unhide** .

The selected rows or columns are hidden and are indicated by double lines.

The following graphic shows that rows below **Detailed Requirements** and that columns to the right of **General/ System Description** are hidden.



	Σ	1			
Introduction	1	1			
General System Desc..					
Detailed Requiremen..					
Packaging					
Technical Attribu					

- (Optional) To unhide rows or columns, select a double line and then select **Hide/Unhide**  **> Unhide**. You can also select **Unhide All** to unhide all hidden rows and columns at once.

Drill down, filter, and sort rows or columns in the trace link matrix

### Drill down the hierarchy with headings

Choose an option to expand or collapse a structure hierarchy to view or hide child trace links:

- Double-click a matrix column or row heading.
- Click the arrow next to the row or column label.

Click arrow to expand or collapse rows or columns

PEAK System Specification						
		Σ	4			
PEAK System Specific...	Introduction	1	1			
	Document Purpose	1	1			
	Specific System P..	1	1			
	Definitions, Acro..	1	1			
	Customer and Stan..					
	General System Desc..					
	Detailed Requir...					

- Select a row or column, and then click **Expand** and choose to **Expand Below** the selected row or column, **Collapse Below** the selected row or column, or enter a number that represents the number of levels to expand.


### Sort rows and columns by headings

To sort by rows and columns by the heading name, click the up-down arrow next to the row or column specification name to sort by ascending or descending order. You can also remove the applied sorting.

Sort rows and columns with arrows

		Σ	4	6	4	1
Vehicle R...	Complete Vehicle Sy..	8	1	2		1
	Body System	4		2	2	
	Frame And Mounting ..	3		2	1	
	Engine System	1			1	
	Suspension System					
	Driveline System					

## Filter trace link by properties


- To filter the trace links, Click **Filter** .
- The **Filter** panel appears.
- Select a **Trace Link Type**.
- Click **Filter**.

### Note:

The **Filter** operation requires that the objects are indexed using Smart Discovery indexing. See *Administration of Smart Discovery for Structures* for information about Smart Discovery indexing.

## Save or a load a tracelink matrix

### Save a trace link matrix


- Open or generate a trace link matrix.
- Click **Save** .
- The **Save** panel appears.
- Enter a filename and then click **Save**.

Active Workspace saves the matrix to your **Newstuff** folder.

**Note:**

- If a trace link matrix is modified after it is saved, you are prompted to refresh the trace link matrix upon reopening.
- The trace link matrix can be saved multiple times, but the trace link object is updated with the last saved content and name.

**Load a trace link matrix**

1. Navigate to your **Newstuff** folder.
2. Locate the trace link matrix file and then click **Open** .
3. Click **View** and then select **Count** or **Heatmap**.

The trace link matrix appears.

**Working with Parameters and Requirements**

**Parameters** let you define and manage variables, characteristics, measurements, calibration, configuration, and requirements. They are maintained in a single source dictionary shared across domains, applications, and life cycle. You can make a visual connection between the text in your requirement and a specific parameter (also known as measurable attribute) by marking text in a requirement to map to a new or existing parameter.

**View existing parameters**

1. **Locate the requirement specification** that you want to view, then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Click **Show Parameters** .

The parameters table is visible under the documentation view, and contains options to **Add**, **Save**, and **Edit** parameters.

PARAMETERS											
Name	Description	Direction	Source	Units	Meas...	Goal	Min	Max	In Co...	Value File	Goal File
Radius Req_Text			Req_Text	m		10			False		Radius Req_Text

## Add a parameter

1. With the parameters showing at the bottom of the screen, click **Add** ⊕.

You can also select text in the document, right click, and select **Add Parameters**.

The **Add Parameters** dialog opens.

The screenshot shows the 'Add Parameter' dialog box. At the top, there is a title bar with the text 'Add Parameter' and two icons: 'Pin Panel' and 'Close'. Below the title bar, there are three tabs: 'New', 'Palette', and 'Search'. The 'New' tab is selected. The main area of the dialog contains several input fields:

- Name:** A text input field with a red asterisk indicating it is required. The text 'Required' is entered.
- Description:** A text input field that is currently empty.
- Data type:** A dropdown menu with 'Double' selected.
- Unit of Measure:** A dropdown menu with '%' selected.
- Goal:** A text input field that is currently empty.
- Min:** A text input field that is currently empty.
- Max:** A text input field that is currently empty.

2. Add the attributes for the parameter and click **Add**.


The parameter is added to the table.

PARAMETERS							 
NAME	MIN	OPERATOR MIN	GOAL	OPERATOR MAX	MAX	UNITS	
Cruise Control Max Speed	30	<	70	<	102	Mph	
Longitudinal Separation	5	<	20	<	25	m	
Speed of Target and Subject	25	<	35	<	75	Mph	
Yaw Rate for Target	5	<	5	<	15	Deg/S	
Brake Torque	20	<	23	<	25	ft. lbs.	

### Edit a parameter

1. At the top of the parameters table, click **Edit**. .

Clicking **Edit** causes cells to become active. Make any changes to the properties in the table.

2. Click **Save** .

**Note:**

You can also select the parameter in a document and update it, and the parameters table updates accordingly.

Parameters associated with a document appear as a link in the document. Click the link to open the associated parameters table.

## View test results in the requirement tree view

You can view the rollup test result also known as the **Test Result** property, of a requirement or requirement specification in the **Tree** view.

For more information about verification or test request and test results, see *Managing and monitoring test packages*.


### Prerequisites

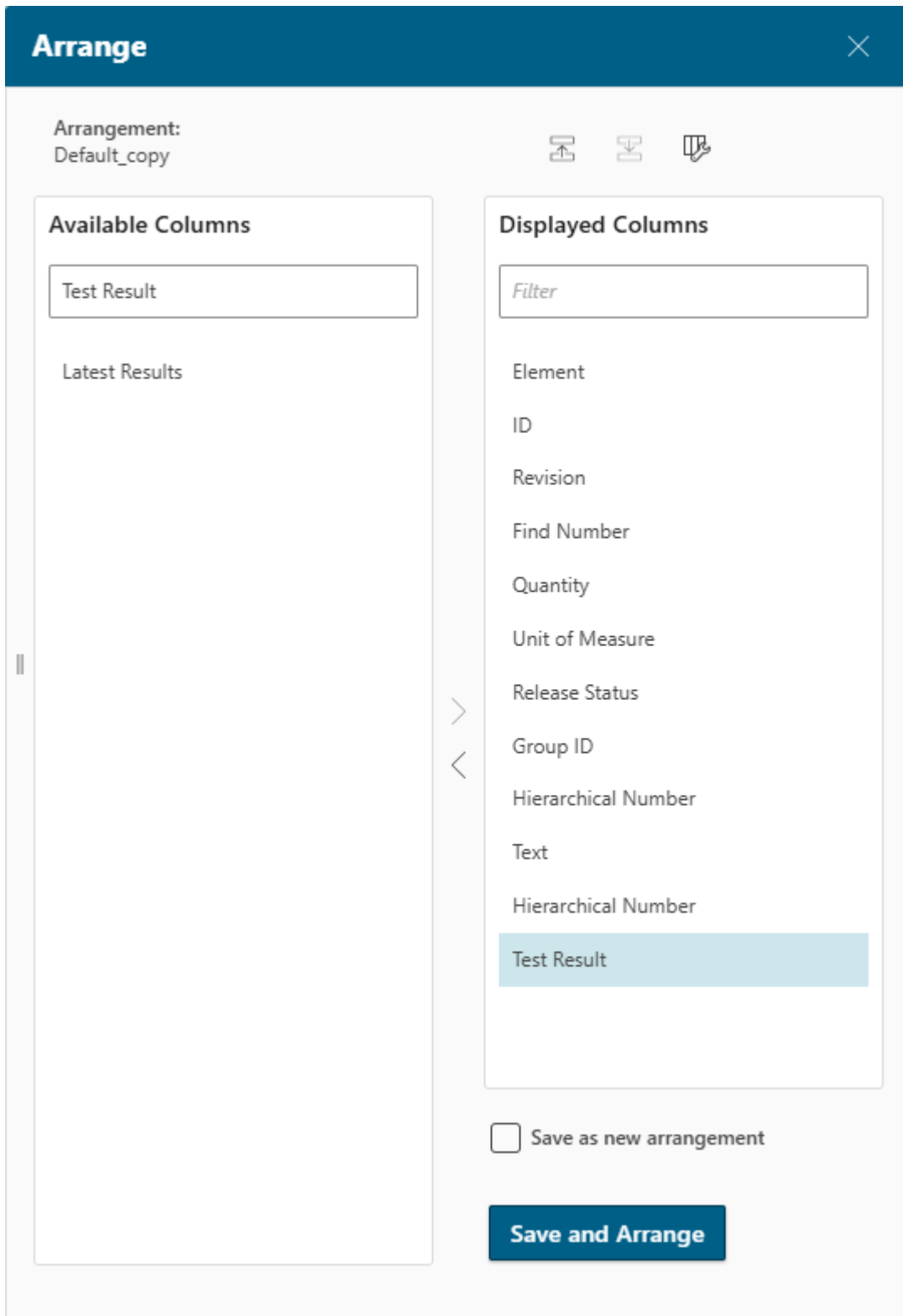
Install Verification Management and set the **PLE\_Rollup\_RequirementManagement** preference to **Overall** or **Latest**. The default value is **Overall**.



### Procedure

1. **Locate the requirement specification or requirement** that you want to view, then click **Open** .
2. In the **Tree** view, click **Table Settings**  > **Arrange**.

The **Arrange** panel displays.

3. In the **Available Columns** section, enter **Test Result** in the **Filter** field to search the column in the **Available Columns** list.
4. Select **Test Result** and click **Add** .



- (Optional) In **Displayed Columns**, select the property that you want to reorder and click **Move Up**  or **Move Down** .

6. Click **Save and Arrange**.

## Results

The **Test Result** column appears, displaying the rollup results for the respective requirements.


The following image shows a sample requirement specification structure in the **Tree** view.

Element	ID	Revision	Test Result	Fin...	Unit of ...
Specification		A			
Requirements	REQ-	A		10	each
1.1 Basic Control Strategy	REQ-	A			each
1.2 Adjust Subject Vehicle Speed	REQ-	A			each
1.3 Brake within Set Distance	REQ-	A			each
1.4 Minimum distance for emer...	REQ-	A	Pass		each
1.5 Maintain Following Distance	REQ-	A	Pass		each
1.6 Maintain Set Speed	REQ-	A		10	each
1.7 Maintain Set Speed	REQ-	A		20	each
1.8 Maintain Set Speed	REQ-	A	Fail	30	each
1.9 Maintain Set Speed	REQ-	A	Pass	40	each
1.10 Maintain Set Speed	REQ-	A		50	each

## Assign a reviewer to a requirement for approval

To approve a requirement specification or requirement, you can assign a reviewer or multiple reviewers.

### Procedure

1. **Locate the requirement specification or requirement** that you want to view, and then click **Open** .
2. In the requirement specification or requirement **Tree** view, do one of the following:
  - Right-click the requirement specification in the structure and click **Assign Reviewers**.
  - Click **More commands ... > Assign Reviewers**.

The **Assign Reviewers** panel appears.

- In the **Available Users** section, search for a reviewer and click **Add Select Users** ⊕.

Note:  
You can add multiple reviewers for a requirement.

- Click **Assign**.
- In the **Tree** view, click **Table Settings** ⚙️ > **Arrange**.

The **Arrange** panel appears.

- Click **Column Arrangements** 📄 and select **RM Approvals**.
- Click **Arrange**.

## Results

The requirement **Tree** view is displayed with columns like **Review status**, **Initiator**, and **Reviewers**.

The following image shows a sample requirement specification structure **Tree** view.

Element	Release Status	Initiator	My Signoffs	Reviewers
Specification				
Requirements				
1.1 Basic Control Strategy				
1.2 Adjust Subject Vehicle Speed	In Review	User1		Reviewer1
1.3 Brake within Set Distance				
1.4 Minimum distance for emergency stop	In Review	User2		Reviewer2
1.5 Maintain Following Distance	In Review	User2		Reviewer1, Reviewer2
1.6 Maintain Set Speed				

## Approval process for a requirement

You can review the requirement by going through its content and choosing to reject or approve with or without comments.

## Restrictions and limitations

You cannot modify the content of a requirement or requirement specification in **Released** status.

## Prerequisites

To make the content of a requirement or requirement specification with the **Release Status** of **In Review** or **Rejected** editable in the **Documentation** tab, run the following commands in the command prompt:

1. Export existing ACLs.

```
am_install_tree -u=infodba -p=pw_infodba -g=dba -operation=export -
path=C:/Apps/tc/tc2412/TR/install/arm0activeworkspacereqmgmt/data/
existing_ACLS.xml -format=xml
```

2. Import ACL for Requirement Review process

```
am_install_tree -u=infodba -p=pw_infodba -g=dba -operation=import -
path=C:/Apps/tc/tc2412/TR\install/arm0activeworkspacereqmgmt/data/
Arm0_review_process_ACL.xml -mode=replace_all -format=xml
```



3. Import the existing exported ACLs from step one.

```
am_install_tree -u=infodba -p=pw_infodba -g=dba -operation=import -
path=C:/Apps/tc/tc2412/TR\install/arm0activeworkspacereqmgmt/data/
existing_ACLS.xml -mode=replace_all -format=xml
```

## Procedure


1. **Locate the requirement specification or requirement** that you want to view, then click **Open** .
2. In the **Tree** view, click **Table Settings**  > **Arrange**.

The **Arrange** panel displays.


3. Click **Column Arrangements** , select **RM Approvals** and click **Arrange**.
4. Click **Layout Manager**  on the primary toolbar, and select a three-section layout.

The three sections display the following:

- The **Tree** view
- The **Documentation** tab view.

- The **Details** tab view.
5. Do one of the following to search for the requirement that is assigned to you for review:
- Click **Filter**  and in the filter list click **Released status** and select **In Review**.

Note:

This option is available only for those structures that are indexed using Smart Discovery Indexing. Such structures can be identified by the *Indexed* indicator .

- In **My Signoffs** column, look for your name.
6. In the **Details** tab section, click **Approvals** tabs.

The following sections appear:

- **Task to Perform**— Displays the task instructions and the options to approve or reject the requirement.
  - **Current and Completed Tasks**— Displays both current and completed tasks, with a list of reviewers and their review statuses.
7. In **Task to Perform**, enter text in the **Comment** box, and then select a decision from following options:

- **Approve**— Give the review your approval.

Note:

If you are the only reviewer assigned to the requirement, it moves to **Released** status. In the case of multiple reviewers, the requirement will not reach **Released** status until all reviewers approve.

- **Reject**— Complete the review without your approval.

Note:

- If you reject a requirement, the **Release Status** updates to **Rejected**.
- When the requirement is in **Rejected** status, the **Task to Perform** section displays the **Comment** box where you can enter text and click **Complete**. After clicking **Complete**, the requirement status goes back to **Review** and the **Task to Perform** section displays the options to approve or reject.

## Submit requirements to workflow for approval

You can start a review process for any requirement specification, requirement, or paragraph. You most commonly select a signoff team and submit the requirements for their review and approval.

1. **Locate the requirement specification or requirement** that you want to view, and do one of the following:
  - In the search results, right-click the requirement specification and click **Submit to Workflow**.
  - Click **More commands ...** > **Manage** > **Submit to Workflow**.
  - Right-click the requirement specification in the structure on the left and click **Submit to Workflow**.

The **Submit to Workflow** panel appears.

2. Enter a description for the workflow participants, select the appropriate workflow template, and then click **Submit**.

If a default workflow is defined for the requirement component, it is automatically selected as the workflow template.

Depending on the workflow, you may be required to assign the signoff team to approve the requirement. You receive a task to create the signoff team in your **INBOX** tile on the **My Tasks** tab. Active Workspace then assigns an approval task to each signoff participant that you select.

3. Do any of the following:
  - You and your team should check your **Inbox** for workflow tasks and to view the related attachments, workflow, and targets (where you can preview the requirements).

Note:

You receive an email when changes to requirements with trace links occur.

- To check the workflow status, open the requirement specification or requirement in the **Table Summary** view, then click the **Workflow** tab.

Note:

You receive an email when changes to requirements with trace links occur.

## Locating and viewing requirements


### Requirement locate and view process

As a system analyst, system designer, or a system tester, you can locate and view requirements. When product development begins, you may have many established requirements in your corporate library. As system modeling and testing progresses, you often need to locate requirements.

1. **Generate and save a report.**
2. **Locate a requirement.**
3. View or obtain the requirement in any of the following ways:
  - **Display a requirement.**
  - **View a requirement hierarchy.**
4. You can also take the following actions on requirements:
  - Compare requirements in either Microsoft Word or HTML formats.
  - **Download the requirement.**



### Tips for locating requirements

You track requirements throughout the life cycle of product development. As a system analyst, you ensure that requirements are measured and reported to the right stakeholders. You view reports and dashboards to verify that the requirements management process is operating correctly. You can monitor the requirements in the following ways:

- View the **Relations** tab for a selected requirement, which provides the relationship among requirements, targets, deviations, and so on.
- Participate in the discussion for a selected requirement using Active Collaboration.
- View the **Workflow** tab for a selected requirement that is pending team review to see the workflow status. As a system analyst, you are most often required to select the signoff team for reviews and sign off on requirements, deviations, and change requests. You can view your workflow tasks in the **My Tasks** tab of your **INBOX** .

### Locate requirement specifications, requirements, and paragraphs

You can locate requirement specifications, requirements, and paragraphs, by the following methods:

- Search for the content using the type **Specifications** or **Specification Templates**.
- Navigate to the content by clicking your **EXPLORER** tile or the **Explorer**  icon in global navigation.
- Navigate to the content by clicking your **FAVORITES**  tile.
- View the trace links on a system model block to find associated requirements.

## View requirement specifications, requirements, and paragraphs

As a system analyst, system designer, or system tester, you can view requirement specifications, requirements, and paragraphs.

1. **Locate the requirement specification, requirement, or paragraph** that you want to view, and then click **Open** .

The requirement specification opens in the **Documentation tab editor**.

Note:

Opening a specification or requirements opens the latest revision. To view previous revisions or baselines, click **Details > History**.

2. Navigate the requirement specification in the following ways:
  - Click a requirement in the content pane to open and display that requirement in the work areas. Conversely, click a requirement in the **Documentation** tab to select the requirement in the content pane.

Tip:

Click a link in the requirement specification breadcrumb to return to the top of the specification in both the content pane and **Documentation** tab.

- Click in the **Documentation** tab and use the scroll bar.

Tip:

If the scroll bar control is no longer viewable, scroll down and then up to reset the scroll bar control.

Your administrator sets the number of requirements that the **Documentation** tab loads at a time.

Note:

If you create requirement content in Microsoft Word and insert a callout, the callout text appears inline with the requirement content in the **Documentation** tab.

3. Do any of the following (if available):


- Click **Details > Parameters** to view the parameters entered against the requirement. Parameters are more common with system models and system blocks.

If the parameters are contained in separate files, these are identified in the relevant property columns, for example, **Goal File** and **Value File**.

- Click **Details > Where Used** to show the parent assemblies, products, and references to the selected requirement specification, requirement, or paragraph.
- Click **Details > Attachments** to view any files or documents that support the requirement content.


Note:

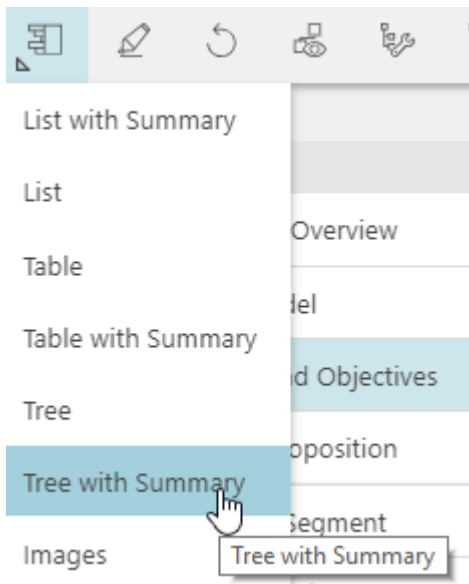
Custom notes are not displayed in **Attachments**.

- Click **Details > Relations** to view the relationship between the selected requirement specification, requirement, or paragraph with other requirements, system models, and so on.
- Click **Details > History** to view the history for the selected requirement specification, requirement, or paragraph.
- Click **Details > Reports** to view reports associated with the requirement.
- Click **Discuss**  on the primary tool bar to create a new discussion or view and reply to existing discussions about the requirement. See [What are discussions?](#) for information.


## View requirement hierarchies in the Table or Tree view

You can view the requirement hierarchy in several views, but you will find the **Table with Summary** and the **Tree with Summary** views most helpful when working with requirements. These views display both the requirement specification structure and the requirements in the **Documentation** tab editor.

1. **Locate the requirement specification** that you want to view, then click **Open** .
2. Select the **Table with Summary** or the **Tree with Summary** view.



By default, requirements and paragraphs are listed in the order they appear in the requirement specification. When you search for requirement components, they appear in the order returned by the search.

3. (Optional) Perform any of the following:
  - Click a column to sort by that column.
  - Click **Arrange**  to modify the columns that appear and to change the column order.

## Collaborating on requirements

### Collaborate on requirements

Requirements Management provides a collaboration service where you can develop and edit requirements simultaneously with other team members. This collaboration can improve communication among team members, create better specifications, and produce better outcomes.

The collaboration service enables you to see in real-time any edits made by collaborators with the location of the edit in the document and the content modifications being indicated. Each change that a user makes and then saves on a requirement is tracked and recorded on the History tab. Each user's edits are denoted in a specific color and all updates are immediately visible to all users in the collaboration session.

#### Note:

If you change the tab after editing a Requirement in collaboration, the cloud copy of updated contents is not deleted if you have a single sign-on (SSO) environment configured along with the

Transport Layer Security (TLS) certificates. Therefore, when you return to the **Documentation** tab, you will continue to see the old edits.

## Collaboration session

A collaboration session is active only in the Documentation tab. If the collaboration service is installed and running, you are automatically joined to a collaboration session when you open the Documentation tab. If you move to a different tab, then you automatically exit the collaboration session. Users in the collaboration session are listed in the upper right corner of the page. Each user's avatar also appears in the list. See **Install the Collaboration Service** for information on installing the collaboration service.

### Note:

Users who do not have rights to modify requirements can see the real-time changes as they occur, but they can not make changes of their own.

## Simultaneous edits

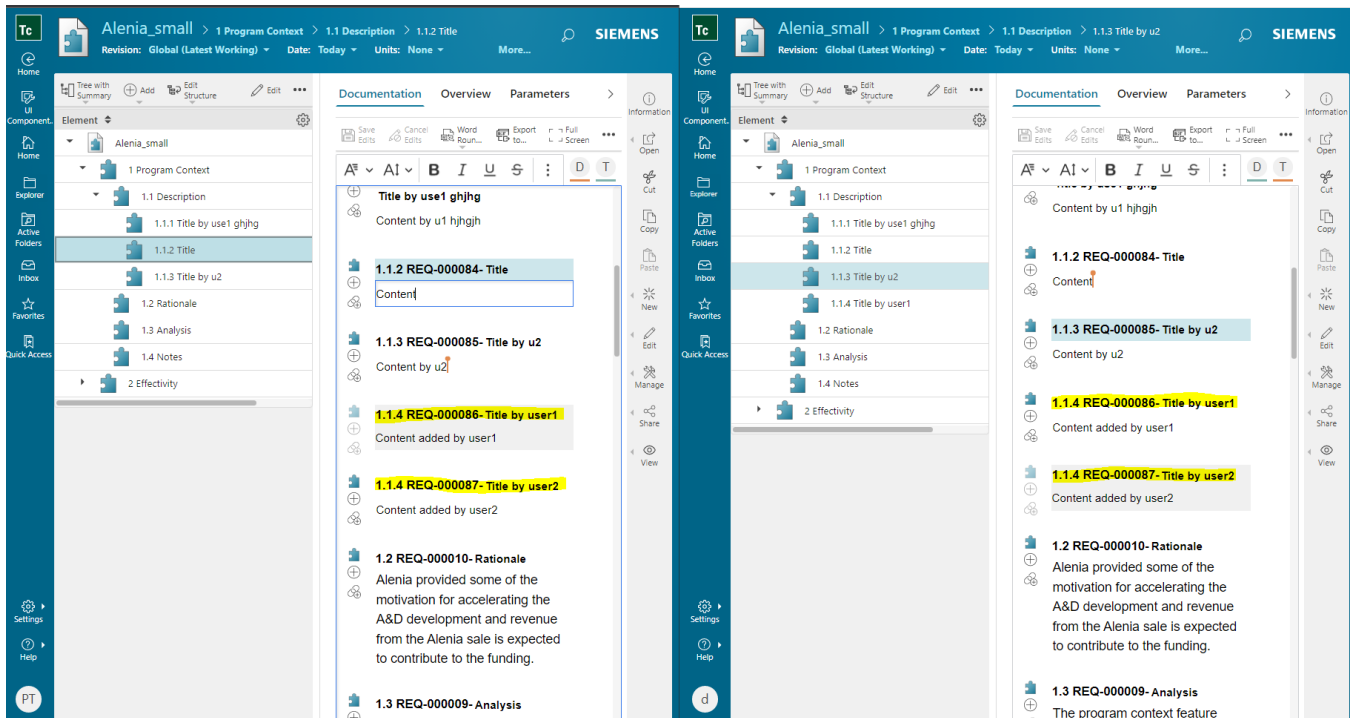
If multiple users in the same collaboration session edit the same requirement simultaneously, the **Save Edit** action performed by either of them will persist all changes, regardless of which user made those changes.

If the same requirement is edited in two different collaboration sessions, an error message appears when a user performs a **Save Edit** action. This message indicates that a modified version of the requirement exists. You can click **Changes done by others** to review those edits. Then, you can click **Overwrite** to overwrite the changes from the other user, or **Cancel** and refresh your session to pull in the latest version of the requirement. If you pull the latest version, you can then proceed to make your own edits.

## Structural edits

Edits involving structural change, such as the addition of a new requirement can only be updated by the user who initiated the structural change until that change is saved in the system. For example, if User 1 modifies a structure to insert a new requirement, then only User 1 can rename that requirement before the requirement is saved. User 2 can see the addition of the requirement, but cannot make changes to it until User 1 completes the **Save Edits** action. User 2 also cannot create any sibling, child, or trace link to the requirement created by User 1 until User 1 saves the edit.

Also, if multiple users create new child objects within the same parent object during a collaboration session, paragraph number updates are delayed until one user refreshes the application. The following graphic shows where User 1 and User 2 have both added child objects under the same parent. Initially, both child objects have the same paragraph number of 1.1.4. The paragraph number for the second child object will update to 1.1.5 upon refresh.




## Track Changes and Show Comments

You can enable the functionality to track changes and show comments on requirements, and then other users in the collaboration session can see those changes and comments during the session. See [Track changes to requirement text](#) information on tracking changes. See [Manage comment threads for requirements](#) for information on showing comments.

## Manage comment threads for requirements

When multiple team members collaborate on a requirement, comment threads can be created and viewed by the team members. This is helpful for keeping track of who made the comment and any replies. These procedures show how to manage comments in the **Documentation** tab editor, but you can also manage comments in the **Summary Table** tab.

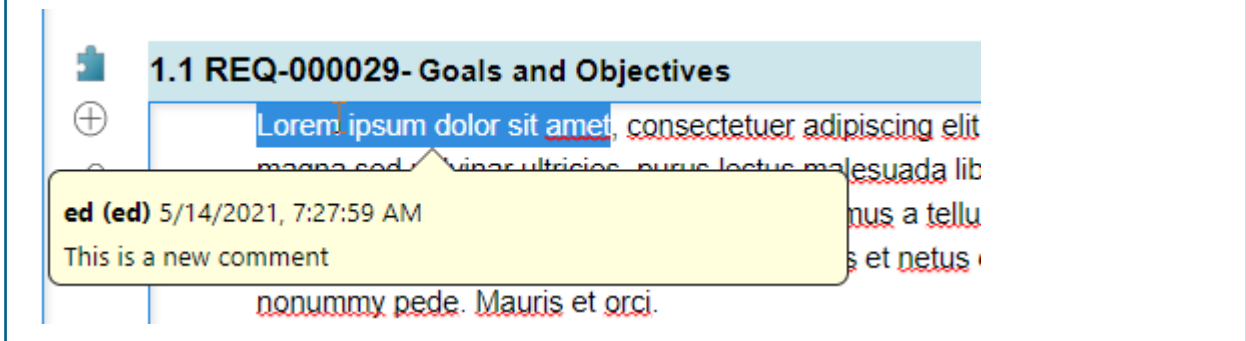
### Add comments


1. Open a requirement specification in the **Documentation** tab editor.
2. Click **Show Comments** .

The **COMMENTS** panel appears showing all comments and changes for the current specification.

Note:

You can also hover over highlighted text to view a pop-up of the related comments their authors.



3. Select the text you want to comment on and then click **New Comment** .

The selected text is highlighted and a new comment field appears in the **COMMENTS**.

4. Enter your comment and then click ✓.

### View, edit, or delete comments or change the status

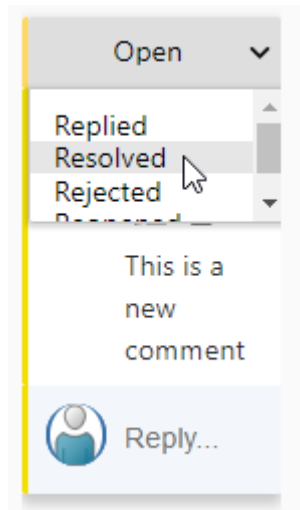
1. In the requirement you want to view comments for, click **Show Comments** .

The **COMMENTS** panel appears showing all comments for the current specification.

Note:

You can also hover over highlighted text to view a pop-up of the related comments.

2. (Optional) Perform any of the following:
  - **Edit** or **Delete** the comment.
  - Click the status drop-down to change its value.



## Filter comments

Comments are automatically aligned with the text that the comments reference.

Enter a search term to filter the comments by user, date, or status.

The screenshot displays the Teamcenter software interface. The top navigation bar includes tabs for Architecture, Documentation, Test Coverage, Overview, Parameters, Test Results, Classification, Diagrams, Where Used, Attachments, and History. Below the navigation bar is a toolbar with various icons for editing and viewing. The main content area shows a requirement document with sections like '029408-Aliena', '1 REQ-000018- Program Context', '1.1 REQ-000017- Description\*', '1.2 REQ-000022- Rationale\*', '1.3 REQ-000023- Analysis', and '1.4 REQ-000013- Notes'. The '1.1 REQ-000017- Description\*' section is expanded, showing text about Alenia implementing Teamcenter Engineering. On the right side, there is a 'COMMENTS' sidebar. At the top of the sidebar is a search filter: 'Filter by User, Status, Date'. Below the filter, there is a dropdown menu currently set to 'Replied'. Two comments are visible: one from 'demo' at 'Today 12:44AM' with the text 'This needs to be corrected', and another from 'ed' at 'Today 12:46AM' with the text 'Will correct it in next revision, is it fine?'. At the bottom of the sidebar, there is another dropdown menu set to 'Open' and a comment from 'ed' at 'Today 12:46AM' with the text 'Is it a correct Teamcenter version?' and a green checkmark icon.

## View revision history for selected requirement objects

You can compare requirements and view the changes made between them, and compare the document to another historical version. You can view who made a change, a summary, and view the entire document in the **History** tab. You can also manage comments and changes in the **Documentation** tab. For more information, see [Manage comment threads and track changes for requirements](#).

Note:

Your system administrator must perform the following prior to viewing specification history:

- The **Compare** microservice must be installed.
- **REQ\_Microservice\_Installed** preference must be set to **true**.

Note:

Each version is saved with a default limit of 25. Your administrator, however, can set a preference to limit the number of versions saved per revision. In this case, you will only see the number of instances allowed instead of the entire history.

1. **Locate the requirement specification** that you want to view, and then click **Open** .

The requirement content appears in the **Documentation** tab editor.

2. Select the **Tree view**.
3. Select the requirement objects for which you want to view revision history.
4. Click the **History** tab.

The requirement history for the selected item is shown in a table, with added content appearing in green and removed content appearing in red. Content that has been formatted appears in blue.

## Compare requirements

### Compare requirements authored in Microsoft Word

You can compare requirements and requirement baselines that are authored in Microsoft Word. The requirements may contain plain text or rich text. You can compare two revisions of a requirement, or two similar requirements from different programs or structures. The comparison is performed using Microsoft Word capabilities and the comparison result can be displayed in either a Microsoft Word document or a PDF.

For this comparison to work, some prerequisite software must be installed on the Dispatcher machine. Contact your application administrator for assistance.

To compare requirements or paragraphs:



## Compare during export and apply template

You can perform a compare during an export while using a specific template.

1. Select two requirements or paragraphs or revisions of a requirement or paragraph from one of the following:
  - Search results
  - **History** tab
  - Favorites

Note:


- You can compare child requirement specifications under a parent requirement specification. You cannot compare requirement specifications that do not have a parent requirement specification.
- If you are comparing against a baseline requirement, select the original requirement *before* selecting the baseline.

2. Click **More commands**  > **Share**  > **Export**.
3. In the **Export** panel, select **Word** or **PDF** option and then select the **Compare** check box.
4. Click **Export**.

If you selected **Word**, an MSWordX dataset is created and attached to the first object revision selected for comparison. If you selected **PDF**, a PDF file is created and attached.

Note:

If you compare the same requirements or paragraphs more than once, then the result MSWordX dataset or PDF file is overwritten. However, if you compare different requirements or paragraphs, a new MSWordX dataset or PDF file is created for each comparison under the first requirement or paragraph selected for comparison.

5. To view the MSWordX dataset containing the result of the comparison, do either of the following:
  - Click **Alerts**  and click the report.
  - Click the **Attachments** tab of the first object revision selected for comparison.



## Compare text or content

You can compare either the text or the structure of two specifications or requirements.

1. Select two requirements or paragraphs or revisions of a requirement or paragraph from one of the following:
  - Search results
  - **History** tab
  - Favorites

Note:

You can compare child requirement specifications under a parent requirement specification. You cannot compare requirement specifications that do not have a parent requirement specification.

2. Perform one of the following:
  - To compare the text, click **View**  > **Compare Text**.
  - To compare the structures, click **View**  > **Compare Content**.


You will receive a notification  when your report is ready.

## Compare two versions of a requirements specification


Note:

Review the settings for comparison configuration.

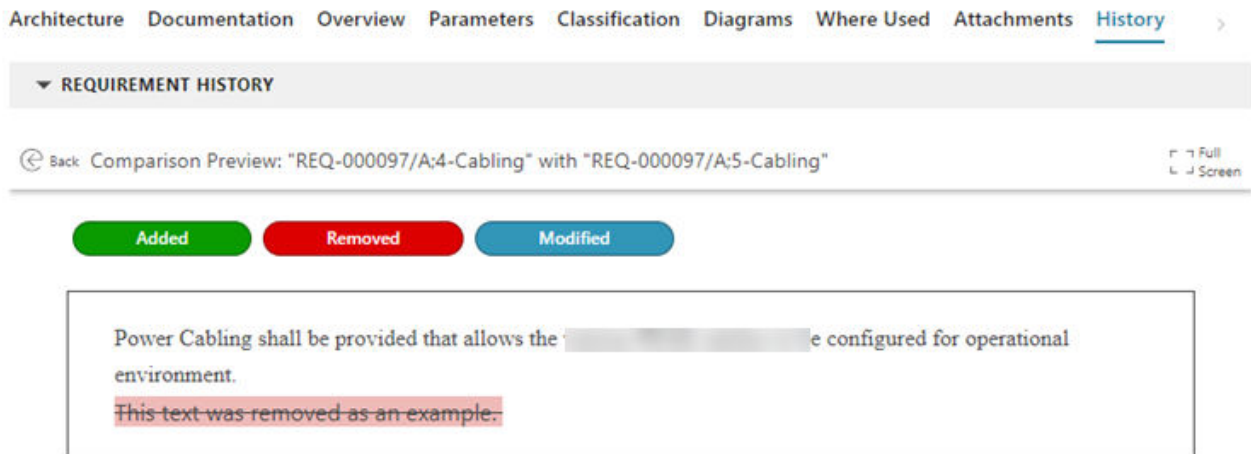
You can compare and view what changes were made between two sets of requirements and return to the specific change history. You can view who made a change, the change summary, view the entire document, or compare the document to another historical version.

1. From the **Home** screen, click **Explorer**  icon from the left panel.
2. Select a requirement document.
3. Click the **History** tab.

The history tab displays the list of changes for the requirement document.

- Select two rows in the requirement document by holding the **Ctrl** key and clicking **Compare Contents** .

The requirement comparison for the two versions appears on a single page. Added content is shown in green, while removed content is shown in red, and modified content is shown in blue:



Tip:

To return to the **History** tab, click **Back**.

Note:

The **Compare** icon appears only when two rows are selected. If more than two rows are selected, the icon is hidden.

Note:

If you are comparing against a baseline requirement, select the original requirement *before* selecting the baseline.

## Compare requirements authored in HTML

You can compare either the text or the structure of two specifications or requirement that are in HTML format.

- Select two requirements or paragraphs or revisions of a requirement or paragraph from one of the following:
  - Search results
  - History** tab

- Favorites

Note:

If you are comparing against a baseline requirement, select the original requirement *before* selecting the baseline.

2. Perform one of the following:

- To compare the text, click **View**  > **Compare Text**.
- To compare the structures, click **View**  > **Compare Content**.

You will receive a notification  when your report is ready.


### Compare and revise requirements upon import

You can compare requirements and revise them upon import, or discard any changes made.

Note:

These preferences must be set by your system administrator:

- **REQ\_Microservice\_Installed=true**
- The **compare** microservice must be installed.


1. **Locate the requirement specification** that you want to view, then click **Open** .
2. In tree view, click **New > Import Specification**.

## Import Specification ✕ Close

File: ?

Choose File PRO-X Refrigerator Market Requir...

**TO**

 PRO-X Refrigerator Market Requirements

Changes to Specification

Retain Paragraph numbers

Create default requirement type items for missing sections





Specification Type: \*

Requirement Specification

Default Requirement Type: \*

Requirement

**RULES**

 Save Rule  Add Rules  Move Up  Move Down

Saved Rules:

Run in Background

Compare

Import




3. Select the file and click **Open**.
4. From the first drop-down, select **Changes to Specification**.
5. In the import panel, make your selections and click **Compare**.

Added content is highlighted in green, while removed content is highlighted in red.

6. Click the gear icon and select one of the following:
  - **Update:** Default selection. If a change is detected, the content is updated.
  - **Revise:** Revises the requirement and sets the most recent content on the revised revision object.
  - **Delete:** Deletes the object from the second document.
  - **Discard Update:** Select this option if you decide not to update a particular requirement from the second document.

## Generate a report

You can generate and view reports related to a selected requirement to monitor the requirements process.

1. **Locate the requirement specification (or other requirement)** that you want to generate the report for, then click **Open** .
2. Click **More commands**  > **New**  > **Generate Report**.

The **Generate Report** dialog opens and displays reports related to requirements.

3. Click a report.

Each report has one or more style sheets associated with that report. These style sheets are created by your application administrator and control the properties that display in the report and the display order. Style sheets also define the report look and format. Each report may also have additional report-specific fields.

Tip:

Reports of interest for requirements include:

- Traceability trail.
- Packed-complying defining traceability.
- Structure comparison of two BOM structures.



If these reports are unavailable, contact your application administrator.

4. Select a style sheet and complete all report fields.
5. Click **Generate**.

The **Generate Report** panel closes and the **Reports** tab refreshes with the generated report.

## Download requirement components for offline viewing

You can save requirement components, which includes requirement specifications, requirements, or paragraphs, to your local drive for review when you are not using Active Workspace.

1. **Locate the requirement component (requirement specification, requirement, or paragraph)** that you want to download, and then click **Open** .
2. Click **More commands ...** > **Share**  > **Download Requirements**.

Note:

If you opened a requirement specification but want to download only one of its requirements, select the requirement before downloading.

3. Use the options that appear in your browser to save the requirement component to your local drive and also to view them in your browser.

## Establishing product design targets

### Establish design targets

You develop design verification methods that detail how the system tester should validate your system model solution through product targets.

Requirements management and verification planning come together through the design verification plan. This plan is designed to answer the question, "Did we design the product right?" The system designer collaborates with other team members such as the system analyst and the system designer to create the plan based on the requirements. The purpose of each verification is to determine that the output at any given stage meets a defined requirement. The plan details the activities that the system testers, such as a 3D simulation engineers, perform to verify the product design.

Common verification methods include the following activities:


- Worst case analysis
- Fault tree analysis
- Design comparison
- Safety

1. **Locate the requirements.**

2. **Define the assessment method** to detail how the system tester verifies that a product aspect meets a requirement.
3. **Assign the targets** that the product must meet. The system tester verifies these targets through simulation.
4. **Define the desired analysis results as goals.**
5. Review the assessment results.

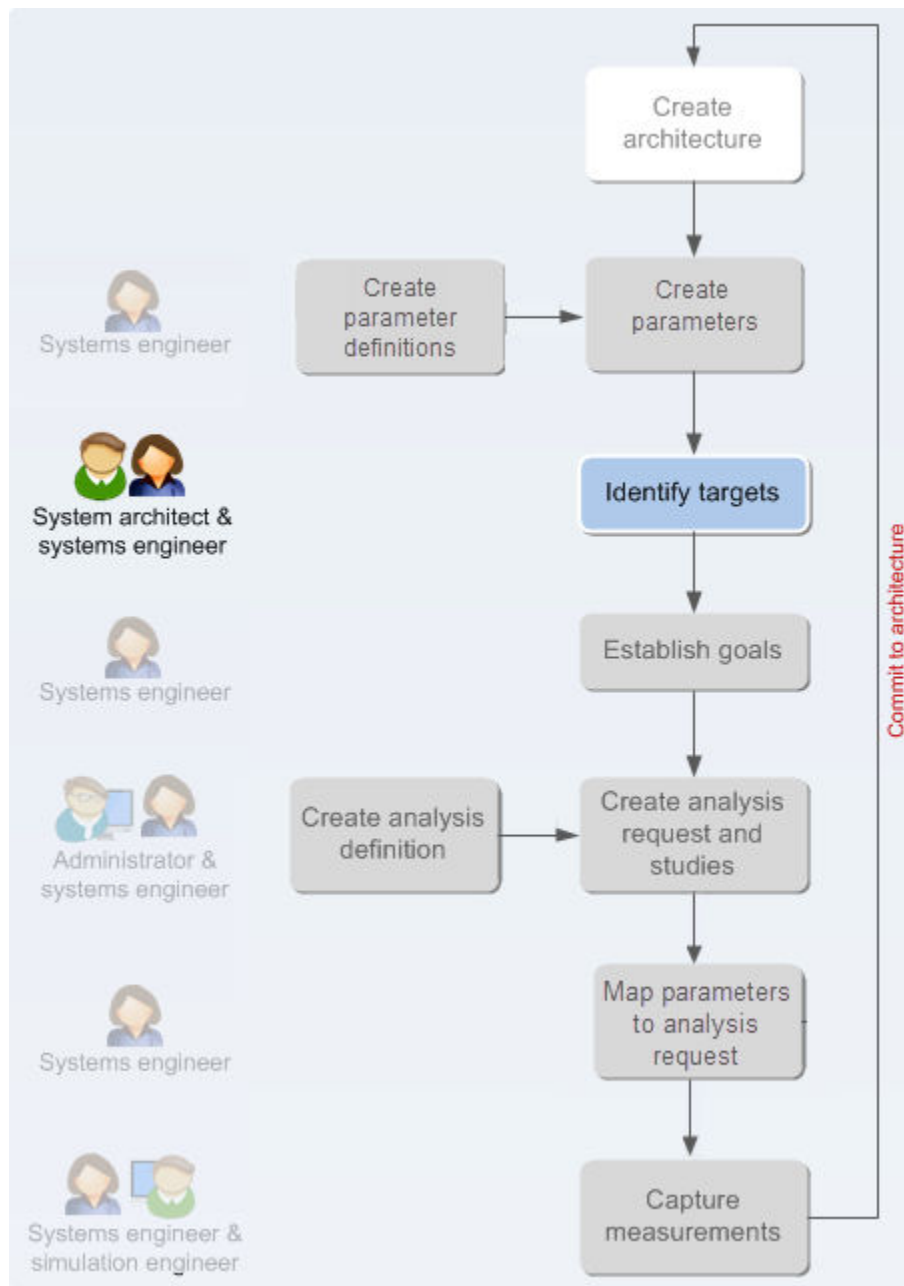
## Create the assessment method and set the product targets

You can create assessments that define the methods for verifying that a product design meets the requirement or requirement set. You can also set targets related to the assessment method. A target (product target) is a specific, measurable outcome that is allocated to the product and can be decomposed onto subsystems. A target identifies a requirement that establishes the need for technical measurements. The target is usually created as the result of business strategy, market analysis, or government regulation.

1. Determine the assessment product targets for the requirement.
2. Set the **parameters** or **measurable attributes** on the targets.
3. **Locate the requirement** that you want to view, then click **Open** .
4. Click **Details > Attachments**, and then add supporting assessment files, documents, and URLs.

## Applying targets for the product

### Working with targets




A *target* (product target) is a specific, measurable outcome that is allocated to the product and can be decomposed onto subsystems. A target identifies a requirement of a program or product that establishes the need for technical measurements. The target is usually created as the result of business strategy, market analysis, or government regulation.

For example, in the fuel efficiency and emissions integration domain, you may have overall fuel consumption targets of 39 miles per gallon for 2016 and 54 miles per gallon for 2025. Alternatively, you may have a target to achieve a 10% reduction in fuel consumption for the new model.

If the product target is simple (for example, weight), the product target's goal value can also be decomposed onto the contributing measurable attributes.

#### Create a target for the product

You can identify a target that the product must meet in one or more markets. Targets are typically captured as a requirement in the product requirement specification.

1. Navigate to the folder containing program targets, and then click **More commands ... > New**  **> Add**.

The **Add** panel appears.

2. In the **Add** panel, filter and select **Target**.
3. Enter the required information, and then click **Add**.

The system creates the target.

4. Select the target, and then click **Copy** .
5. **Locate the requirement content (requirement specifications, requirements, or paragraphs)** that you want to edit, and then click **Open** .

The requirement content appears in the **Documentation** tab editor.

6. Create a trace link to a requirement by using the keyboard shortcut Alt + L, and paste the target into the link as **Start** or **End**.
7. Click **Create**.

The system creates a trace link between the requirement and the target.

#### Establish a parameter for a target

You can add one or more parameters to a target to allow the achievement of the target to be assessed quantitatively. For example, a target for fuel economy may have parameters for weight, drag coefficient, and tire friction attached to it.

1. Click **More commands ... > New**  **> Add**.

The **Add** panel appears.

2. In the **OTHER** field, filter for measurable attribute and select it.
3. To open the **Measurable Attributes** panel and then do one of the following:
  - Select an existing measurable attribute.
  - Add a new measurable attribute by clicking  $\oplus$ , and then define the required values and click **Add**.

You can attach several measurable attributes to a single target.

## Relating a target to the model

You can relate the target to the elements that fulfill the objectives of the target. For example, you may relate an improved fuel economy and emission target to the fuel system block in the system model.

You can relate a target to the model in one of two ways:

- In the **diagram**
- In the **Trace Link tab**

## Add parameters to requirements

You can add parameters quickly to a requirement by using the **Add Parameter** panel or the **Quick Add** feature.

Note:

Parameter definitions are not required by default.

If you prefer to require a parameter definition, you can **set a preference** to establish the requirement. The parameter inherits all of the available properties on the parameter definition.

For more information about parameters, see **Managing Global parameters**.

## Prerequisites

You must have a specification with a requirement where you can add parameters before proceeding with the procedure.


## Procedure

1. Open the specification where you want to add parameters, select the requirement, and click **Details > Parameters**.

The **Parameters** table appears.

2. Do you want to use the **Add Parameter** panel or the **Quick Add** feature?

- To use the **Add Parameter** panel, do the following:

- a. Click **Add**  to use the **Add Parameter** panel.

The **Add Parameter** panel appears.

- b. Enter a name for the parameter.

- c. Select a **Parameter Definition** if required.

The parameter inherits values from the remaining fields from the parameter definition.

- d. Select a **Data Type**.

- e. (Optional) Set values for the remaining parameter fields.

- f. Click **Add**.

The new parameter appears in the parameter table.

- To use the **Quick Add** feature, do the following:

- a. Click **Quick Add**.

A row appears in the **Parameters** table.

- b. Enter a name for the parameter.

- c. Select a **Parameter Definition** if required.

The parameter inherits values from the remaining fields from the parameter definition.

- d. Select a **Data Type**.

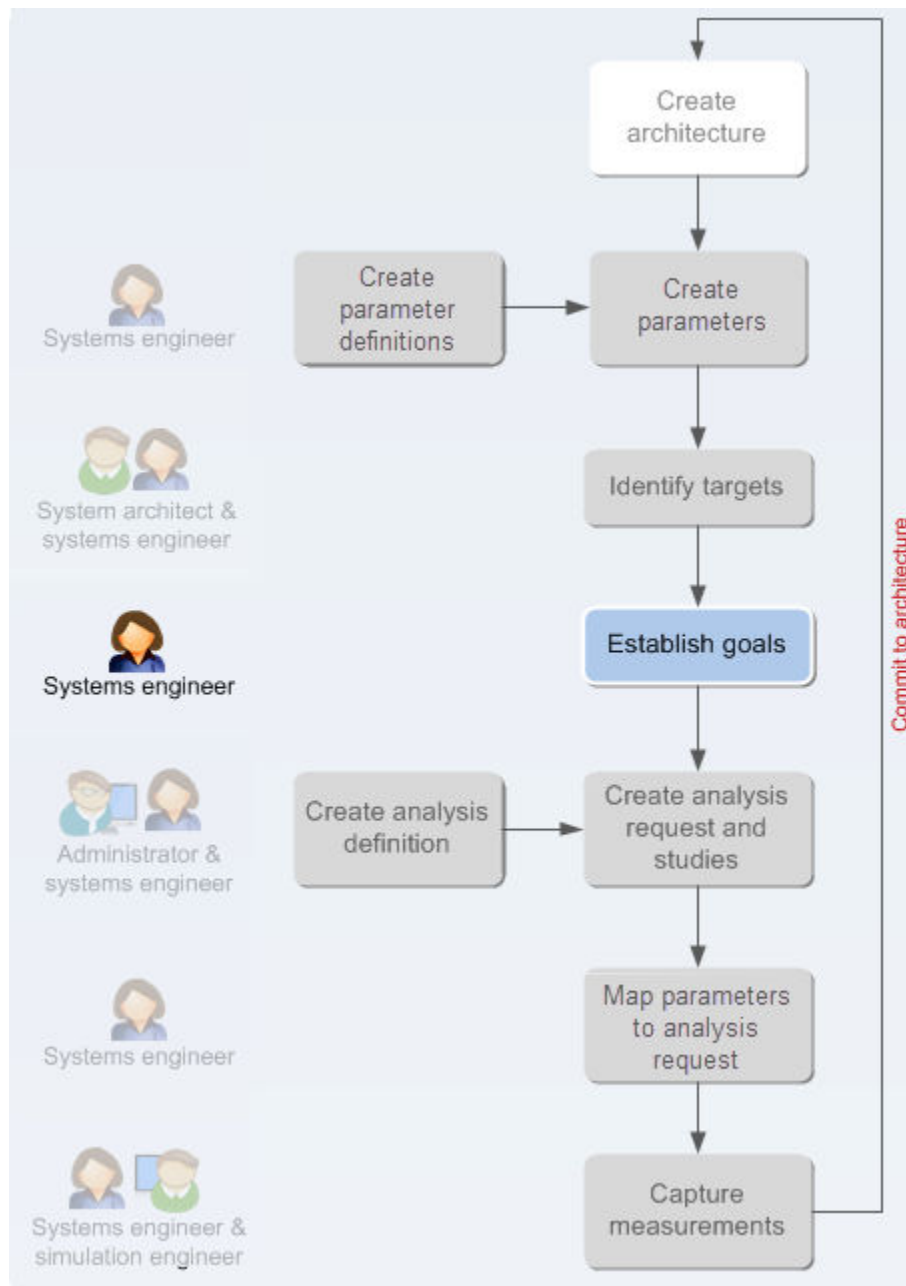
- e. (Optional) Set values for the remaining parameter fields.

- f. Click **Save Edits** .

The new parameter appears in the parameter table.

## Defining the desired analysis results

### Establishing a goal



A *goal* is the result you want to obtain from an analysis activity. You typically choose a goal that ensures the relevant targets are met. For example, if you have a full consumption target of 40MPG, you may set a goal of 50MPG, that is, a goal that exceeds the target.

## Set goals in the context of the product

The system designer sets the parameter goal (required measurement) in the context of the product on the system structure, typically on the block. You can override attribute values in the context of the product to determine if the product is meeting expectations and evaluate solution viability. You can override an attribute value if:


- The **Allow Override** check box is selected (the property set to **True** on the attribute definition).
  - You have write access to the attribute value.
1. From the domain content, select a block, connection, requirement, or target.

2. Click **Details > Parameters**.

The system displays a list of parameters associated with the selected object.

3. Select one or more parameters that you want to override.

The system shows the properties of the parameter. Only parameters with the **Allow Override** property set to **True** can be overridden.

4. Click **Edit**  **> Start Edit**, edit the parameter values you want to override, and then click **Save**. You can only edit values to which you have write access.

### Measurable Attributes ✕

**WEIGHT (GM)**

**PROPERTIES**

Attribute Definition: ATM\_AR\_000002/A;1-Weight

Name:

Description:

Goal:

Min:

Max:

Allow Override:

**Add**



# 4. Modeling systems




## What is system modeling?

### System modeling with MBSE



The product definition is a phase in an integrated MBSE approach where the necessary assets are authored such as requirements, parameters, system architectures, interface diagrams, and domain architectures to produce 1D/ 3D models, along with software and test specifications. Your team uses these elements to build models that represent system and subsystem physics, and the connection of those systems through interfaces across all relevant development domains. Once available, these reusable digital models can be used in later development stages of simulation and product validation. This results in a fully specified and traceable product aligned to stakeholder requirements.

System modeling with MBSE allows you to govern domains and coordinate product development through an approach that connects the system models, their creators, and relevant development data. Therefore, software, mechanical, and electrical engineers can easily collaborate in a co-development environment.

The following table shows the high-level roles engaged with system modeling:

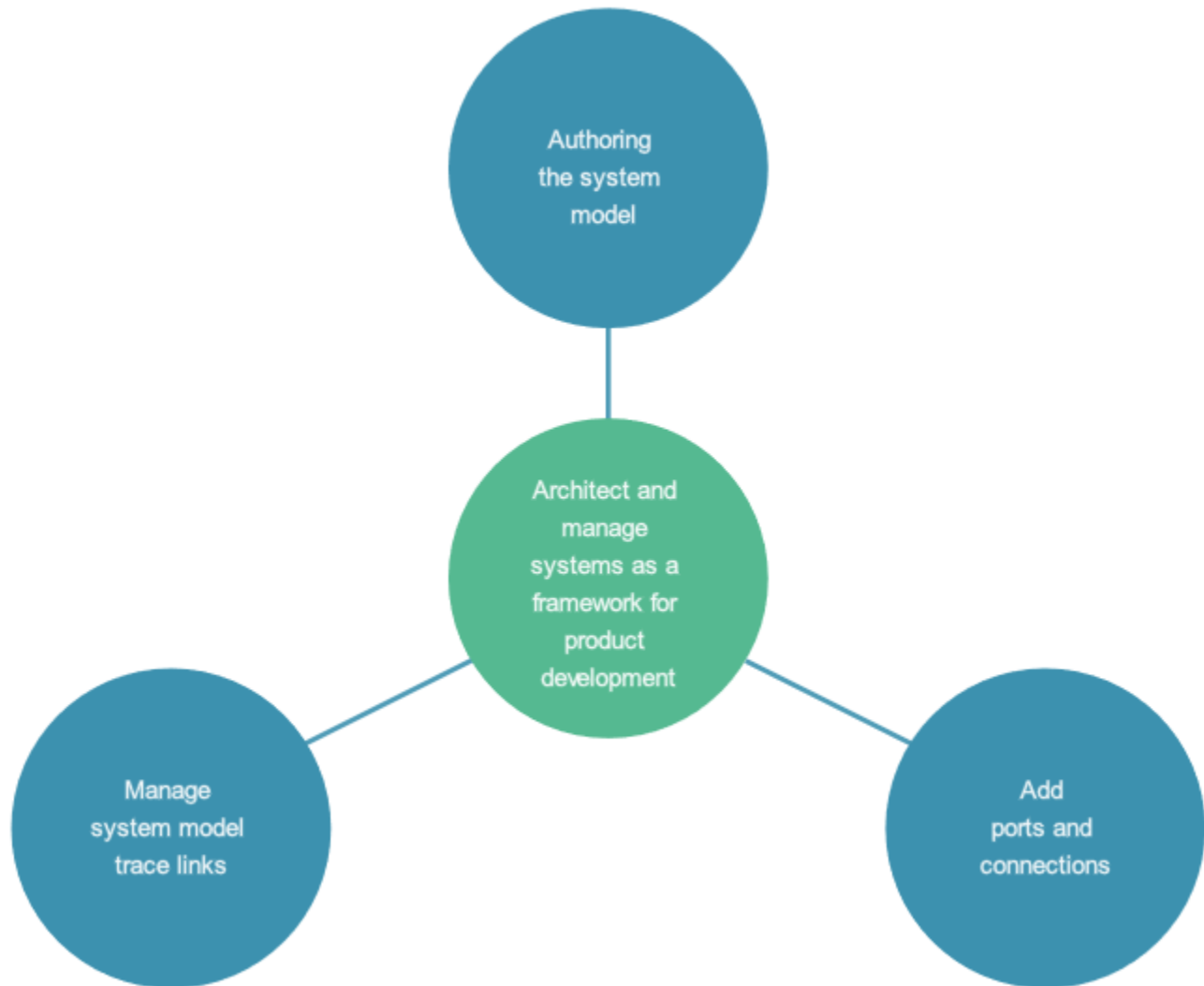
 Administrator	As an application administrator, you should be familiar with configuring requirements management, system modeling, analysis requests, targets, and attributes.
 System architect	Manages the overall product system model architecture that meets the requirements.
 System designer	Manages specific domain areas (software, electrical, mechanical, and so on) of the system model.

### Where do I go from here?

 Administrator	See Configuring system modeling.
 System architect, system designer	
Learn more about how the MBSE features work together in Active Workspace.	See What is Model-Based Systems Engineering (MBSE).
Begin authoring model diagrams	See Overview of model diagramming.

## System modeling business process

The system modeling business processes provides the creation of the system model solutions in the context of the program or project. Solution models may be classified in a library, for example, when several systems designers are collaborating on a program or if they are candidates for possible reuse.



Business process	Description	Roles
<i>Plan program or project</i>	<p>(This is a prerequisite process that assumes the product is a new concept.)</p> <p>Provides the steps to set up the study project, including:</p> <ul style="list-style-type: none"> <li>• Create the project context.</li> <li>• Create the study project. The project acts as a collector of artifacts in the study.</li> </ul>	<p>System architect</p> <p>Configuration developer</p>

Business process	Description	Roles
	<ul style="list-style-type: none"> <li>• Create the system context. Search for and copy architecture elements where appropriate.</li> <li>• Define the conceptual system architecture. Cascade or decompose requirements from functional model to system model when appropriate.</li> <li>• Conduct a feasibility study. Search for and compare similar solutions.</li> </ul>	
<i>Model the solution system</i>	<p>Create a logical decomposition of the system that satisfies the solution needs. Tasks include:</p> <ul style="list-style-type: none"> <li>• Author the system model, which is a conceptual solution that may span multiple features.</li> <li>• Decompose the system model into blocks that eventually correspond to sub-assemblies and components.</li> </ul> <p>If necessary, consume the next level of requirements and iterate.</p> <ul style="list-style-type: none"> <li>• Define and manage the system model interfaces and ports that link the model elements.</li> <li>• Validate and release the system model into the program architecture.</li> </ul>	<p>Technical architect</p> <p>System analyst</p> <p>System designer</p>
<i>Consume model solutions from library</i>	<p>Provides the classification of system models in a library for collaboration between participants or subsequent reuse. Tasks include:</p> <ul style="list-style-type: none"> <li>• Add models to the library.</li> <li>• Retrieve model solutions from the library.</li> </ul>	<p>System architect</p> <p>System designer</p>
<i>Prepare data for simulation</i>	<p>Associate parameters with system model blocks, allowing their characteristics to be simulated. Successful achievement of targets confirms that requirements are satisfied. Tasks include:</p> <ul style="list-style-type: none"> <li>• Assign requirements and function occurrences to model elements using <b>trace links</b>.</li> <li>• <b>Allocate parameters</b> and targets to model elements.</li> <li>• <b>Request analysis</b>.</li> </ul>	<p>System architect</p> <p>System designer</p>

## System Modeling tabs

Use the following tabs to build and to explore system models:

<b>Overview</b>	Summarizes a system modeling item, its properties, history, and status in the product lifecycle.
<b>Requirements</b>	Covers the requirements involved in, specifying, or extending a system model item.
<b>Diagrams</b>	Shows various static visual representations of a system model item.
<b>Documents</b>	Shows related document artifacts for a system model item.
<b>Architecture</b>	Shows various static graphic representations of a system model item.
<b>Diagrams</b>	Provides a static visual exploration of structure, connectivity, and traceability across many system modeling items within a configured scope.
<b>Documents</b>	Shows related document artifacts for a system model item.
<b>Architecture</b>	Provides a visual exploration of structure, connectivity, and traceability across many items in a configured scope
<b>Interfaces</b>	Provides contextual navigation for connectivity among systems.
<b>Relations</b>	Provides browse traceability between system model items without a particular scope, variability, effectivity, or configuration.
<b>Parameters</b>	Provides the technical assumptions about the system model item, such as: weight, performance, cost, and so on.

## Creating diagrams of models

### Overview of model diagramming

Active Workspace allows users to capture requirement specifications and models as hierarchies of requirements, paragraphs, function blocks, logical diagrams, and system blocks. In Active Workspace, you can navigate and build these hierarchies using model diagrams.

Model diagrams are interactive representations that use a number of visual indicators that allow you to expand and collapse the hierarchy to navigate up, down, and throughout its various branches.

For example, you want to view the requirements for a component you are redesigning, so you can make requirement-constrained design decisions. You search for the component and view it in a model. You see indicators for related functions and click the indicators to expand the functions. You also see indicators for related requirements, and you click those indicators to expand the requirements.

Working with a model diagram, you can:

- View and navigate through representations of related data, including the hierarchies of requirements, and functional, system, and physical (part/design) models in a working context.
- View relation types between elements, including parents, children, connections, and trace links.

- Add and remove elements from structures.
- Copy and paste elements within a structure and between structures in a working context and in other working contexts.
- Add trace links and connections between blocks in a model.
- Open requirements from within a model.
- Modify the properties of elements from within a model.
- Access standard features such as context, access control, checkin and checkout, change management, workflows, variants, and effectivity.

## Open model diagrams

Using model diagrams in Active Workspace, you can navigate the hierarchy of requirements as well as functional, system, and physical models. You can also use the diagram to modify the models, including adding and removing content and modifying the relationships between content elements. You can open any element and display its location in the model. However, you must open a working context to view a model with multiple top levels, for example, requirements or functional, system, and physical models.

1. Open the model or the working context of the model you want to navigate.
2. Click the **Content** tab, and then click the **Architecture** tab.

Tip:

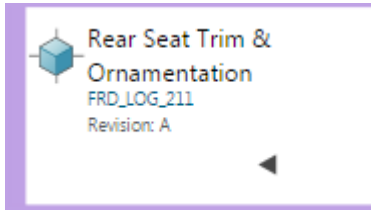
Select the **List with Summary** display to see the **Architecture** tab.





By default, the diagram is empty. In the content list on the left, the thumbnail that is unavailable (appears grayed out) indicates that the element is not displayed in a diagram.


However, if you previously viewed the diagram or opened the diagram from the favorites, the previously displayed context, items, ports, connections, trace links, filters, zoom level, and focus appear.

3. Click the thumbnail to the left of the object's name to display the object in a diagram.




- (Optional) Click **Network view**  and **Nesting view**  to switch between the two types of displays. The **Nesting view** shows parent and child relationships and is used for viewing models.

The **Network view** displays general relations and is used for viewing related objects.

- (Optional) Click  to change the model display orientation you want to work with: **Top-to-Bottom**, **Right-to-Left**, **Left-to-Right**, **Incremental**, or **Bottom-to-Top**.


Note:

The balloon orientation is not available in the **Network view**.

- (Optional) Click **Relation Filters**  to filter elements in the graph such as connections and, traceability. Options that appear in the filter are configured by the administrator.

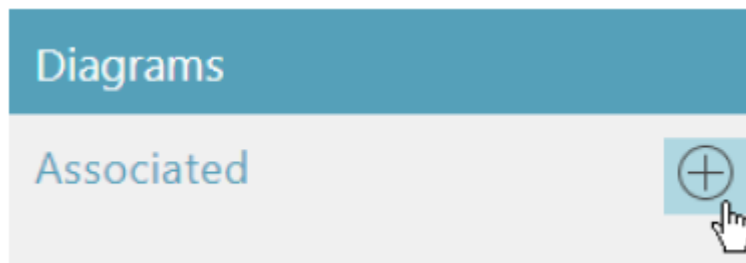
## Create a diagram from a system block

You create diagrams that define the structure of a selected system.

- Open the structure content, select a system block, and then click the **Architecture** tab.
- Click **Open**  > **Diagrams**.

The **Diagrams** panel appears.

- Click **Add Diagram**.




Note:

If multiple diagram types are available, select a type.


## Preview and open diagrams associated with a system block

You can display a list of diagrams associated with a selected system block. You can then select and display a diagram from the list.

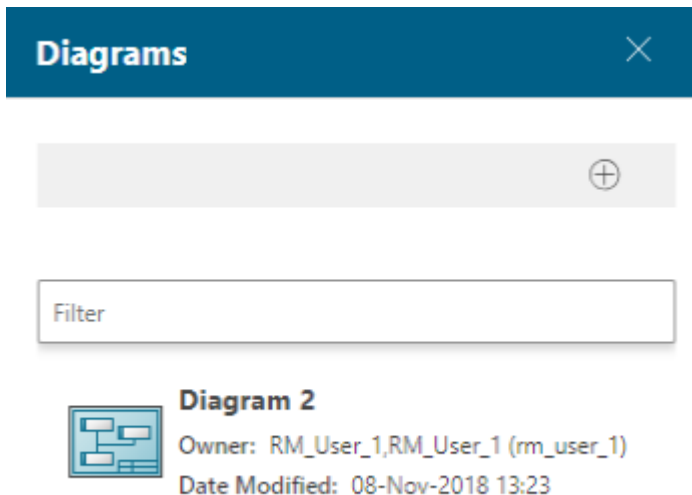
1. Navigate to a system block and click the **Diagrams** tab.
2. Select a diagram from the list to view the diagram in the **Preview** section.
3. To open the diagram, click **Open** .

## View and manage system diagrams in Diagrams tab

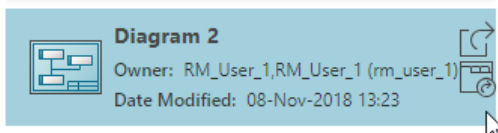
You view and open diagrams that define the structure of a system. You can also add, delete, and duplicate diagrams. When you are ready to publish, you submit one or more diagrams to an approval workflow and save the diagram as a preview snapshot.










1. Open the structure content, select a system block, and then click the **Architecture** tab.
2. Click **Open**  > **Diagrams**.

The **Diagrams** panel appears. The panel lists only those diagrams that share the same context as the selected system block.



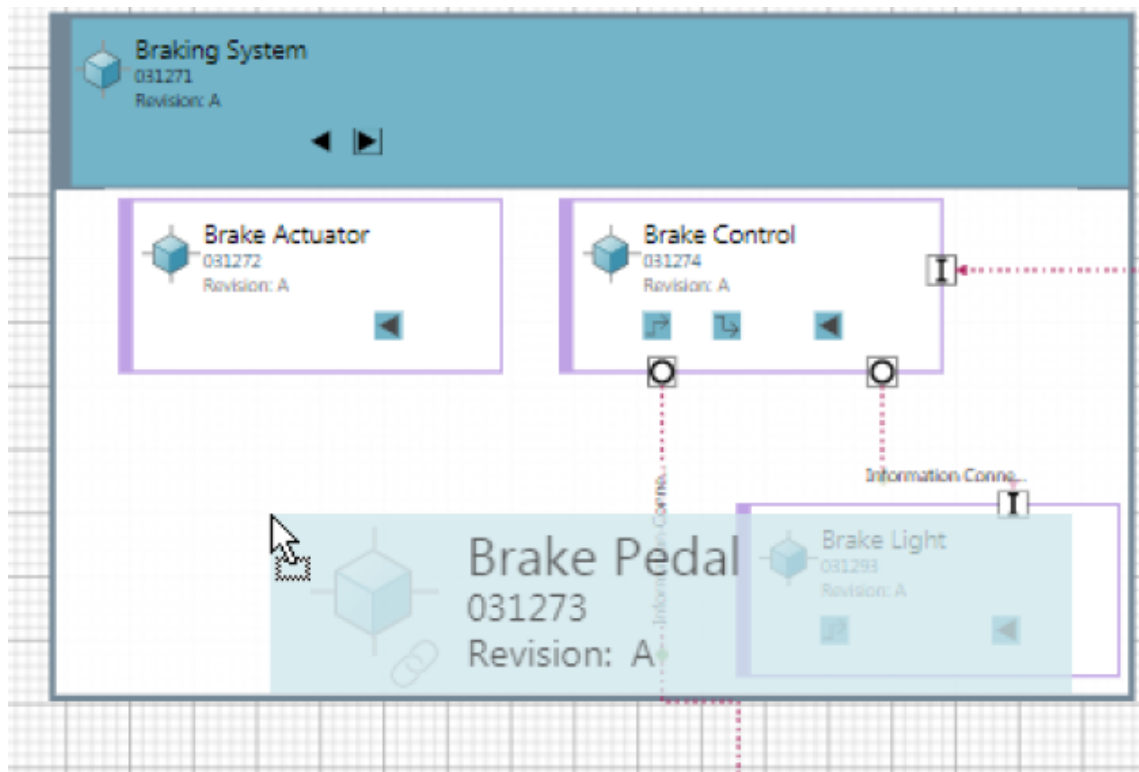
3. Click a diagram to show the methods available to open the diagram.



4. Do one of the following:
  - Click **Open**  to open the diagram in the current browser tab.
  - Click **Open in New Tab**  to open the diagram in a new browser tab.
5. Click the **Diagrams** tab to view all diagrams related to the system and then select a diagram to see a preview of the diagram.
6. Perform any of the following:
  - To change the column order, click **Arrange** .
  - To save the diagram as a preview snapshot, open a diagram in the **Architecture** tab, and then click **Save Diagram** .
  - To create a diagram, click **Add diagram** . This icon is available if you have no diagrams selected.
  - To delete diagrams, select one or more diagrams and then click **Delete diagram** .
  - To duplicate a diagram, select a diagram and then click **Save as diagram** .
  - To publish diagrams, click **More Commands**  > **Manage**  > **Submit to Workflow**.


## Drag content into a diagram

You can drag content from the left-hand content list, search panel, or relations panel on to your diagram.



## Manually position systems in a diagram

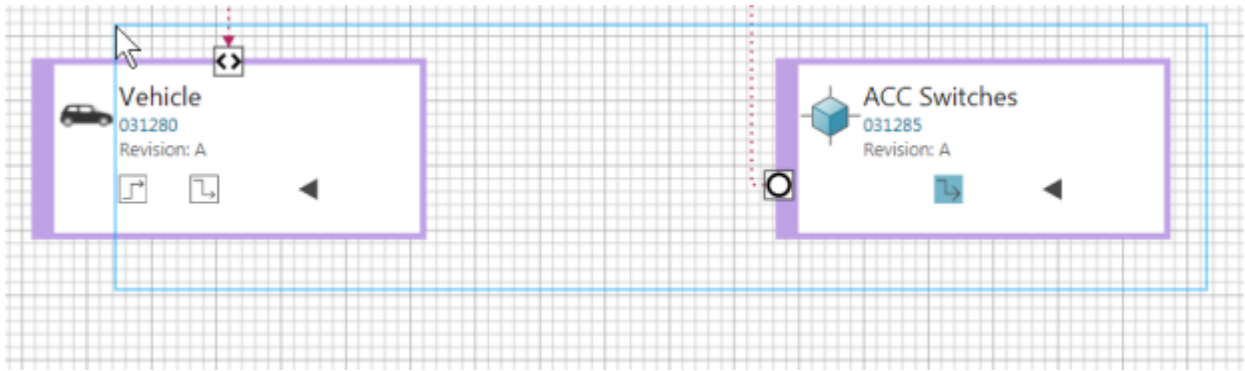
You can turn off auto-positioning so that you can manually adjust the system layout in your diagrams, such as overlapping systems.

1. Open a diagram.
2. Click **Automatic Layout**  to deactivate it.


## Multi-select, move, and align systems in a diagram

You can select multiple systems in a bounding box and then move or align the systems.

1. Press **Ctrl**+left-click and hold the mouse button.
2. Drag the cursor to create the bounding box around the systems.



3. Do one of the following:

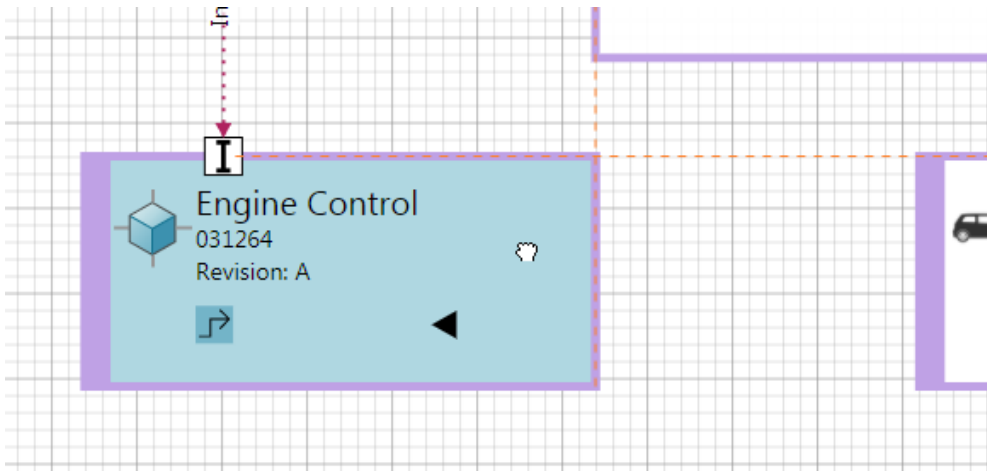
- Release the mouse button, click **Apply Layout** , and then select the orientation: **Top-to-Bottom**, **Right-to-Left**, **Left-to-Right**, **Bottom-to-Top**, or **Incremental**.
- Drag the selected systems to a new location, then release the mouse button.

The selected systems align.

## Align systems manually

You can manually align systems in your diagram, using guide lines.

Select and drag a system. Dashed orange lines appear to help you align with other systems.



## Snap to grid

You can have your diagram content automatically align with grid lines to lay out your diagram more efficiently.

1. Click **Diagram Settings** .

2. In the **Diagram Settings** panel, do the following.
  - a. Select **Show Grid** to toggle the grid on and off.
  - b. Select **Major Lines** or **Minor Lines** to determine which lines display.
3. Click **Apply**.

## Add an annotation to a diagram

You can create a rectangular text box annotation in a diagram. Annotations are diagram-specific and they do not reflect the structure content. The annotation is layered behind the system structure in the diagram.

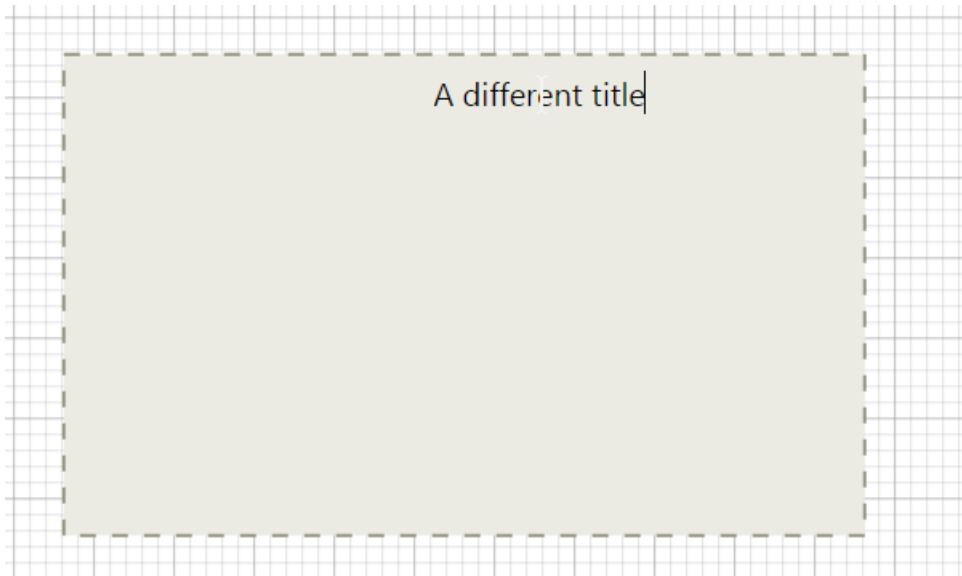
1. Open a diagram.
2. Click **Start Authoring**  and then click **Legend** .

The **Legend** panel displays.

3. Under **Annotations**, click **Rectangle**.



4. Click a location in the diagram to create the annotation.
5. Click the **Title** and rename the annotation.



6. Click **End Authoring** .

## Manually route connections in diagrams

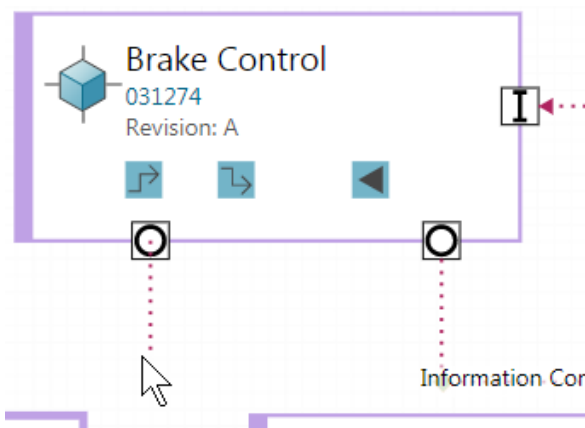
You can manually create different connection configurations in your diagrams.

### Simple connection

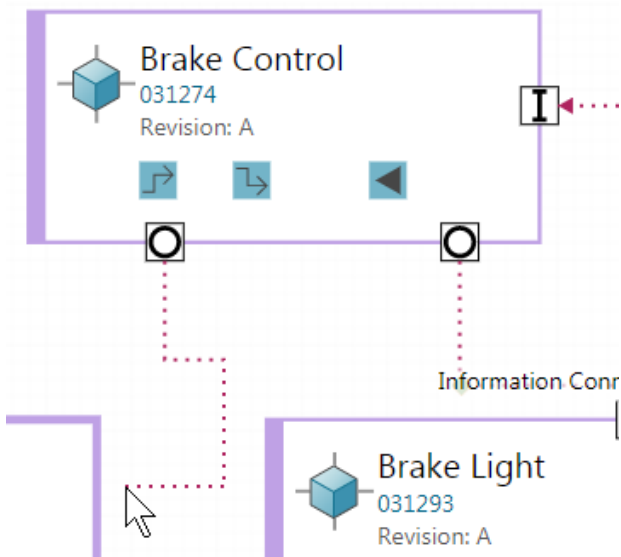
Click and drag from the starting connection point to the ending connection point.

### Single bend

1. Click a connection point and then begin dragging the cursor.

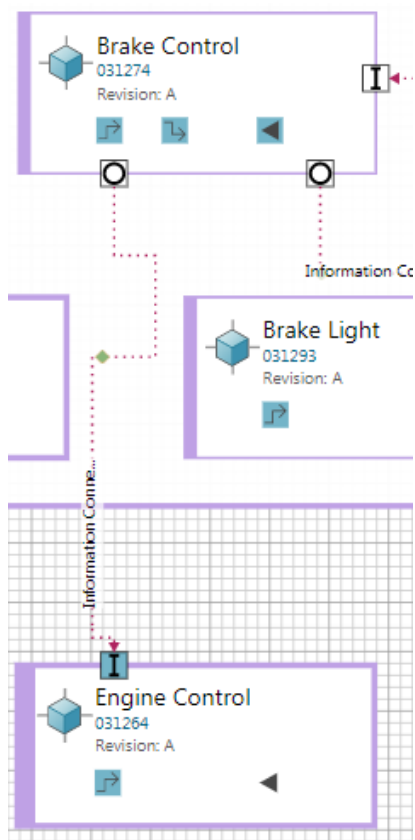


2. Release the mouse button at the bend point and then continue dragging the cursor.



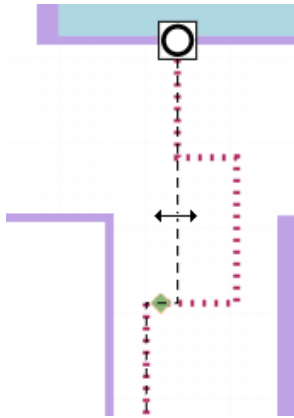
### Multiple bends

Click a segment and then click on a valid ending connection point.



## Adjust segments

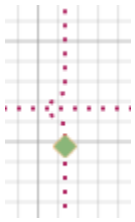
1. Ensure that no connection is selected.
2. Click a segment and drag the cursor ←→.



## Jumpers and connection arrows


Active Workspace provides visual indicators for relationship lines and port directions.

When two or more relationships cross between systems in the diagram, a jumper appears.



When you change port directions, the end-line symbol for the connection where it meets the port changes also. Port changes can result in routing changes.

**Output port:** No arrow. 

**Input or bi-directional port:** Arrow points towards system block port. 


## Submit a diagram to a workflow

You can submit a diagram to a workflow for team review and approval.


1. Open a diagram and click **Details > Architecture**.

The diagram appears in the main work area.

2. Do one of the following:

- Click **More Commands** > **Manage**  > **Submit to Workflow**.

The **Submit to Workflow** panel appears.

- Right-click the diagram in the structure and then click **Submit to Workflow** .

The **Submit to Workflow** page appears.

3. (Optional) Select a template for the workflow.

4. Enter a name for the workflow.

5. (Optional) Click **Assignments** and update workflow assignment information, as necessary.

6. Click **Submit**.

## Delete a diagram from a system

You can delete a diagram that defines the structure of a system.

Note:

When you delete a diagram, the diagram is removed from the associated system. The diagram is *not* deleted from Active Workspace.

1. Open the structure content, select a system block, and then click the **Architecture** tab.

2. Click **Open**  > **Diagrams**.

The **Diagrams** panel displays. The panel lists only those diagrams that share the same context as the selected system block.

3. Select one or more diagrams and then click **Delete** .


4. Confirm the deletion.

## Author component details


You can edit the details for components such as system models, system blocks, and connections.

**Note:**

Active Workspace uses CKEditor in the **Documentation** tab to author documentation. For information about the CKEditor interface, see the [CKSource website](#). Your environment may not support all features.

1. Select the **List with Summary**, **Table with Summary**, or **Tree with Summary** view.
2. Click **Documentation**.
3. Double-click the text box to begin editing.
4. Edit the content and click **Save** .




## Print model diagrams

1. Open a diagram.
2. Click **Architecture**.
3. Click **More commands ...** > **Share**  > **Print**.
4. On the **Print** panel, select the desired template, and then click **Print**.

An HTML file downloads.

5. Open the HTML file.
6. Use your browser print function to print the diagram.

## Using anchor nodes

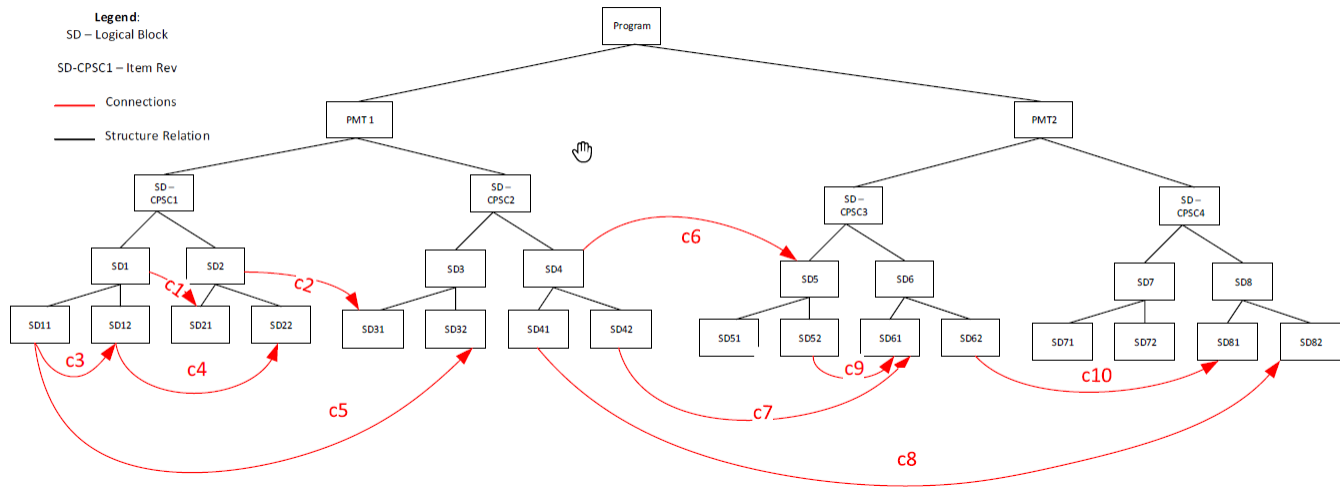
- You can specify that an element in the diagram is a root element by setting it the element as an anchor node. Click **Set Anchor**  to set or unset an element as an anchor node.
- To set anchor nodes on multiple elements, select multiple elements and click the **Set Anchor**  command.
- To unset anchor nodes on multiple elements, select multiple elements that are anchor nodes and click the **Unset Anchor**  command.

If you hide an element that is an anchor node, all its associated elements that are not anchor nodes will also be hidden.

A diagram should have at least one anchor node, otherwise the diagram will be empty.

## Understanding rolled-up connections

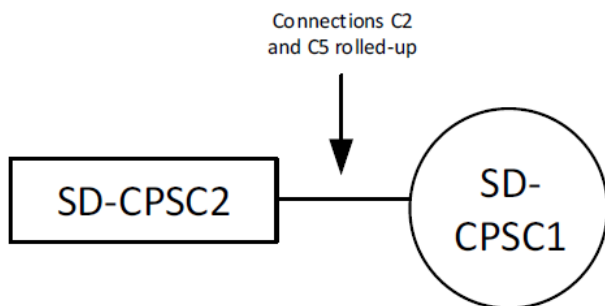
The following example graphic shows how Active Workspace rolls up connections to the appropriate level and displayed as a single line between two systems. The functionality depends on certain business type constants set by your administrator, which we will indicate in this example.



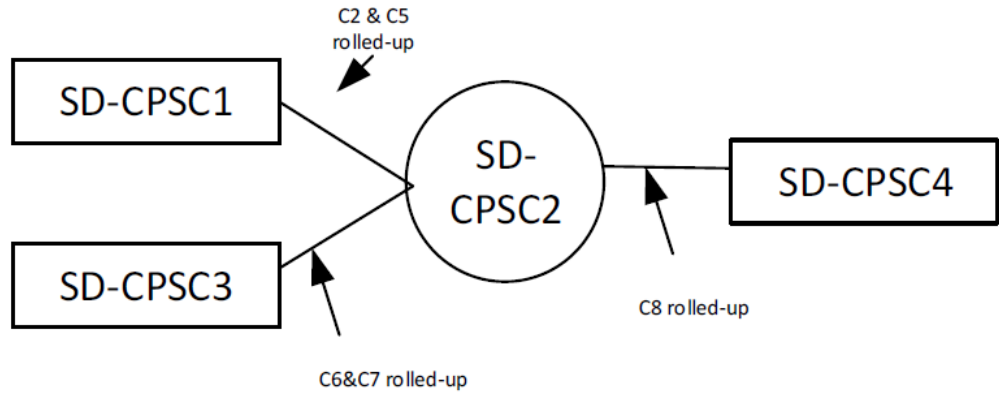
### Show peers

Your administrator has set business object type constant **AseoShowPeerInterfaces** to **true** or has not set the constant.

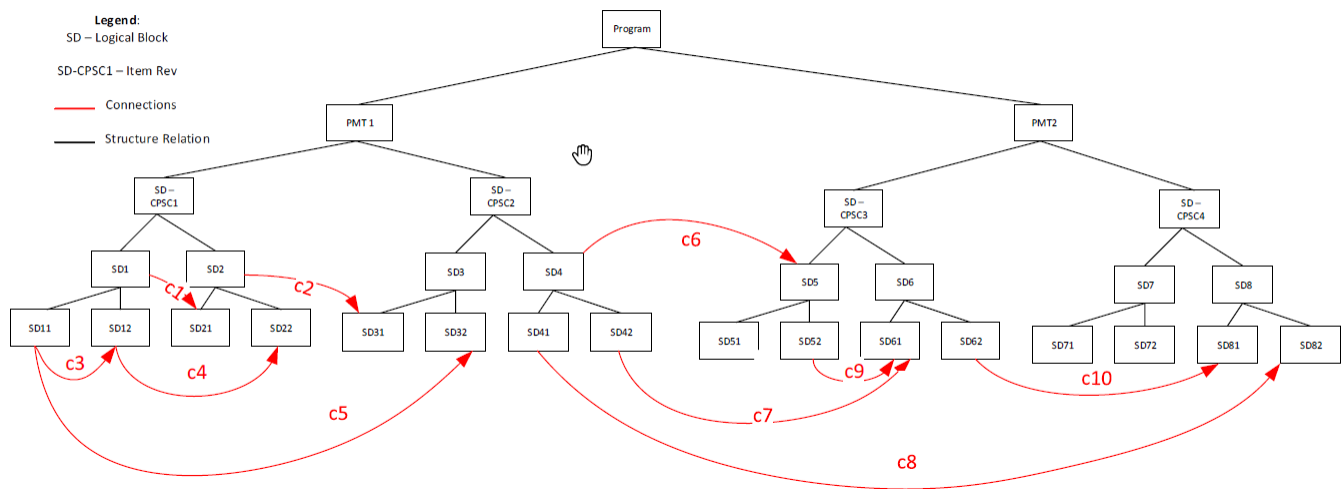
Based on the example, if you select **SD-CPSC1** as the system of interest in the content panel, the **Interfaces** tab displays a single rolled-up connection to represent connections **c2** and **c5**.



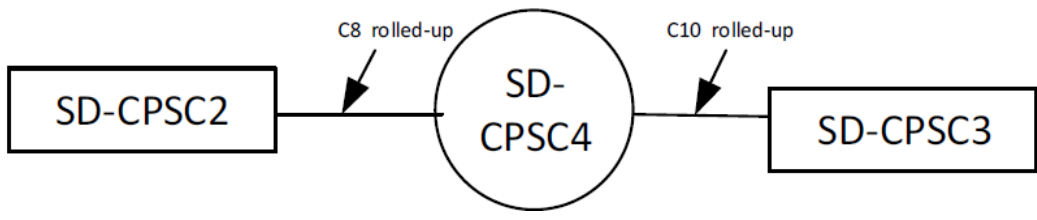
Similarly, if you select **SD-CPSC2** as the system of interest in the content panel, the **Interfaces** tab displays a single rolled-up connection to represent connections **c2**, **c5**, **c6**, **c7**, and **c8**.



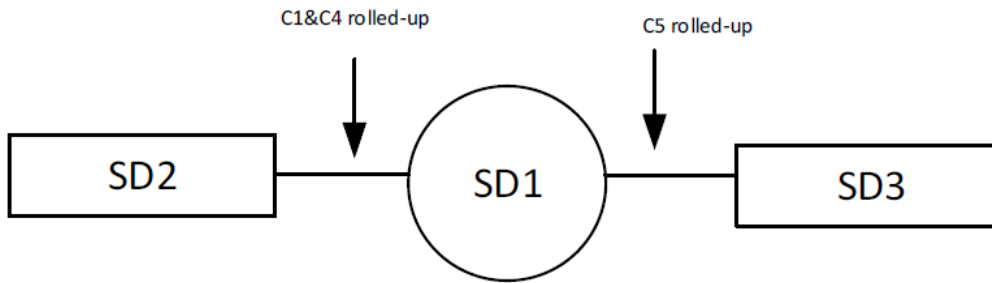
If you select **SD-CPSC3** as the system of interest in the content panel, the **Interfaces** tab displays a single rolled-up connection to represent connections **c6**, **c7**, and **c10**.



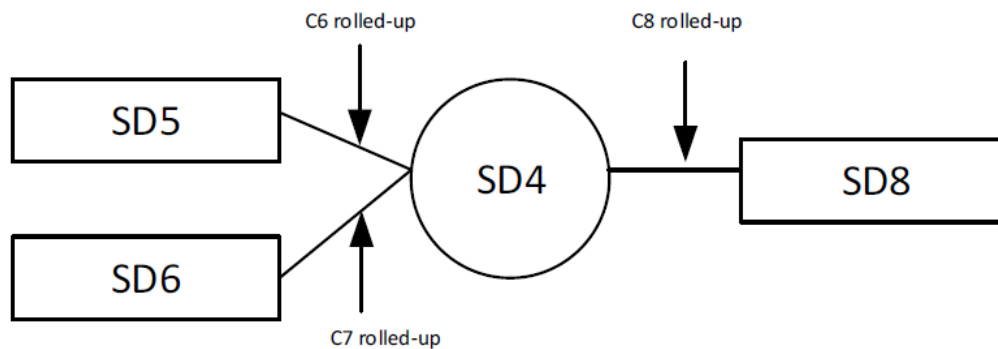
If you select **SD-CPSC4** as the system of interest in the content panel, the **Interfaces** tab displays a single rolled-up connection to represent connections **c1**, **c4**, and **c5**.



Also, if you select **SD1** as the system of interest in the content panel, the **Interfaces** tab displays a single rolled-up connection to represent connections **c1**, **c4**, and **c5**.



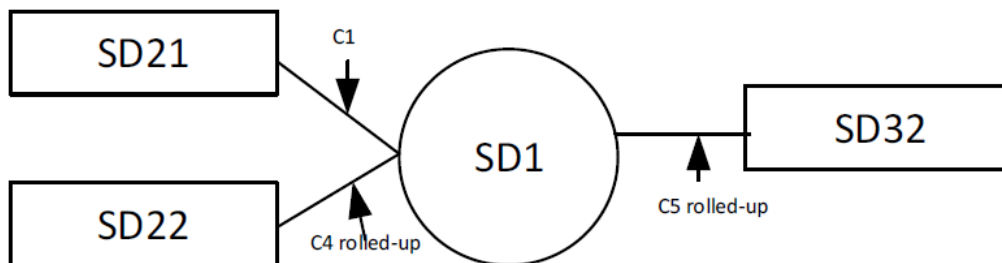
If you select **SD4** as the system of interest in the content panel, the **Interfaces** tab displays a single rolled-up connection to represent connections **c6**, **c7**, and **c8**.



### Show direct connections

Your administrator has set business object type constant **AseoShowPeerInterfaces** to **false** for object types Logical Blocks.

If you select **SD1** as the system of interest in the content panel, the **Interfaces** tab displays the following:

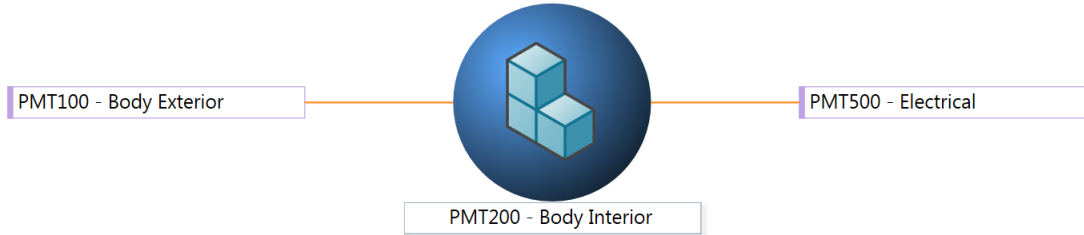


### Find external connections

You can locate external connections to your systems. An *external connection* to a selected system connects that system to any of its children at any level to one or more other systems. For a selected system, the **Interfaces** tab shows one line representing rolled-up external connections to each of

the external systems. Your administrator can configure the **Interfaces** tab to show peers or direct connections from a selected system to the external systems.

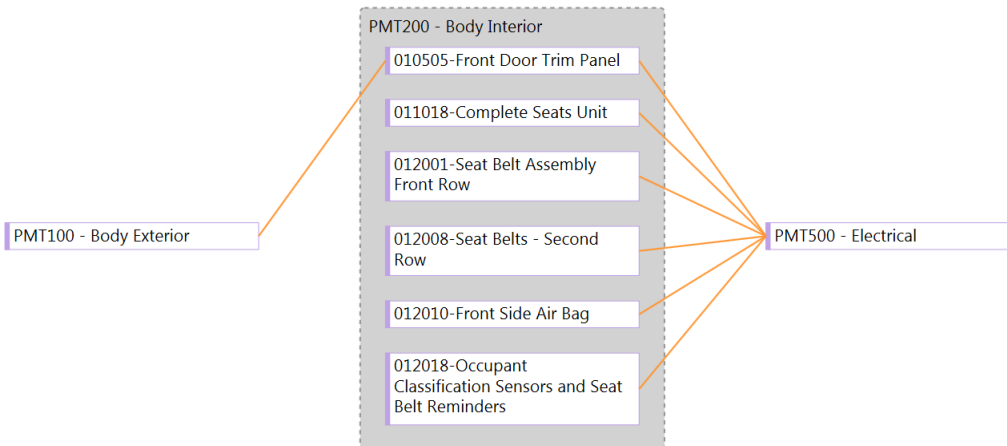
These procedures use the following rolled-up connection example to clarify the functionality.




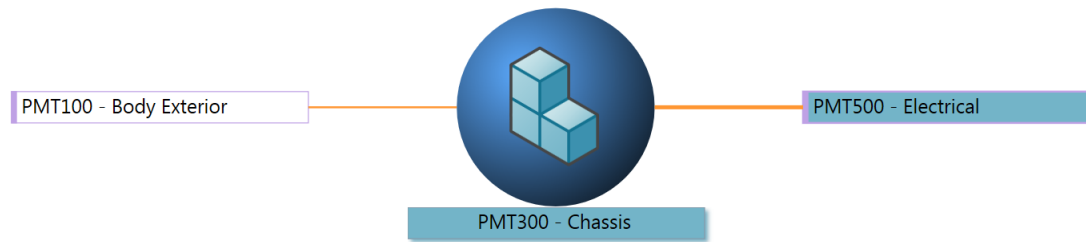
The selected system, **PMT200 – Body Interior**, is represented by the blue circle in the middle. The external systems that communicate with the selected system and the rolled-up connections between them are also shown.

Do any of the following:

- To change the selected system (the blue circle in the diagram), double-click an external system (**PMT100 - Body Exterior** or **PMT500 - Electrical**, for example).
- To view the children of the selected system, double-click the selected system. For example, double-click **PMT200 - Body Interior**.



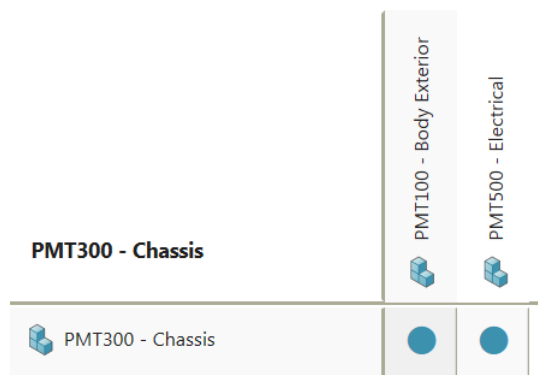
- To move up a level, click **Go up** ↑.
- To show the details of a selected roll-up connection, click **Show Connections Table** . The table shows the underlying connections that comprise the rolled-up connection line.



## Connections

CONNECTION NAME	CONNECTION OWNER	RELEASE STATUS	END1 DIR...	END2 DI...	END1 SYSTEM
I.Module - Body control module (BCM)...	user_1_1 (user_1_1)		Bi-Directional	Bi-Directional	FNA4047773/1-2C219-MOD ASY BI
Info Connection	user_1_1 (user_1_1)		Bi-Directional	Bi-Directional	FNA4049741/1-Module - Steering C


- To reposition the table, click  to switch between **Graph View** and **Grid View**.



Do any of the following:

- To change the selected system, double-click a column heading.
- To view the first level children, click a row heading. The children shown are either directly or indirectly connected to the external systems.
- To view connection details, click a blue circle.

CONNECTION NAME	CONNECTION OWNER	RELEASE STATUS	END1 DIR...	END2 DI...	END1 SYSTEM
E.010201-Front Structure---E.2C444-SN...	user_1_1 (user_1_1)		Bi-Directional	Bi-Directional	FNA4048464/1-2C444-
P.010201-Front Structure---P.2C444-SN...	user_1_1 (user_1_1)		Bi-Directional	Bi-Directional	FNA4048464/1-2C444-
P.010102-Body Dash and Cowl---P.2C4...	user_1_1 (user_1_1)		Bi-Directional	Bi-Directional	FNA4048464/1-2C444-

- To toggle labels, click .

## View parts with 3D data and trace links to systems

You can view 3D data (JT) associated with system blocks in the **3D** tab. Conversely, you can view trace links associated with parts in the **3D** tab.

**Note:**

The BOM structure with 3D data (JT) must be under the same root node or the saved bookmark that is currently in the left-hand content pane. You must create either occurrence or revision-to-revision trace links from system blocks (logical blocks) to relevant parts or subassemblies in the BOM structure.

For complete information about working with 3D data, see Visualizing 3D Product Data.

The following diagram shows the trace link as a green arrow between the system blocks.

Vehicle System > 100-Body Interior > 011018-Complete Seats Unit >

Viewer **Diagrams** Overview Requirements Interfaces Where Used Trace Link Documents DVP FMA

Custon Frame Assembly (Track Asy Front Seat / Module Asy Front Seat Custon)  
062315  
Revision: 1

PWRD ASSEMBLY-4 WAY-L/H  
1571-K61465-B  
Revision: A

1. Select a system block.
2. Click the **3D** tab to display the 3D data (JT).
3. To view trace links associated with a part, do the following:
  - a. Click a part in the **3D** tab, and then open the **Navigate Associated** panel.  
  
A list displays that shows the items in-context that have trace links to the selected part.
  - b. Select an item in the panel to close the panel and display the selected item in the left-hand content panel.

## View requirements with trace links to systems

You can view requirements that are associated with systems in the **Requirements** tab.

### Note:

Trace links (occurrence to revision-to-revision) must exist between systems (logical blocks) and requirements. The requirements must be under the same root context (root node or saved bookmark) that is selected in the left-hand content pane.

1. Select a system block.
2. Click **Details > Requirements** to view a table of associated requirements.
3. Click a requirement to display its content.

## Tips for navigating model diagrams

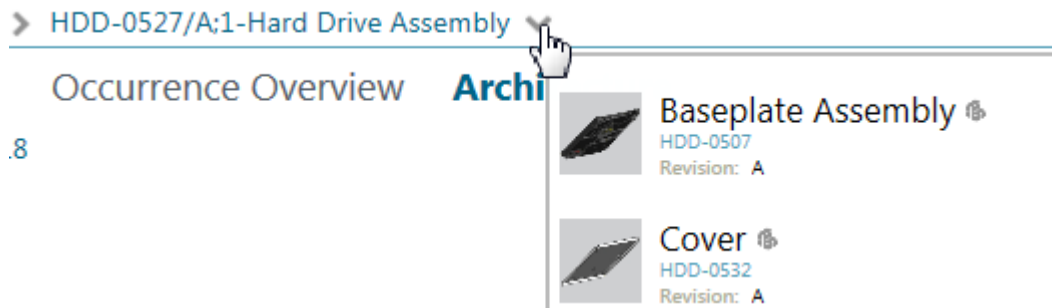
- Use the breadcrumb to navigate the diagram:
  - Click a breadcrumb to update the list of model elements.
  - Click a breadcrumb to select an element in the model if it is displayed.
  - Click a name in the list of structure elements to change the breadcrumb leading up to the element.

The breadcrumb also changes when you select an element in the diagram.

- To select the child structure from the breadcrumb:
  1. To select the child structure from the breadcrumb, click **>**.

The arrow changes to  $\surd$  and lists the children.

- Click the child you want to select.



- Double-click in the background to resize the network to fit all content in the display.
- To drag the diagram in the display area to see parts of it that may be hidden, click and hold in an unselected element first. To move an individual element, select the element first and then click and hold to drag it.
- Use the scroll wheel on the mouse to zoom in and out of the diagram.
- Click  $\square$  in the lower right corner of the pane to view the overview map of the network.



Move the rectangle in the overview map to browse the diagram.

- Hover over an element in the diagram to display additional details, such as the last name.
- Hover over a relation (the line connecting two objects) to display the type of relationship between the elements.
- You can specify that an element in the diagram is a root element by setting it up as an anchor node. Click **Set Anchor**  $\square$  to set or remove an element as an anchor node.

If you hide an element that is an anchor node, all its associated children will also be hidden.

## Modify your view of models using options in the Architecture tab

### Show, hide and filter model elements in the diagram

While viewing model diagrams in the **Architecture** tab, you can navigate the hierarchy of models. As you navigate, you can use options to hide and show (expand and collapse) child and parent elements as well as hide and show trace links and connections.

1. To show or hide model elements click the **Visibility Control**  and choose from following options:

a. If no model elements are selected, choose from the following options:

Command	Description
<b>Fit</b>	Fits all the model elements in the diagram.
<b>All Off</b>	Hides all the model elements in the diagram.

b. If model elements are selected in the diagram, choose from the following options:

Command	Description
<b>Fit</b>	Fits all the model elements in the diagram.
<b>Fit Selected</b>	Fits only the selected model elements in the diagram.
<b>All Off</b>	Hides all the model elements in the diagram.
<b>Selected Only</b>	Shows only the selected model elements in the diagram.
<b>Selected Off</b>	Hides the selected model elements in the diagram.

2. You can also show and hide model elements by selecting model elements from the Content list on the left hand side and using the **Visibility Control** .







3. Use the following option to filter relations:


a. Click **Relation Filters** .

By default, connectivity, traceability, and occurrence traceability are supported.

You can select multiple filters and the applied filters appear in the diagram.

The **Relations Legend** panel also maps to the filters applied.

- b. Click **Relation Controls** .
4. Click the arrows inside the relations box to expand or collapse the incoming and outgoing relations:
    -  Click to show all the incoming relations.
    -  Click to show all the outgoing relations.
    -  Click to hide the incoming relations for the object.
    -  Click to hide the outgoing relations for the object.
    -  Click to hide all the incoming relations or show all the incoming relations.

This button appears when incoming relations are partially shown in the diagram. When you click this button, a list appears, asking whether you want to hide the visible incoming relations or show all incoming relations.
    -  Click to hide all the outgoing relations or show all the outgoing relations.

This button appears when outgoing relations are partially shown in the diagram. When you click this button, a list appears, asking whether you want to hide the visible outgoing relations or show all outgoing relations.
- 

These commands only expand or collapse the connections and relations and not the objects in the diagram.

5. (Optional) To hide individual model elements, click the icon on the model element.

The model element fades and then turns off. You can stop the model element from disappearing by clicking the icon again.

Any children associated with the block are also hidden as well.

You can also hide and show model elements using **Relations Controls**.

### Use the content list to make elements visible or not visible

When you are navigating a model diagram using the content list on the left, you can control which elements appear in the **Architecture** tab.

1. Open the model or working context you want to view.

2. Click the **Architecture** tab.
3. In the content list on the left, at a single level, click ► next to an element to list the children in the list.
4. Drill down to lower levels by continuing to click ► next to each successive child.
5. Click the thumbnail for an element in the list to switch between displaying or not displaying it in the diagram

When an element is selected to display in the diagram, the thumbnail appears enabled next to it in the list. When the thumbnail is grayed out in the list, as shown in this example, the element is not shown in the diagram.







## Show and hide children

When you want to focus on viewing certain levels in a model diagram, you can show and hide children.

**Note:**

The hide command does not work when the children have an internal relationship between themselves.

1. Open the model or working context for the model you want to view.
2. Click **Architecture**.
3. In the diagram, click the arrow ► in the lower right corner of the element to show and hide (expand and collapse) its children. A number appears next to the arrow to indicate how many *direct* children the element has. The arrow appears in one of four ways:

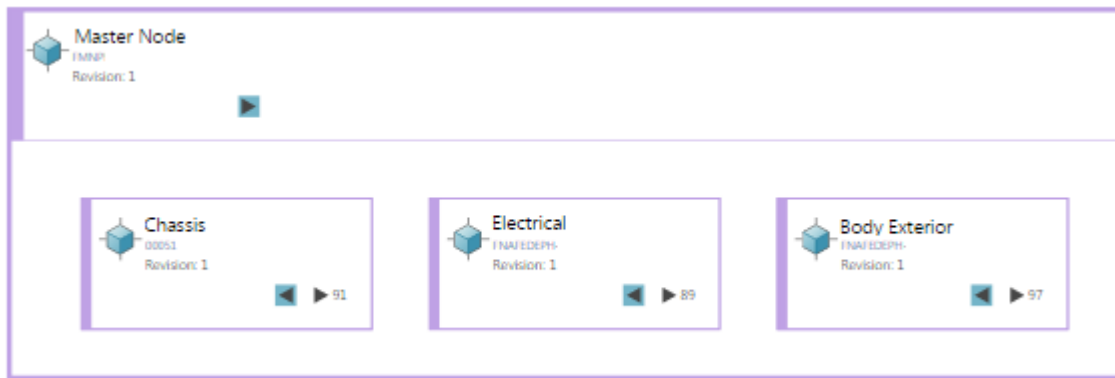
Example	Description
 <p data-bbox="321 268 553 386"><b>Master Node</b> FMNP Revision: 1</p>	<p data-bbox="716 243 1419 306">The number next to the arrow indicates the number of children.</p> <p data-bbox="716 331 1089 361">Click to show all the children.</p>
 <p data-bbox="321 512 524 617"><b>Master Node</b> FMNP Revision: 1</p>	<p data-bbox="716 495 1357 525">This arrow indicates all the children are expanded.</p> <p data-bbox="716 550 1214 579">Click this arrow to hide all the children.</p>
 <p data-bbox="321 764 527 869"><b>Master Node</b> FMNP Revision: 1</p>	<p data-bbox="716 743 1370 772">This arrow indicates that some children are hidden.</p> <p data-bbox="716 798 1451 861">Click this arrow to show all children. Right-click to hide all the children.</p>
 <p data-bbox="321 995 537 1100"><b>Master Node</b> FMNP Revision: 1</p>	<p data-bbox="716 978 1430 1041">The number next to the arrow indicates that all children are hidden.</p> <p data-bbox="716 1066 1175 1096">Click this arrow to show all children.</p>

4. You can hide child elements from the graph by clicking the icon of the element.

If the child element is an anchor node, the hide action makes the parent the anchor node.

If a child element that has connections is rolled up, the parent element displays the edge.

5. You can hide child elements from the content list by clicking the icon of the element that appears in the content list.
6. In the **Nested View**, when the parent is expanded, the children are contained within a box with the parent. You can expand the children in the same way to drill down to lower levels of the structure.

**Note:**

If one of the children is a root node, when you hide children, the parent becomes the root node.

## Show and hide parents

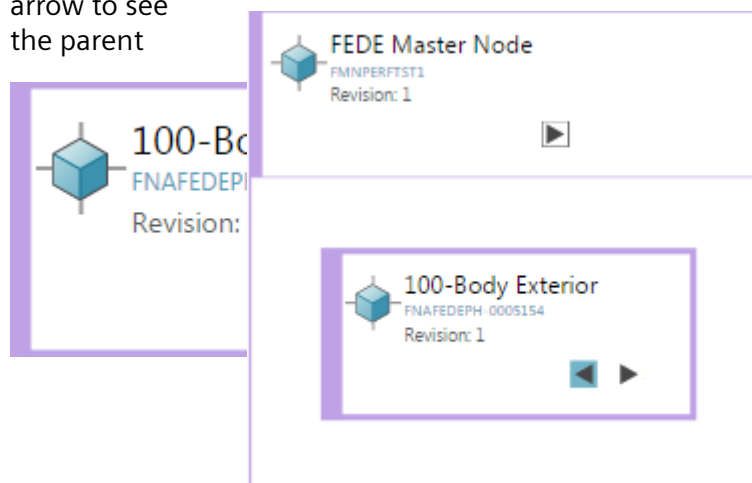
When you want to focus on viewing certain levels of a model, you can show and hide parents.


1. Open the model or working context for the model you want to view.
2. Click **Architecture**.
3. In the diagram, click the ◀ arrow in the lower right corner of a child to show its parent.

**Example:**

Click the ◀ arrow to see the parent

Parent visible



- To hide the parent, click the  icon in the child element.

If the parent is an anchor node, the child element that performed the hide action becomes the anchor node.

## Modify your view of trace links and connections

### Show and hide trace links and connections in the Architecture tab

You can hide trace links and connections so that you can focus on other diagram elements.


When an element has trace links or connections, an indicator appears on the diagram.

- Click the indicator to show and hide its trace links and connections. A number appears next to the arrow to indicate how many trace links and connections it has. The indicator appears in one of four ways:


 The outgoing relations are hidden. Click this to show the outgoing relations.

 The outgoing relations are shown. Click this to hide the outgoing relations.

 The incoming relations are hidden. Click this to show the incoming relations.

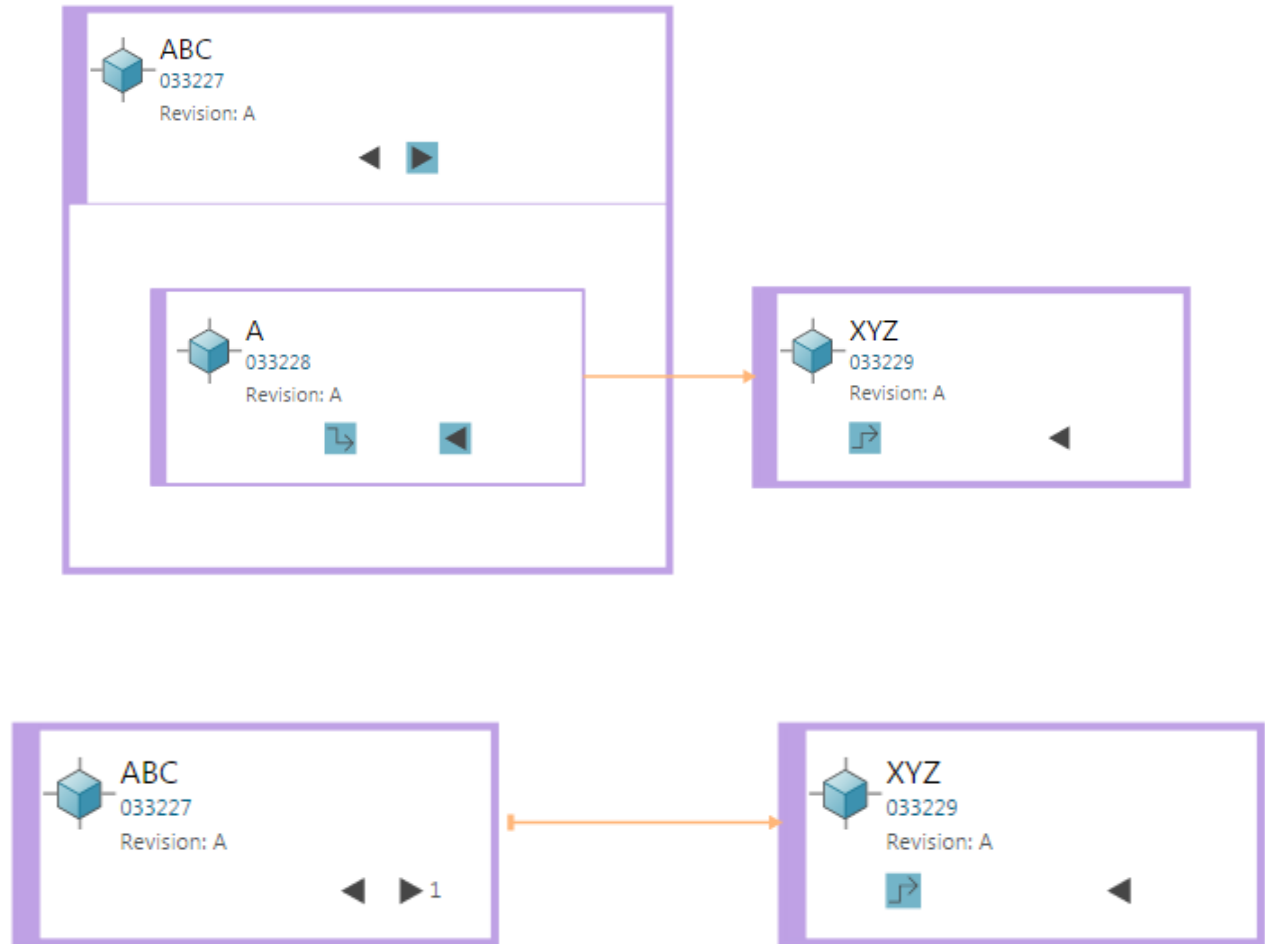
 The incoming relations are shown. Click this to hide the incoming relations.



- When you collapse a parent element and the child elements have trace links with elements outside the parent, these trace links end with T-lines () instead of straight lines.

Note:

This feature is activated by your application administrator.



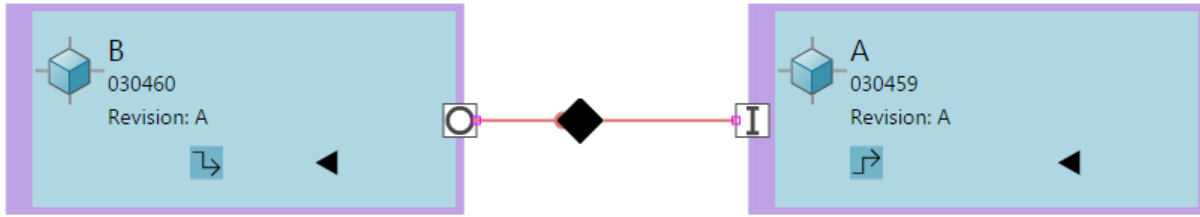
You can **configure** if you want these T-lines to appear.

An administrator can configure if you want these T-lines to appear.

- A black diamond displays if there are trace links or connections to a design.


Note:

This feature is activated by your application administrator.



### Show or hide trace links and connections in the Relations panel


To focus on viewing certain elements of a model, you can show and hide trace links and connections using the **Relations** panel.

- Select an element in the work area and click  to open the **Relations** panel.

The trace links and connections are listed in separate tabs.

- You can filter the content that appears in the **Relations** panel based on name of the element, name of the connection, trace link, or direction.
- Click the thumbnail next to the element to display or hide the trace link or connection in the work area.

A grayed out thumbnail indicates that the element is hidden in the work area.

- Click  to select all the available entries in the **Relations** panel.

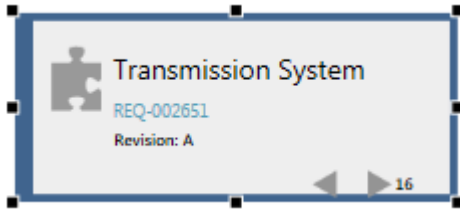
The *select all* functionality is only available in the **Trace Links** and **Connections** tab.


### Resize a box

To optimize the fit of boxes and their contents in a diagram, you can resize a box.

1. Open the diagram and click **Architecture > Start Authoring** .
2. Select the box you want to resize.

The box appears with sizing handles.



3. Drag any of the sizing handles away from or toward the center of the box until it is the size you want.
4. When you finish modifying the diagram, click **End Authoring** .

## Edit properties and names

To update the information stored with an element, you can edit its properties and its name as they appear in a diagram. Connections, blocks, and relations are sorted and highlighted based on these property values.



### Edit properties

1. Select the relevant element, and click **Information** .

The properties appear in the **Information** panel.

2. Click **Edit**.
3. Change the properties you want to edit.
4. Click **Save**.

### Edit name

1. Open the diagram and click **Architecture**.
2. Click **Start Authoring** .
3. Click the element name in the diagram to highlight it.
4. In the editing box, type the name as you want it to appear.
5. Press **Enter** or click outside the editing box.
6. When you finish modifying the diagram, click **End Authoring** .

If you edit the occurrence name, the occurrence name is displayed in the element.

## Modify models using options in the Architecture tab


### Add, copy, and paste elements in a model diagram

Working with a model diagram in the **Architecture** panel, you can modify requirement specifications and block structures, including adding and removing elements.

You can:

- Add an element as a child or sibling of an element selected in the model.
- Paste an element copied from within the context you are working in or from elsewhere.
- Paste the same element from the clipboard multiple times to add it in multiple places.
- Paste a copied element to an external application that supports formatted URL linking, such as Microsoft Word. The URL link opens the **Show Object** page for the element. You can copy and paste multiple elements to paste multiple URLs. Linked URLs are also generated in reports.

### Add a new element to a model diagram


1. Open the diagram, click **Architecture** panel, and click a spot in the diagram where you want to create the element.
2. Click **Add Element** .
3. If the option to select either **Child** or **Sibling** appears, select where you want to add the element in the diagram relative to the selected element.
4. Select the type of element you want to create either from the list of recently used types or from the full list of available types. You can also search for a type using the **Filter** box.
5. Enter the properties for the new element.

The required properties vary depending on the type of element being created. These properties are configured by stylesheets, which are maintained by the administrator. The required properties are denoted with a red asterisk (\*).

6. Click **Create and Add**.

You can also add elements to the model diagram as follows:


1. Open the diagram and click **Architecture > Start Authoring** .
2. Click  to display the **Relations Legend** panel.

3. From the objects list, click the object you want to add to the diagram. Double-click for sticky mode. In sticky mode, you can add multiple elements to the diagram.
4. Click a spot in the diagram to add the model element.
5. After adding the model elements, click **End Authoring** .


Note:

If the authoring mode is on, and you leave the **Architecture** tab or launch a new panel, the authoring mode is turned off.

## Remove an element from a diagram

1. Open the diagram, click **Architecture**, and select the element in the diagram that you want to remove.
2. Click **Edit Diagram > Remove** .

## Paste an element from the clipboard

1. With the model diagram open in the **Architecture** panel, click in the diagram where you want to create the element.
2. Click **Add Element** .
3. If the option to select either **Child** or **Sibling** appears, select where you want to add the element in the diagram relative to the selected element.
4. Select the type of element you want to create either from the list of recently used types or from the full list of available types. You can also search for a type using the **Filter** box.
5. Enter the properties for the new element.

The required properties vary depending on the type of element being created. These properties are configured by style sheets, which are maintained by the administrator. Required properties are designated with a red asterisk.

6. Click **Create and Add**.

## Copy an element

1. With the model diagram open in the **Architecture** tab, click in the diagram where you want to create the element.
2. Click **Add Element** .

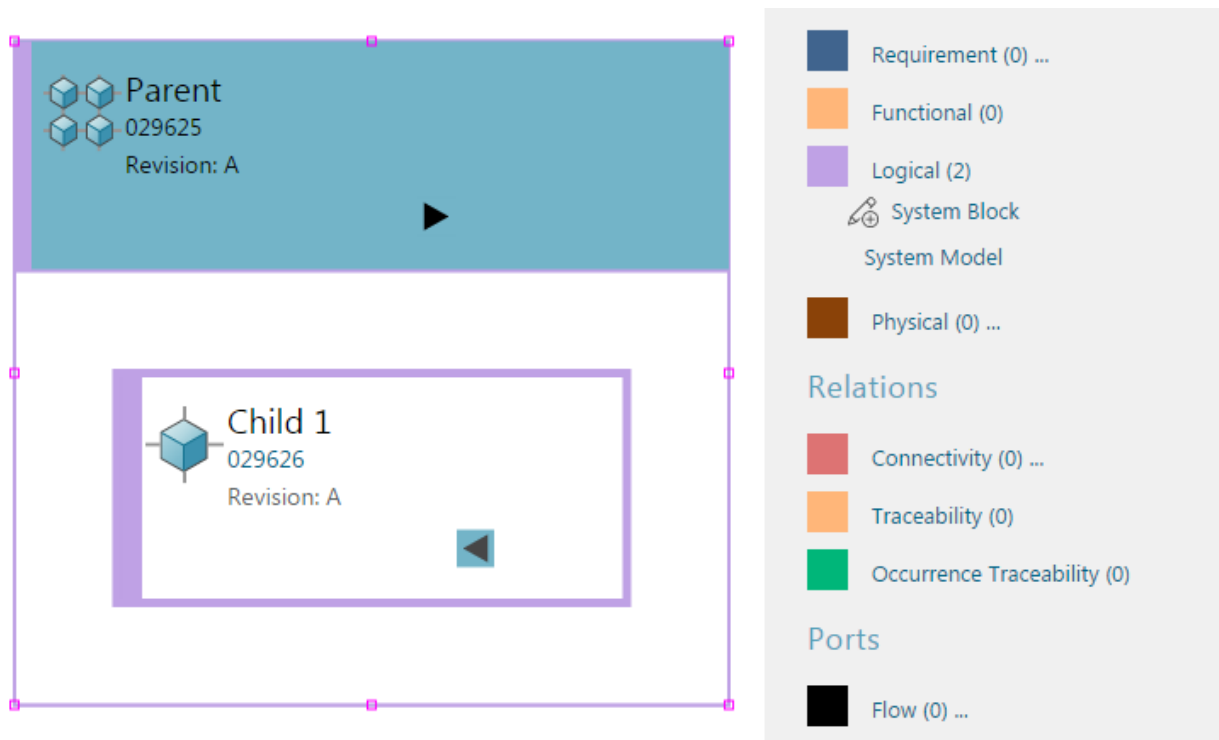
3. If the option to select either **Child** or **Sibling** appears, select where you want to add the element in the diagram relative to the selected element.
4. Select the type of element you want to create either from the list of recently used types or from the full list of available types. You can also search for a type using the **Filter** box.
5. Enter the properties for the new element.

The required properties vary depending on the type of element being created. These properties are configured by style sheets, which are maintained by the application administrator. Required properties are designated with a red asterisk.



6. Click **Create and Add**.

### Convert an element to a parent element

You cannot add child elements to an existing child element using **Relations Legend**.



The **Convert to Parent** command creates the space to add child elements. After using this command you can create child elements using Relations Legend. To convert an element to a parent element:


1. With the diagram open in the **Architecture** tab, click **Start Authoring** .
2. Select the element that you want to convert to a parent element and click **Convert to Parent** .

You can now add child elements to the parent element you created.

## View and modify models using Relations Controls

### Modify your view of models using Relations Controls

In the diagram, you can modify the model by adding and removing related elements and trace links and connections. You can do this by working with the model in the **Architecture tab**, or you can use Relations Controls. Relation Controls is a tool that lists elements and relation types, from which you can show and hide the elements you want to view in the model.

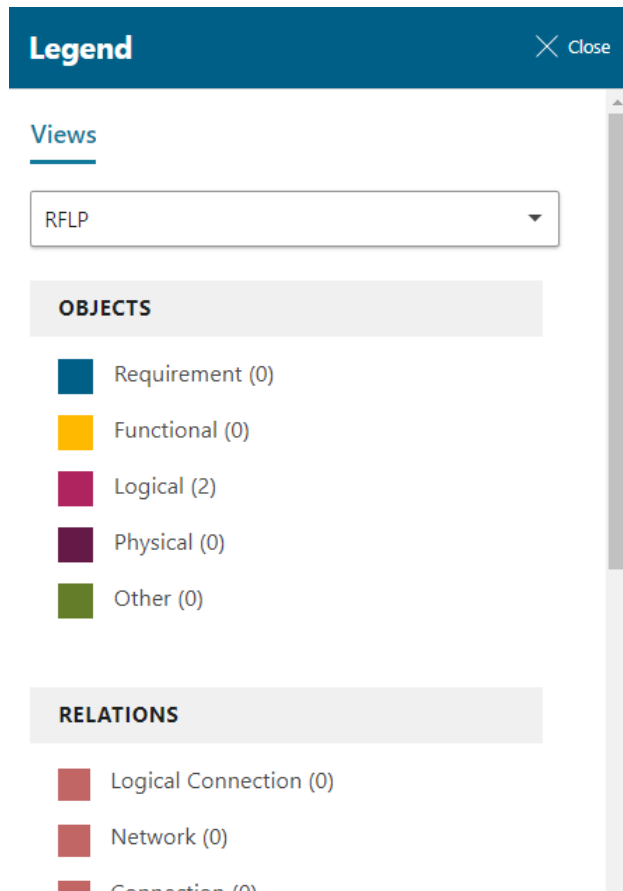
- You can view the types of elements and relations that exist in the context you are viewing. The objects and relations are color-coded. To see the legend describing the colors, click **Legend** . The **Legend** shows you the number of each of the relation types for the elements and the number for each element type currently displayed in the model. For example, the model represented by the **Legend** that follows has four functional blocks, one system block, and one physical model related to the top object displayed in the model. There are also relations (child/parent) and trace links, one connection and two ports. (Connections may exist only on system and functional blocks.)

Select a methodology from the drop-down, such as RFLP or ARCADIA. The **Legend** box colors update based on the methodology that you select.

Click the colored box for an element or relation type to switch the display of elements or relations of that type on and off in the architecture model.

**Note:**

Once you turn on the display of child nodes, these nodes become root nodes, and you cannot turn off the display again.



- When **Structure** relations are enabled in the **Legend**, indicators appear to help you **show and hide child and parent elements**.





### Add and remove related elements in a model

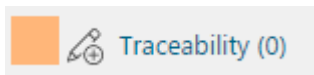
Using the model diagram, you can add related elements in hierarchies of requirements or functional, system, and physical models. Relations can be created between two elements only if the relation type is

valid for the element and the relation rules defined. For example, you may only add a functional block to another functional block; you cannot add a requirement to the functional block. However, if you are in a working context, you may add different types of elements to the working context, even though they may not be directly related to one another.

## Add a related element

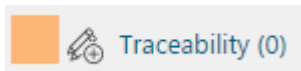
1. With the diagram open in the **Architecture** tab, open the element to be the parent of the related element you want to add.
2. Click **Start Authoring** .
3. Click **Legend** .
4. From the **Objects** list, click the name for the type of element or relation you want to add.

The edit mode icon appears next to the type.

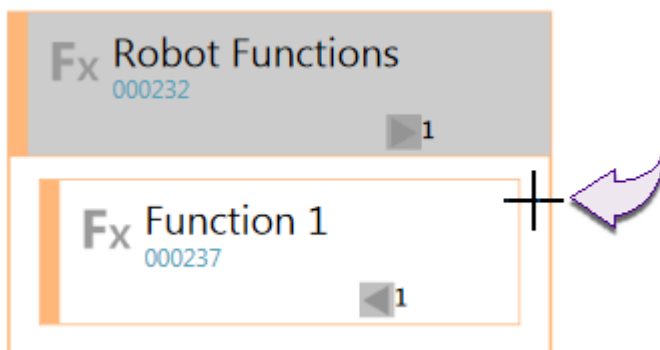


5. (Optional) To add multiple elements, click the type name again.

The multiple element edit mode icon appears next to the type.




6. Click in the blank area of the parent container.





When you release the mouse button, the element is created.

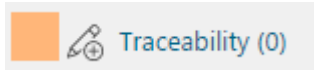
7. If properties are required for the element, the **Create** panel appears. Enter the information, and click **Create**.

8. Do one of the following to add another related element:
  - In multiple element edit mode, click again in the parent container.
  - In single element edit mode, click the object type in the **Legend**, and then click in the parent container.
9. When you finish modifying the diagram, click **End Authoring** .

### Remove a related element

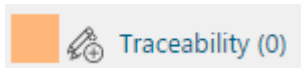
1. With the diagram open in the **Architecture** tab, open the element to be the parent of the related element you want to add.
2. Click **Start Authoring** .
3. Click  to display the **Relations Legend**.
4. From the **Objects** list, click the name for the type of element or relation you want to add.

The edit mode icon appears next to the type.

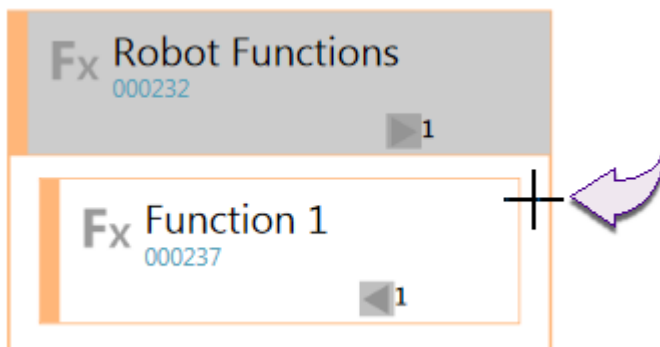


5. (Optional) To add multiple elements, click the type name again.


The multiple element edit mode icon appears next to the type.



6. Click in the blank area of the parent container.



When you release the mouse button, the element is created.

7. If properties are required for the element, the **Create** panel appears. Enter the information, and click **Create**.
8. Do one of the following to add another related element:
  - In multiple element edit mode, click again in the parent container.
  - In single element edit mode, click the object type in the **Relations Legend**, and then click in the parent container.
9. When you finish modifying the diagram, click **End Authoring** .

## Saving a view of an architecture model

When you modify a model and set the view to display certain elements, relations, connections, and ports, you can save it as a working context, so that you and others can recall it later with the same elements and settings.

The state of the model in the working context always appears as it did when you saved it, even if you make changes to the model later. If you change the model or the view of the model and you want to save the new view, you must create a new working context for the view. This includes changes such as adding and removing elements in the structure.

You can use saved models in workflows, change packages, and other services available to primary elements.

When you save a view of a model as a working context, it saves:

- Selected elements
- Positions of all elements
- Which elements are shown and hidden
- Which relations are shown or hidden
- Positions and display of ports and connections
- Type of display: network or nesting
- Zoom level
- Center point of the model

## Working with ports and connections

### How you use ports and connections

When you work with models in Active Workspace, you can build hierarchical structures or blocks that represent functional and system architectures. A *functional model* illustrates a function set that the product delivers, and a *system model* illustrates a proposed solution for implementing the requirements and functions. You use *connections* in models to show how the functional and system blocks are integrated with one another.

A *connection* joins functional or system blocks. *Ports* are the points where inputs are consumed or outputs are generated. When you add functional and system connections to an architecture model, ports are automatically added. You can also manually add a connection between ports or between a port and a block that does not have a port.



You can add a connection only if it is valid for the blocks you have selected. For example, you can add connections only between functional or system blocks.

You add, move, and remove connections on a model when you want to change how functional or system blocks are integrated in the model.

### Adding, moving, and removing connections

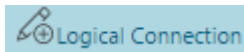
#### Add, move, remove, and view connections

When you add a connection between functional or system blocks, bidirectional ports are automatically added to the blocks. You can also add a connection between two existing ports that are not yet connected. Ports may have multiple connections. You can add a connection only if it is valid for the blocks you have selected.

1. Open the model containing the blocks you want to add a connection to and click the **Architecture** tab.
2. Click **Start Authoring** .
3. Click  to display the **Legend** panel.
4. From the **RELATIONS** list, click the **Connectivity**.
5. Under **Connectivity**, select the type of connection you want to add:
  - **Logical Connection** adds a connection between two system blocks.
  - **Network** adds a connection between two functional blocks.

- **Connection** adds a connection for other types of block.

The edit mode icon appears.




6. (Optional) To add multiple connections, click **Connectivity** again.

The multiple item edit mode icon appears.



7. Click the starting block or port, and hold the mouse button. Hover over the ending block or port, and then release the mouse button to create the connection.

If either block does not have a port, a new port is created when the connection is created. The connection type appears as the connection name, and **P** appears as the name of the ports if they are newly added.


8. (Optional) **Edit the name of the connection and ports.**
9. When you finish modifying the model, click **End Authoring** .

Connections are displayed at the bottom of the structure in indexed and nonindexed states.

When you turn off the connections, all related and orphan tiles are also turned off. Only anchor nodes are not turned off.

## Move a connection

You can move a connection from one port to another.

1. With the model open in the **Architecture** tab, click **Start Authoring** .
2. Click the connection to select it.

Tip:


You can also select it from the content list on the left.

3. Click the port from which you want to move the connection, and drag the end of the connection to the port where you want to end the connection.



When you release the mouse button, the connection is moved.


Note:

Any port with a disconnected connection is highlighted in red.

- When you finish modifying the model, click **End Authoring** .


### Remove a connection

- Open the model in the **Architecture** tab, and click **Start Authoring** .
- Click the connection you want to remove.
- Click **Remove Element** .

This action disconnects the connection. The connection is still available in the product structure. To remove the connection from the product structure, select the disconnected connection element from the list area and click **Remove Element** .

Note:

When you remove a connection the ports are not removed.

- (Optional) **Remove the remaining ports.**
- When you finish modifying the model, click **End Authoring** .

### Reconnect a connection

- Open the model in the **Architecture** tab, click **Start Authoring** .


Note:

Any port with a disconnected connection is highlighted in red.

- From the list area, click the connection you want to use. This connection must be a disconnected connection.

OR



Select a System block or port in the diagram.

- Click **Reconnect Connection** .
- Click the starting block or port, and hold the mouse button. Hover over the ending block or port, and then release the mouse button to reconnect the connection.

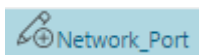
## Managing ports

### Add a port

When you add functional and system connections to a model, ports are automatically added. You can also manually add a port to a functional or system block. When you manually add a port, no direction is assigned to the port. The blocks to which you can add ports and the types of ports you can add depend on your configuration.

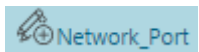
1. Open the model diagram in the **Architecture** tab, and display the block you want to add a port to.
2. Click **Start Authoring** .
3. Click  to display the **Relations Legend** panel.
4. From the **Ports** list, select **Flow**.
5. Under **Flow**, select the type of port you want to add:
  - **Network Port** for functional blocks
  - **Logical Port** for system blocks
  - **Connection Terminal** for other types of objects

The edit mode icon appears.

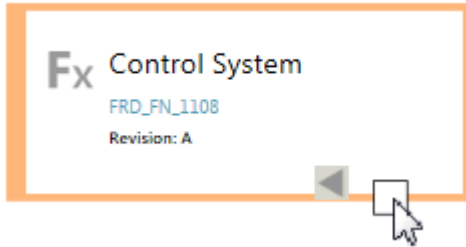


6. (Optional) To add multiple ports, click the port type again.

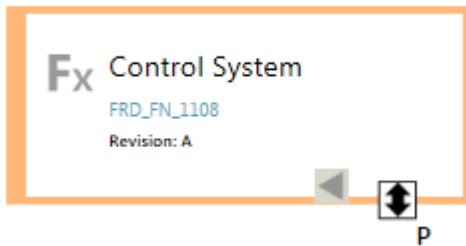
The multiple item edit mode icon appears.




7. Hover over the side of the block in the model where you want to add the port until you see the blank box indicator, and then click the side.







The port is created. **P** appears as the name of the port.



8. (Optional) **Edit the name of the port.**
9. When you finish modifying the model diagram, click **End Authoring** .

### Remove ports

1. Click **Start Authoring** .
2. Click the port that you want to remove.
3. Click **Remove Element** .
4. When the message appears prompting you to confirm the element removal, click **OK**.
5. If the port had a connection, the connection no longer appears, but is still listed in the content list on the left. Select the connection and click **Edit**  > **Remove** to remove it.
6. Click **End Authoring** .

### Change the direction of a port

Working in a model, you can change the direction of any port that has no connections.

You can also change the direction of a port, whether it has a connection or not, by **editing its properties**.

1. With the model diagram open in the **Architecture** tab, click **Start Authoring** .

2. Select the port.
3. Click **Information** ⓘ, and then click **Edit**.
4. Select a direction from the dropdown list menu, and then click **Save**.
5. When you finish modifying the model diagram, click **End Authoring** 🗑️.

## View all the ports in the model diagram

You can see only connected ports in the model diagram. The orphan ports are not seen.

To view all ports in the model diagram:

1. Click **Start Authoring** 🗑️.
2. Click **Show All Ports** 🗑️.

Click this command again to see just the connected ports.

## Editing port and connection properties

### Edit the port or connection properties in the information panel

To update the information stored with a port or connection, you can edit its properties and also its name as it appears in the information panel.

1. Select the port or connection and then click **Information** ⓘ.

The properties appear in the **Information** panel on the right.

2. Click **Edit**.
3. Change the properties you want to edit.
4. Click **Save**.

### Edit the port or connection properties in Table view

To update the information stored with a port or connection you can edit its properties and also its name as it appears in the Table view.



1. With the model diagram open in the **Architecture** tab, click **Start Authoring** 🗑️.
2. Select a port or connection row and click **Edit** ✎ > **Start Edit**.

The properties display in Table view.

3. Click **Edit**  > **Save Edits**.


### Edit the port or connection labels in the model diagram

You can edit the name and description of a connection and those of a port. You can also change the position of their labels.

1. Click **Start Authoring** .
2. If labels are not displayed, click **Toggle Labels**  to view the labels.
3. Click the name of the port or the connection in the model diagram.



Note:

To change the position of the port or connection label, select the port or connection label and move it to the desired location.


4. In the editing box, type the new name.
5. Either press Enter, or click outside the editing box.
6. To change the position of the label, select the port or connection label and move it to the desired location.
7. When you finish modifying the model diagram, click **End Authoring** .

### Reposition the ports or connections

To change the position of port and connection labels and position of ports:

1. With the model diagram open in the **Architecture** tab, click **Start Authoring** .
2. To display labels, click **Toggle Labels**  to turn the labels on.
3. To change the location of the port, select the element and move it.

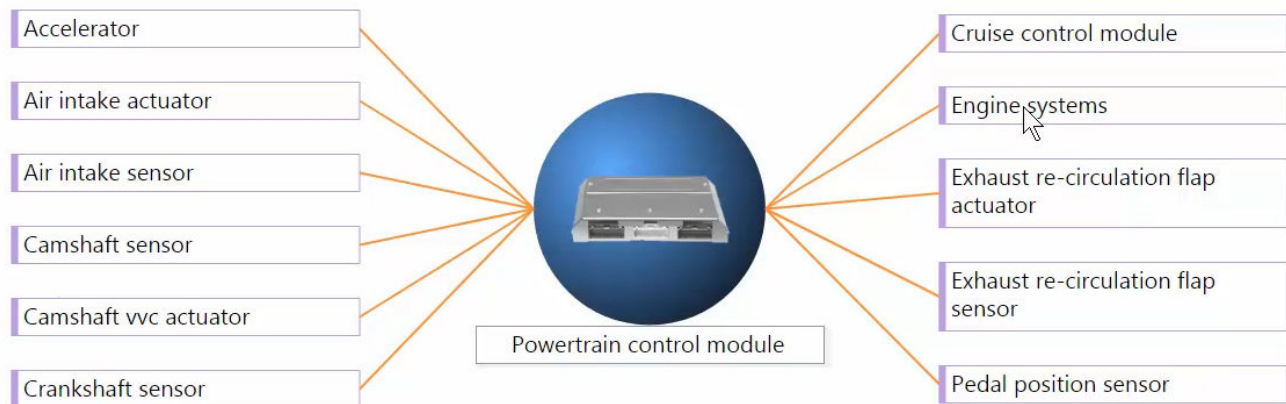
When you move the port, the connection is also automatically rerouted.

4. When you finish modifying the model diagram, click **End Authoring** .

## Managing interfaces

### Interface management with MBSE

Interface management involves the interoperability of systems through interfaces across multiple engineering domains. An interface is the system boundary where the system interacts with other systems. In systems engineering, our goal is to avoid or mitigate interface failures as early as possible in the product lifecycle. Interface management is about standardizing and controlling interfaces to achieve system success. By implementing an interface management solution in your organization you are better able to streamline communication, identify the critical interfaces, and monitor progress on them.

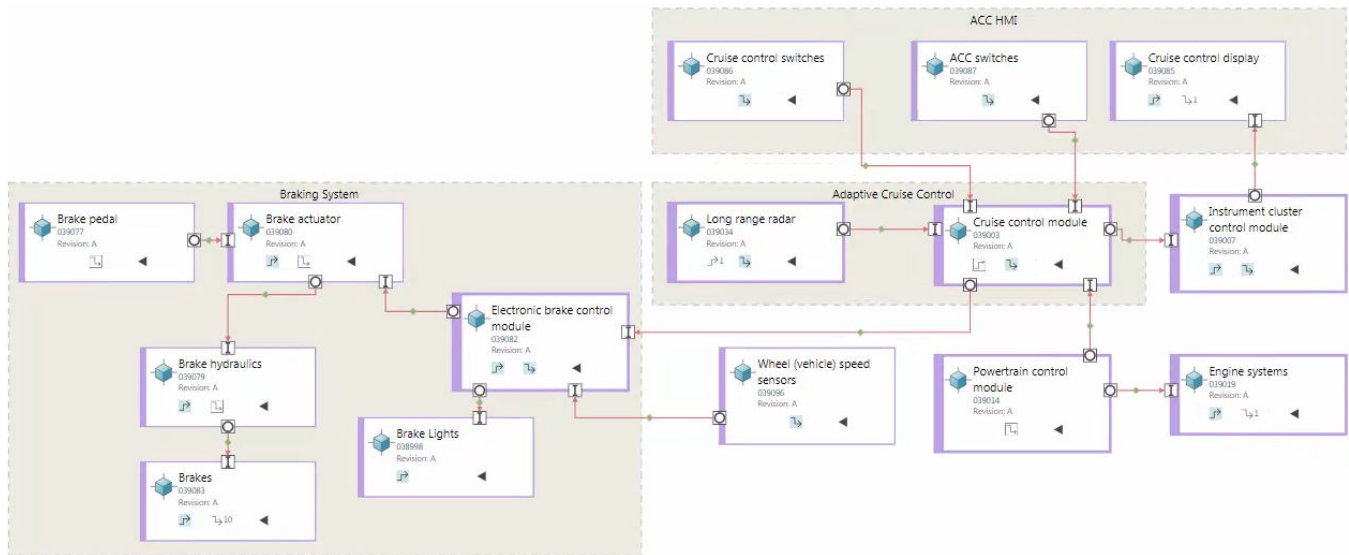


Active Workspace provides the tools to manage interfaces throughout the product lifecycle:

- Revise
- Change process management
- Review and comment

With Active Workspace, you can monitor the product development at differing levels with a roll-up of the critical interfaces, their status, and who is responsible for them. Software, mechanical, and electrical engineers can easily collaborate on interfaces in a co-development environment.

Active Workspace helps manage interfaces with visual definition of your systems along with their points of interfacing, and further elaborating the needs for success.



## View interfaces for a system

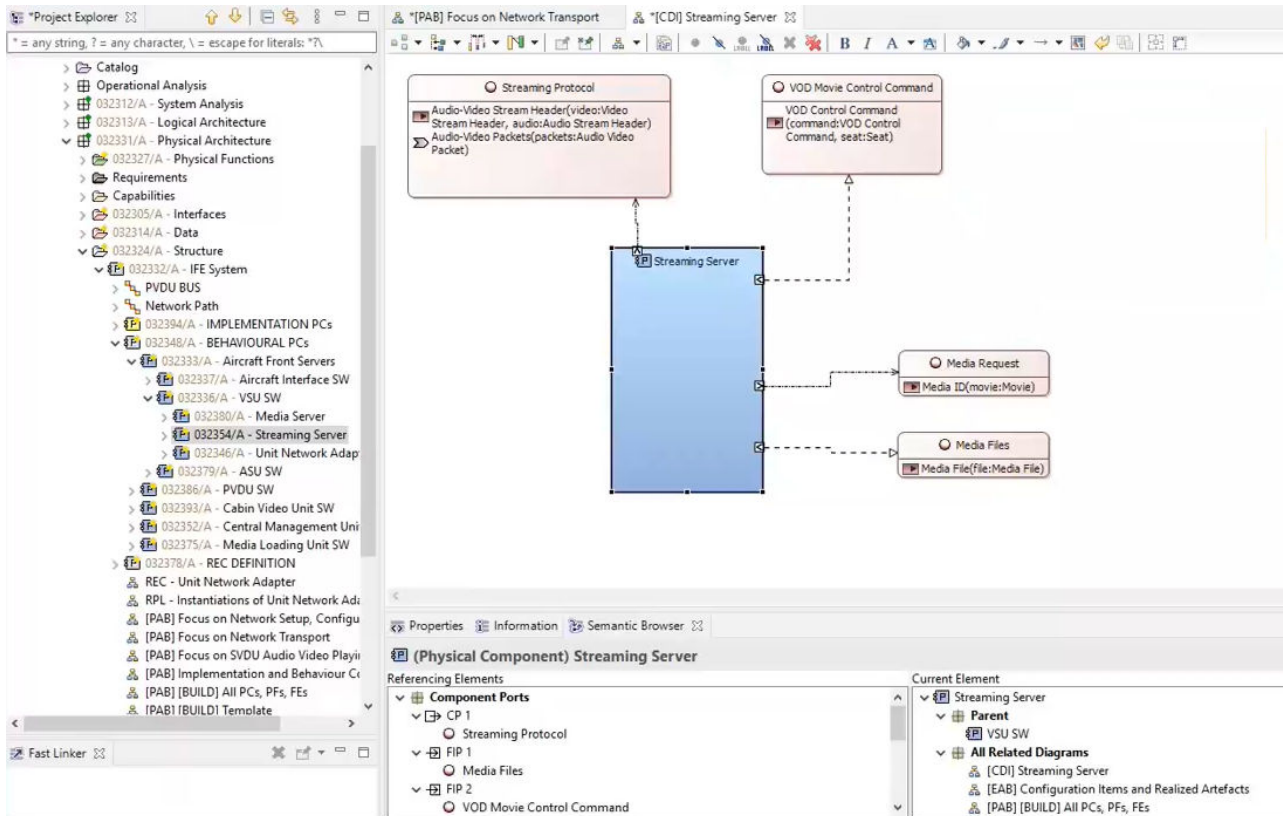
You can select a system and view its interfaces.

### Prerequisites

A system and interface must exist in the system modeling software such as System Modeling Workbench and the data must be published into Teamcenter.

### Procedure

1. Select an interface. The following graphic shows an example interface named Streaming Server that was modeled with System Modeling Workbench, which is integrated with Active Workspace.



## 2. Click the **Interfaces** tab.

This tab provides the following panels:

- **INTERFACES**

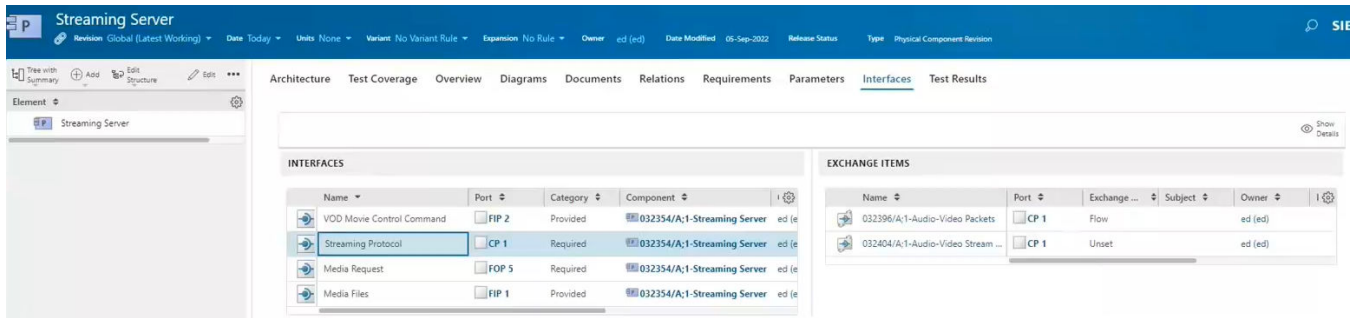
Provides a list of interfaces that interact with the selected interface, the ports of the selected interface, and the interfacing components through those ports.

- **EXCHANGE ITEMS**

Provides the parameters of the exchange items.

## Results

Active Workspace displays the selected interface. The following graphic shows the **Interfaces** tab for the **Streaming Server** interface.



## Display interfaces and related connections, exchanges, parameters, and requirements

You can select an interface and view the related interface data requirements and parameters that are linked to the interface.

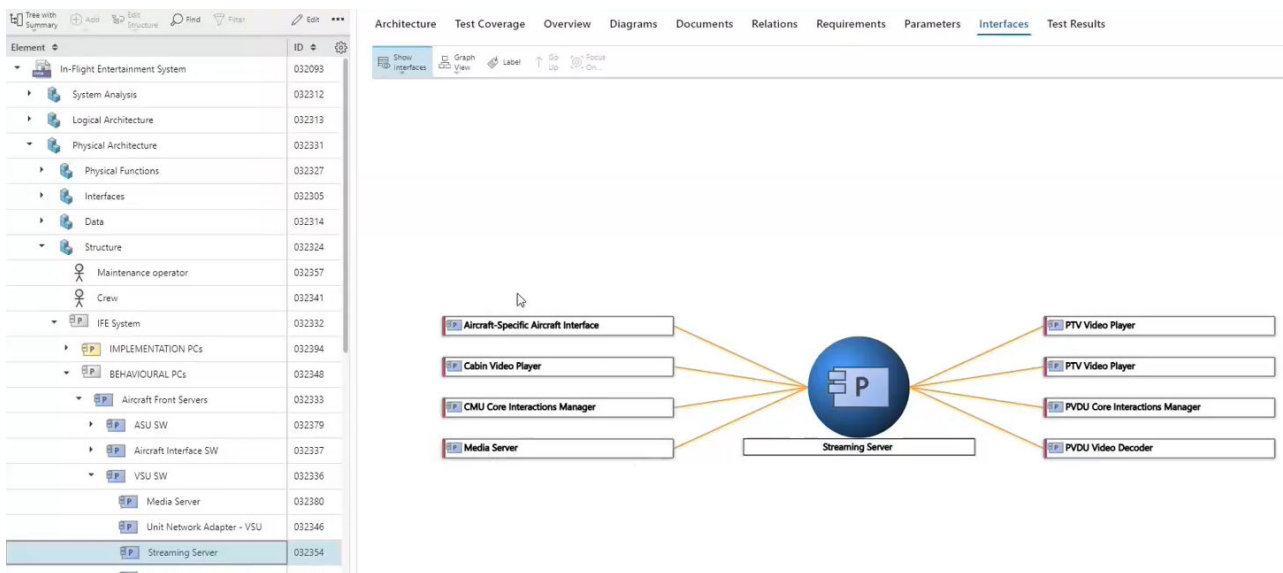
- External connections
- Exchanges
- Parameters
- Requirements

### Prerequisites

A system and interface must exist in the system modeling software such as System Modeling Workbench and the data must be published into Teamcenter.

### Procedure

1. Select an interface and then click **Show Interfaces**. The following graphic shows an example of viewing the **Streaming Server** interface.



The orange lines on each interface indicate an exchange.

- Click an interface to display the external connections, exchange items, functional exchanges, and interface alignment.
- In the **EXTERNAL CONNECTIONS** panel, select a connection to show its data.

The following graphic shows **Streaming Protocol** connection selected.

The screenshot displays the Teamcenter software interface. At the top, there are navigation tabs: Architecture, Test Coverage, Test Results, Overview, Diagrams, Documents, Relations, Requirements, Parameters, and Interfaces. The 'Interfaces' tab is active, showing 'Interfaces: Streaming Server - Video Player'. Below this, there are three panels: 'EXTERNAL CONNECTIONS', 'Functional Exchanges', and 'Exchange Items'. The 'EXTERNAL CONNECTIONS' panel shows a table with columns: Name, Owner, Aligned, Rev..., and Source. The 'Streaming Protocol' connection is selected. The 'Exchange Items' panel shows a table with columns: Name, Owner, Rev..., ID, Description, and Mechanism. Below these panels, there are tabs for 'Parameters' and 'Requirements'. The 'Parameters' tab is active, showing a table with columns: Name, Revision, Units, Measurement, Goal, Min, and Max. The table contains 12 rows of parameters.

Name	Revision	Units	Measurement	Goal	Min	Max
Parameter 1	A	Meters	[10,3]	[10,3]	[10,3]	[10,3]
Parameter 2	A	Meters	1.5	1.5	1	3
Parameter 3	A	Meters	18	19	17	20
Parameter 4	A	Meters	[10,3]	[10,3]	[10,3]	[10,3]
Parameter 5	A	Meters	1.5	1.5	1	3
Parameter 6	A	Meters	18	19	17	20
Parameter 7	A	Meters	[10,3]	[10,3]	[10,3]	[10,3]
Parameter 8	A	Meters	1.5	1.5	1	3
Parameter 9	A	Meters	18	19	17	20
Parameter 10	A	Meters	[10,3]	[10,3]	[10,3]	[10,3]
Parameter 11	A	Meters	1.5	1.5	1	3
Parameter 12	A	Meters	18	19	17	20

By default, **Show Details** is active, which displays the **Parameters** and **Requirements** tables.

## Results

Active Workspace displays the selected interface. The following graphic shows the **Interfaces** tab for the **Streaming Server** interface.

## Associating system model blocks with other objects using trace links

### View trace links

A *trace link* is a directional relationship between any two objects, including requirements and blocks in functional models, system models, or physical models.

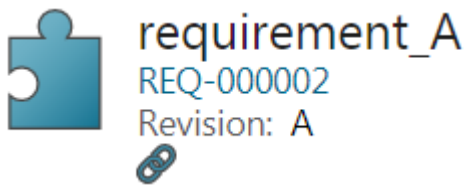
You use trace links for allocating requirements to elements in downstream or upstream structures, such as:

- Other requirements in the same structure or in different requirement views.
- Functional model blocks, systems model blocks, and physical models.

### View trace links associated with an object

1. Search for the requirement (or other object) or navigate to it in your **Home** folder.

In the content list, objects with trace links appear with the trace link (chain) symbol on the thumbnail image. The following example shows a requirement with an associated trace link.



2. Perform any of the following:
  - Hover over the trace link icon to see the number of objects that are linked.
  - Click the **Trace Link** tab, which lists all complying and defining objects associated with the selected object, regardless of where they exist. It also includes the relation type for each object.
  - Click the requirement to open the requirement in the **Documentation** tab.


### View trace links associated with an object in a working context

A working context provides a way to put your work aside for a while, then pick up where you left off when you return. Your working context can be retrieved from any device, allowing you to (for example) work on a desktop machine during office hours and then review your work on a tablet in a meeting or on the shop floor later in the day.

1. Search for the structure that contains the object or navigate to it in your **Home** folder
2. With the context open in the **Navigate** tab, click the **Architecture** tab.

The model shows a visual representation of trace links between objects in the structure.

3. Do one of the following:
  - Use the **Relations** panel to view trace links:

- Click  to open the **Relations** panel.

The **Trace Links** tab lists the model objects that have incoming and outgoing trace links. You can collapse and expand the **Incoming** and **Outgoing** lists.

The thumbnail next to each object indicates whether its trace link is displayed in the model. When the thumbnail is grayed out, the trace link is not displayed in the model. Click the thumbnail to switch between displaying and not displaying it in the model.

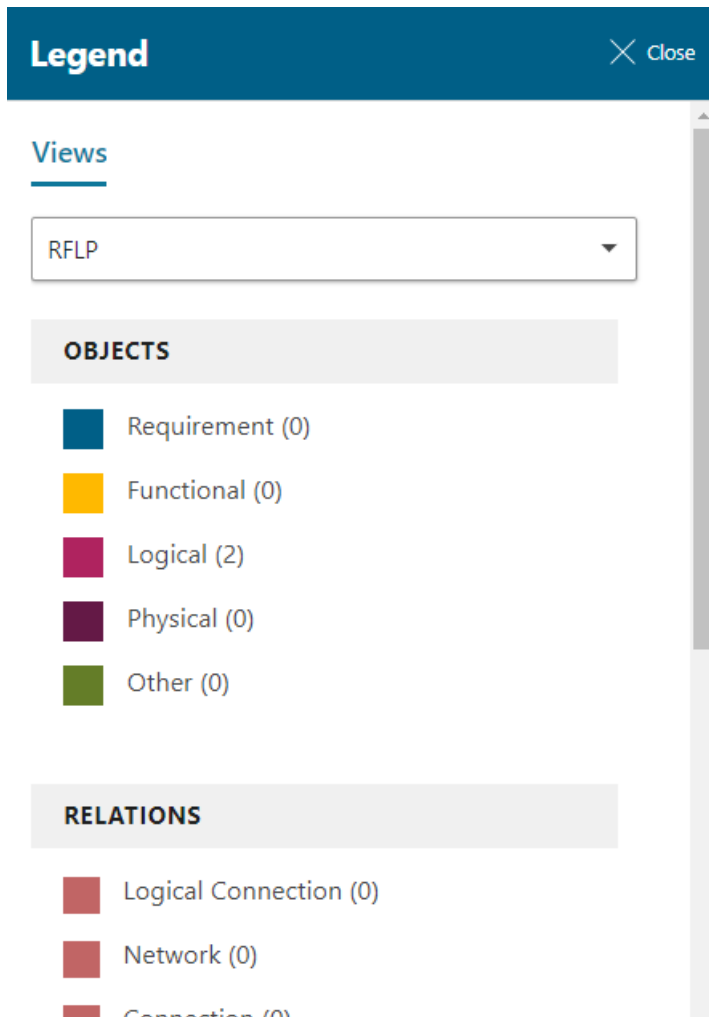
The **Relations** panel shows only the trace links present in the context. If an object has a trace link to another object not in this context, it is not listed.

- Use the Relations Controls tool to view trace links:

- Click **Relation Controls** .

The **Relations Legend** lists the relation types for the objects currently displayed in the model. Trace links are tallied next to **Traceability**. If an object has a trace link to another object not in this structure, it is not listed.

Click the colored box next to **Traceability** to switch the display of all trace links on and off in the model.



## Create a trace link from a system model block

Create a trace link when you want to define a directional defining-complying relationship between objects. For example, you can create a trace link between a requirement and a system model block.

1. Select the **Table** view.
2. Select one or more object revisions to which you want to create a trace link and set as the defining object. Press **Ctrl** or **Shift** to select multiple objects.

These objects represent your **Start** (defining) objects in the trace link relationship.

3. Click **New** ✨ > **Trace Link**.

Note:

You can use the keyboard shortcut Alt + L to create a trace link.






The **Trace Link** panel opens and the selected defining objects appear in the **Start** section.

- Choose the object you want to link to as the **End** (complying) objects by doing one of the following.

Note:

You can create trace links while working among multiple browser windows.

You can use the clipboard to add either multiple **Start** objects or **End** objects. However, you cannot create a trace link with both multiple **Start** objects and multiple **End** objects.

- To add all clipboard objects to **Start** or **End** (in the **New** tab), click **Paste**  to the right of **Start** or **End**.
- To select an object from multiple objects on the clipboard, click **Add** . In the **Palette** tab, and then select an object from the **Clipboard** section.
- To select a favorite or recently viewed object, click **Add** . Click the **Palette** tab and then select the object from the **Favorites** or **Recent** section.
- To search for an existing object, click **Add** . Click the **Search** tab, type search terms to find the object you want to link to, and click **Search** . Select the object from the search results.

- Click **Add**.

The select objects are added to the **End** section.

- If the **Type** list is available, choose the trace link type you are creating. If you select a trace link type that has required properties, the required properties appear directly after the **Type** list. You must enter required properties to create a trace link.
- Click **Create**.

The trace link is created.

- To view the trace link, open the defining object, and click the **Trace Links** tab. The complying object appears in the **Complying Objects** list or table.

## Modify the trace links in a model diagram



### Add a trace link

Using the model diagram, you can add trace links between elements in hierarchies of requirements or functional, system, and physical models.

A trace link is a directional relationship between requirement specifications, functional models, system models, or physical models. In the relationship, one element is the defining element and one is the complying element. The defining element specifies a condition that a product or a component must fulfill. The complying element must partially or completely fulfill the condition specified by a defining element.

The **Tracelink\_Edit\_enabled** preference that is used to create and edit trace links in the Teamcenter rich client is ignored by the System Modeling functionality in Active Workspace.

You can only add a trace link if it is valid for the elements you have selected.

1. With the diagram open in the **Architecture** tab, click **Start Authoring** .
2. Click  to display the **Relations Legend**.
3. From the **Relations** list, click **Traceability**.

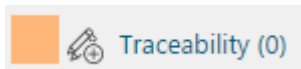
The edit mode icon appears.



To link occurrences, click **Occurrence Traceability**.

4. (Optional) To add multiple trace links, click **Traceability** again.

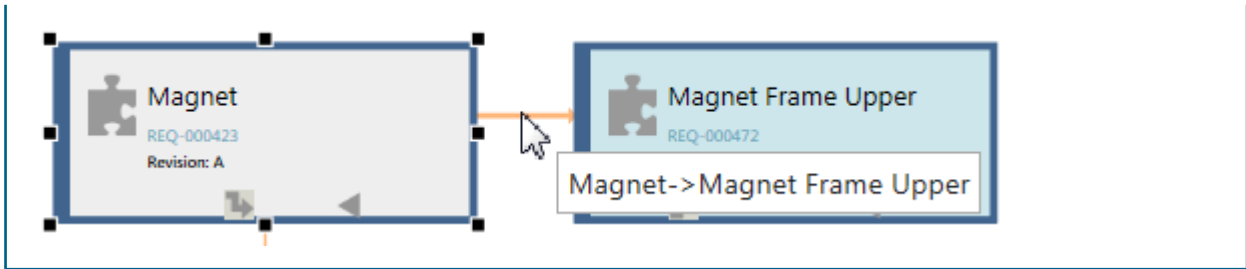
The multiple element edit mode icon appears.




5. Click the starting element and hold the mouse button. Hover over the element you want to create the trace link to. When you hover over the target, the element is highlighted and a red arrow appears. Release the mouse button to create the trace link.
6. If properties are required for the trace link, the **Trace Link** panel appears. Enter the information, and click **Create**.

Example:

A trace link is added between two elements.

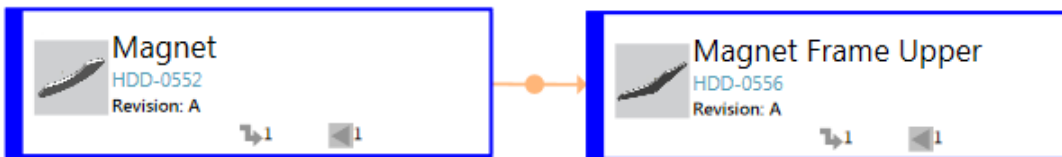



- When you finish modifying the diagram, click **End Authoring** .

## Delete a trace link

- Click the trace link, which is the line between the two elements.

The line appears with a circle on it.



- Click **Delete Trace Link** .
- Confirm the deletion.

## Managing embedded software projects

### What is Embedded Software Management?

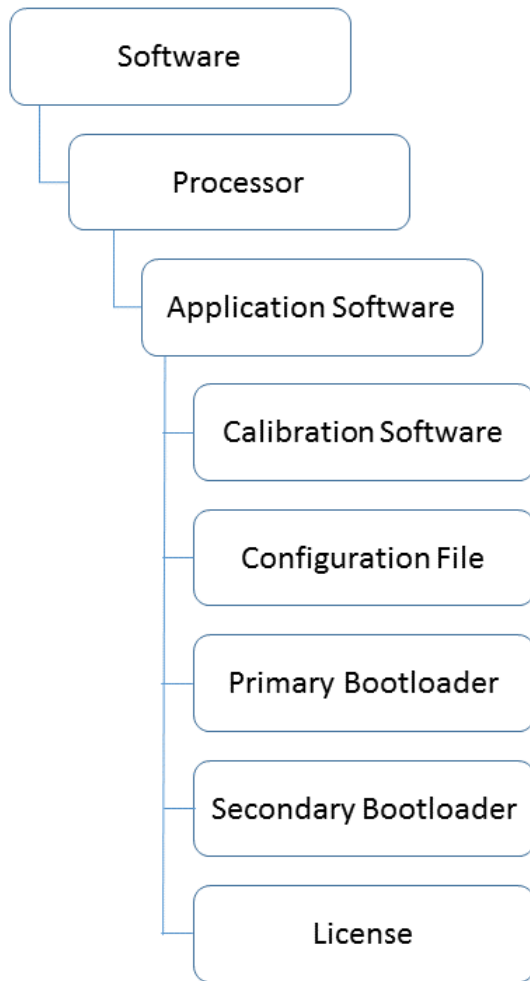
The Embedded Software Management functionality allows you to represent embedded software artifacts using Active Workspace. You can do the following:

- Represent software architecture

A software architecture is similar to a logical model. It represents the system definition of the embedded software.

- Represent embedded software components

You can create embedded software components that represent the actual software deliverables or work such as software release, processor, and calibration. A sample embedded software BOM is as follows:



The structure shown in the graphic is just a sample. You can configure your structure to include any object types.

- Link software components to external tools such as Polarion

In most cases, software tasks are tracked in ALM tools such as Polarion. You can link the Embedded Software components in Teamcenter to the corresponding artifacts in Polarion. This creates traceability across applications that allows you to track dependencies between the different components.

The links are created using Linked Data Framework. To create links, ensure that your administrator has setup Linked Data Framework.

- Link embedded software components with other components with Teamcenter by using tracelinks. This is to create traceability across different Teamcenter components.
- Capture the software deliverables such as code executables, and CFG files from external tools such as Polarion and associate them with Teamcenter components.

This functionality uses Linked Data Framework. To use this functionality, ensure that your administrator has set up Linked Data Framework.

## Representing embedded software in Teamcenter

### Create Software Architecture

1. Go to the folder in which you want to create the Software Architecture.
2. Click **New** ✨ > **Add**.
3. In the **Add** panel, select the **Software Architecture** type and enter the required properties.
4. Click **Add**.

### Create Software

1. Go to the folder in which you want to create the Software.
2. Click **New** ✨ > **Add**.
3. In the **Add** panel, select the **Software** type and enter the required properties.
4. Click **Add**.

### Create Processor

1. Go to the folder in which you want to create the Processor.
2. Click **New** ✨ > **Add**.
3. In the **Add** panel, select the **Processor** type and enter the required properties.
4. Click **Add**.

### Create Application Software

1. Go to the folder in which you want to create the Application Software.
2. Click **New** ✨ > **Add**.
3. In the **Add** panel, select the **Application Software** type and enter the required properties.
4. Click **Add**.

### Create Calibration Software

1. Go to the folder in which you want to create the Calibration.
2. Click **New** ✱ > **Add**.
3. In the **Add** panel, select the **Calibration** type and enter the required properties.
4. Click **Add**.

### Create Configuration File

1. Go to the folder in which you want to create the Configuration File.
2. Click **New** ✱ > **Add**.
3. In the **Add** panel, select the **Configuration File** type and enter the required properties.
4. Click **Add**.

### Create Primary Bootloader

1. Go to the folder in which you want to create the Primary Boot Loader.
2. Click **New** ✱ > **Add**.
3. In the **Add** panel, select the **Primary Boot Loader** type and enter the required properties.
4. Click **Add**.

### Create Secondary Bootloader

1. Go to the folder in which you want to create the Secondary Boot Loader.
2. Click **New** ✱ > **Add**.
3. In the **Add** panel, select the **Secondary Boot Loader** type and enter the required properties.
4. Click **Add**.

### Create License

1. Go to the folder in which you want to create the License.
2. Click **New** ✱ > **Add**.

3. In the **Add** panel, select the **License** type and enter the required properties.
4. Click **Add**.

## Linking embedded software components to artifacts in Polarion

### Components and artifacts that you can link by default

By default, Embedded Software Management links with the following artifacts in Polarion:

Teamcenter component	Polarion artifact
Software Architecture	Software Release
	Requirement Collection
	Requirement
Software	Software Release

### Link a Software Architecture component to an existing Polarion Software Release component

1. Open a Software Architecture component and click the **Attachments** tab.
2. In the **Remote Links** section, click **Create Remote Link** ⊕.
3. In the **Add panel**, link to a Polarion Software Release as follows:
  - a. Select a project from the **Project** list.
  - b. You can link to a Polarion resource in two ways:
    - Create a link with an existing Polarion resource.
    - Create a Polarion resource and then link with it.

To link with an existing Polarion resource, select the **Existing** button.
  - c. Click **Add** ⊕ next to the **Remote Reference** label.

Log on to Polarion if prompted.
4. From the **Resource** list in the **Select Resource** page, select **Select Software Release**.

5. Select the Software Release resource.

The resource is selected and you are directed to the **Add** panel.

6. Click **Add**.

The Software Architecture component is linked to the Polarion Software Release. You can view this link on the **Attachments** tab in the **Remote Links** section.

### Link a Software Architecture component to a Polarion Software Release component by creating the Polarion item

1. Open a Software Architecture component and click the **Attachments** tab.
2. In the **Remote Links** section, click **Create Remote Link** ⊕.
3. In the **Add panel**, link to a Polarion Software Release as follows:

- a. Select a project from the **Project** list.
- b. You can link to a Polarion resource in two ways:
  - Create a link with an existing Polarion resource.
  - Create a Polarion resource and then link with it.

To create a new Polarion resource, select the **New** button.

- c. Click **Add** ⊕ next to the **Remote Reference** label.

Log on to Polarion if prompted.

4. From the **Resource** list in the **Create Resource** page, select **Create Release**.
5. Add the required details and click **Create**.

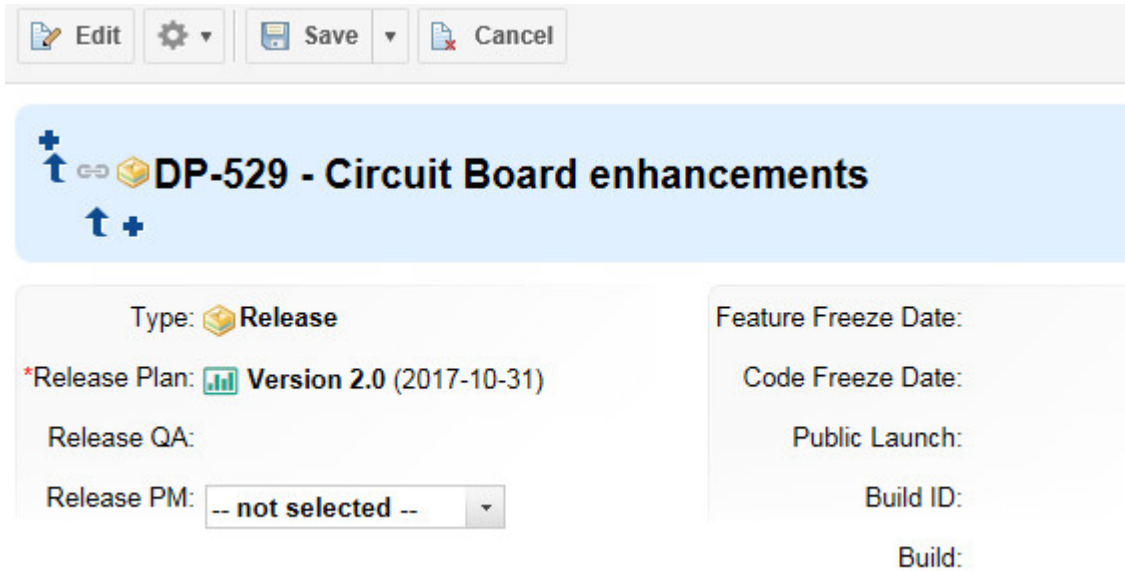
The resource is created and you are redirected to the **Add** panel.


6. Click **Add**.

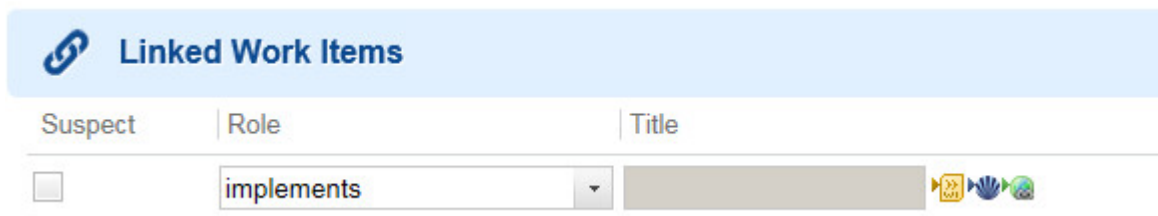
The Software Architecture component is linked to the Polarion Software Release. You can view this link on the **Attachments** tab in the **Remote Links** section.


## Link a Polarion Software Release to a Software Architecture component by selecting an existing Software Architecture component

1. In Polarion, navigate to a project, and select and open a Software Release.



2. In the Linked Work Items section, click **Edit** .



3. From the **Role** list, select a relation that your administrator has configured.
4. Click **Select Work Item From Linked Data Friend Server** .

The **Link External Item (Linked Data)** dialog box appears.

### Link External Item (Linked Data)

**Location:**

- In the **Link External Item (Linked Data)** dialog box, select the location where you want to create the link.

Log on to Active Workspace if prompted.

- In the **Link External Item (OSLC)** dialog box, you can create a link to a Teamcenter Software Architecture component in two ways:
  - By selecting an existing software architecture component
  - By creating a new software architecture component

### Link External Item (Linked Data)

Location:

TC12REG -> Embedded Software Management

Select an Existing Item...

Create New Item...

- To select an existing software architecture component, click **Select an Existing Item**.

An Active Workspace dialog box appears.



- In the **Select** dialog box, select a Software Architecture component.

After you select the component, the **OK** button appears.

Click **OK**.

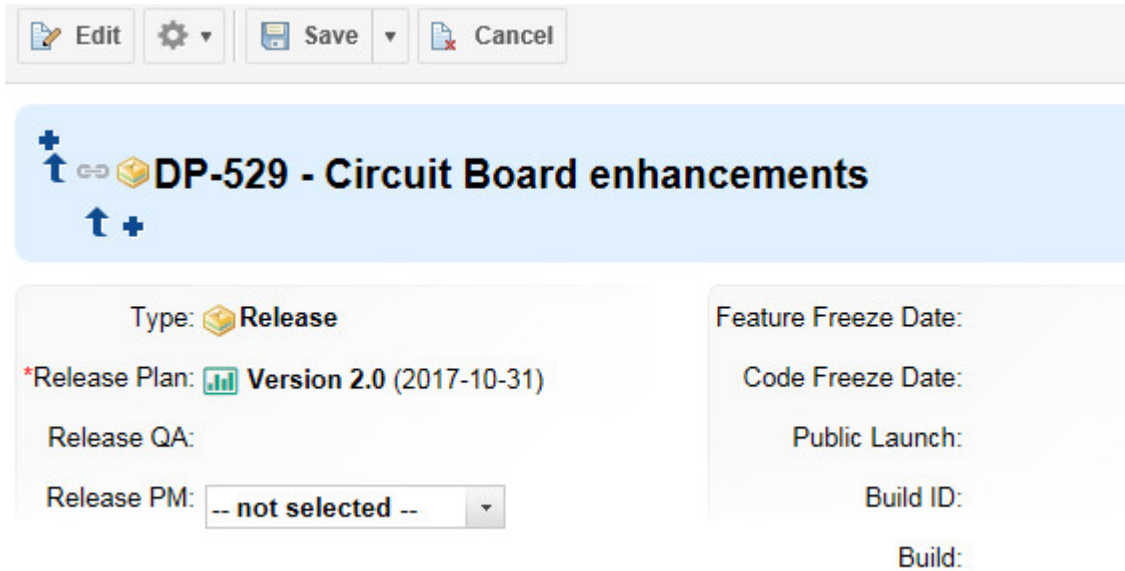
- Click **Save**.

The link is created and it appears in the **Linked Work Items** section.

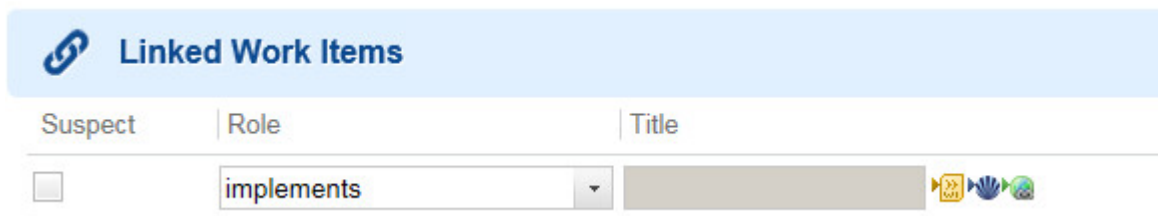
 <b>Linked Work Items</b>		
Suspect	Role	Title
<input type="checkbox"/>	implements	 Circuit Board 2020

## Link a Polarion Software Release to a Software Architecture component by creating a Software Architecture component

1. In Polarion, navigate to a project, and select and open a Software Release.



2. In the Linked Work Items section, click **Edit**.



3. From the **Role** list, select a relation that your administrator has configured.
4. Click **Select Work Item From Linked Data Friend Server**.

The **Link External Item (Linked Data)** dialog box appears.

### Link External Item (Linked Data)

**Location:**

Please select location...

- In the **Link External Item (Linked Data)** dialog box, select the location where you want to create the link.

Log on to Active Workspace if prompted.

- In the **Link External Item (Linked Data)** dialog box, you can create a link to a Teamcenter Software Architecture component in two ways:
  - By selecting an existing software architecture component
  - By creating a new software architecture component

### Link External Item (Linked Data)

Location:

TC12REG -> Embedded Software Management

Select an Existing Item...

Create New Item...

- To create a new software architecture component, click **Create New Item**.



An Active Workspace dialog box appears.

- In the **Create** dialog box, click **Software Architecture**.

The **Create** dialog box is updated with new fields.

- In the **Create** dialog box, which is updated with new fields, add the required properties and click **Add**.
- Click **Save**.

The link is created and it appears in the **Linked Work Items** section.

 <b>Linked Work Items</b>		
Suspect	Role	Title
<input type="checkbox"/>	implements	 Circuit Board 2020

## Delete remote links

1. Select a remote link.
2. Go to **More Commands** **...** > **Edit**  > **Delete Remote Link**.

When you delete a remote link, backlinks are also deleted.

## Creating links between Embedded Software Management components

When you create your Embedded Software structure, you can link to different components using the *trace link* functionality. The following trace links are available specifically for Embedded Software Management components:

- **Depends Link**

Represents a relationship where the defining hardware or software is dependent on the complying hardware or software. An example of this is when a secondary bootloader software depends on a primary bootloader software.

- **Compatible Link**

Represents a relationship where the defining hardware or software is compatible with the complying hardware or software. An example of this is when an application software is compatible with a processor.

- **Relates Link**


Represents a relationship where the defining hardware or software is related to the complying hardware or software, for example, when a License is related to the Application Software.

You can add trace links from the diagram or from the **Trace Link** tab.

## Generating reports of trace links between Embedded Software Management components

You can generate reports for the trace links that you created between Embedded Software Management components. These trace link reports are supported only with item revisions. You can use the following options to generate reports:

### Option 1: Export

1. Select the Embedded Software Management component and click **Export**  in the toolbar.
2. Select the **Excel** option.

3. From the **Template** list, select from the following style sheets:

- **ESM\_TraceLink\_complying\_template**
- **ESM\_TraceLink\_defining\_template**

4. Click **Export**.

The report is exported to an Excel file.


## Option 2: Generate Reports

1. Ensure that a report is defined in Report Builder.

Add the following values to the report you create:

- **Name: TL Complying and Defining Report**
- In the **Report Stylesheets** pane, select the **ESM\_Tracelink\_complying\_template** and the **ESM\_Tracelink\_defining\_template** style sheets from the **Defined Stylesheets** list and move it to the **Selected Stylesheets** list.

The previous steps are performed in the rich client.

2. In Active Workspace, select the Embedded Software Management component and click **Generate Report**  in the toolbar.

3. In the **Generate Report** dialog box, select the **TL Complying and Defining Report** report.

Note:

The name of the report depends on how you have defined it in the Report Builder application.

4. From the **Style Sheet** list, select from the following stylesheets:

- **ESM\_TraceLink\_defining\_template**
- **ESM\_TraceLink\_complying\_template**

5. Click **Generate**.

The report opens in an Excel file.

## Upload binary or software package files to Teamcenter from remote systems

You can upload binary or software packages from remote systems such as Polarion to Teamcenter. For the upload to work, the following conditions must be met:

- A remote link must exist between Teamcenter and the object in the remote system (such as Polarion).
- The remote link must be linked to an artifact that has access to software builds or contains attachments.
- Teamcenter and the remote system must be enabled with Teamcenter Security Services.

1. Select a part in Teamcenter.

Ensure that the Teamcenter part links to an artifact in Polarion that has a software build feature such as a Software Release.

2. In the **Attachments** tab, under the **Files** section, click **Upload Remote Files** .

3. In the **Upload Remote Files** dialog box, select a remote link.


Ensure that you select the remote link whose remote resource has a build feature.

Once you select the remote link, the **Dataset Types** appear.

4. Select the dataset type from the list.
5. Type in the dataset name and description.

**Upload Remote Files**
✕

Select a link below to initiate file upload:

Name	
 Release1	

Dataset Types:

Text
▾

Dataset Name:

Power window files

Dataset Description:

Choose Files: ⊕

Upload

The dataset type will filter out only compatible file extensions that are associated with the dataset type, for example, a zip dataset only gives you a .zip or .7z. These associations are managed in Business Modeler IDE.

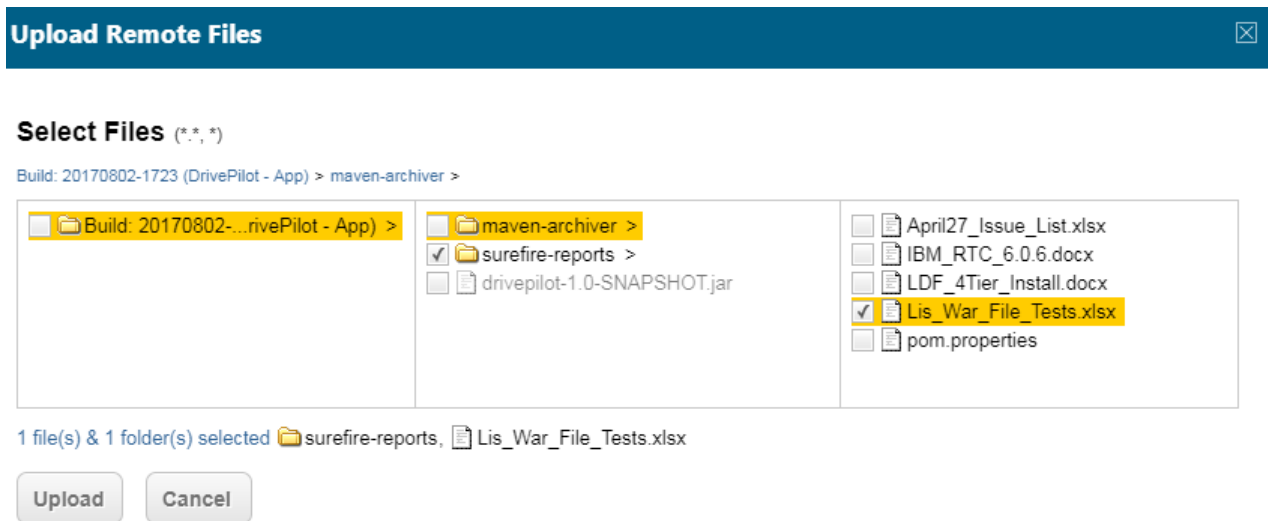
6. Click **Upload** ⊕.

Log on to Polarion if prompted.

7. From the **Upload Remote Files** dialog box, you can upload files or files in folders.

When you select a folder, the files in that folder appear in a list on the right hand side of the dialog box.

Select the files you want to upload and click **Upload**.



The uploaded files appear in the **Files** section. By default, only one file is displayed. If you need to view all the files, you can view them in the rich client.

Overview History Where Used Attachments Relations Collaboration

FILES



**Power window files**

Type: Text



Owner: admin (admin)

Date Modified: 31-Oct-2018 13:55

In case of large files or if there is latency, expect a lag between the start and end of the file upload.

At the end of the file upload process, the user who started the upload receives an alert. The alert is under the category **LDF**.

## View and edit resource information for external applications in Active Workspace

1. Select a remote link and click **Open**  to see information about the link in the **Preview** tab.
2. To edit the external resource, go to **More Commands** **...** > **Edit**  > **Start Edit** and update the resource.
3. To save the edits, use the save functionality of the external application. This is available as an embedded section in Active Workspace.

## View links to objects in external applications using the Relations tab

The Relations tab allows you to view and navigate the relations between different objects in a graph view. With Linked Data Framework, you can view the relations between Teamcenter objects and objects in external applications.

A relations graph shows two elements: objects and relations. When you click the **Relations** tab or expand relations, you may need to log on to external applications.

You can perform all the operations that Relations supports such as expand, collapse, filter content, and change layout with Linked Data Framework objects. You can also expand external objects.

You can also expand to all the levels below the Requirement Collections artifact in Polarion, for example, a Polarion Live Document.

## Open models in standalone Lifecycle Visualization

- To open a model in Lifecycle Visualization, click **Open**  and choose **Open in Visualization**.

Depending on your **configuration**, the Active Workspace session is established in Lifecycle Visualization or only the model opens in Lifecycle Visualization.

Diagram elements are sent to Lifecycle Visualization as follows:

- All elements in the diagram are sent to Visualization.
- Deep traversal of element happens per the element types specified in the **AWC\_LaunchToVis\_TraverseTypes** preference.
- In elements that support deep traversal, logical to physical trace links in the forward direction are shown in the diagram.

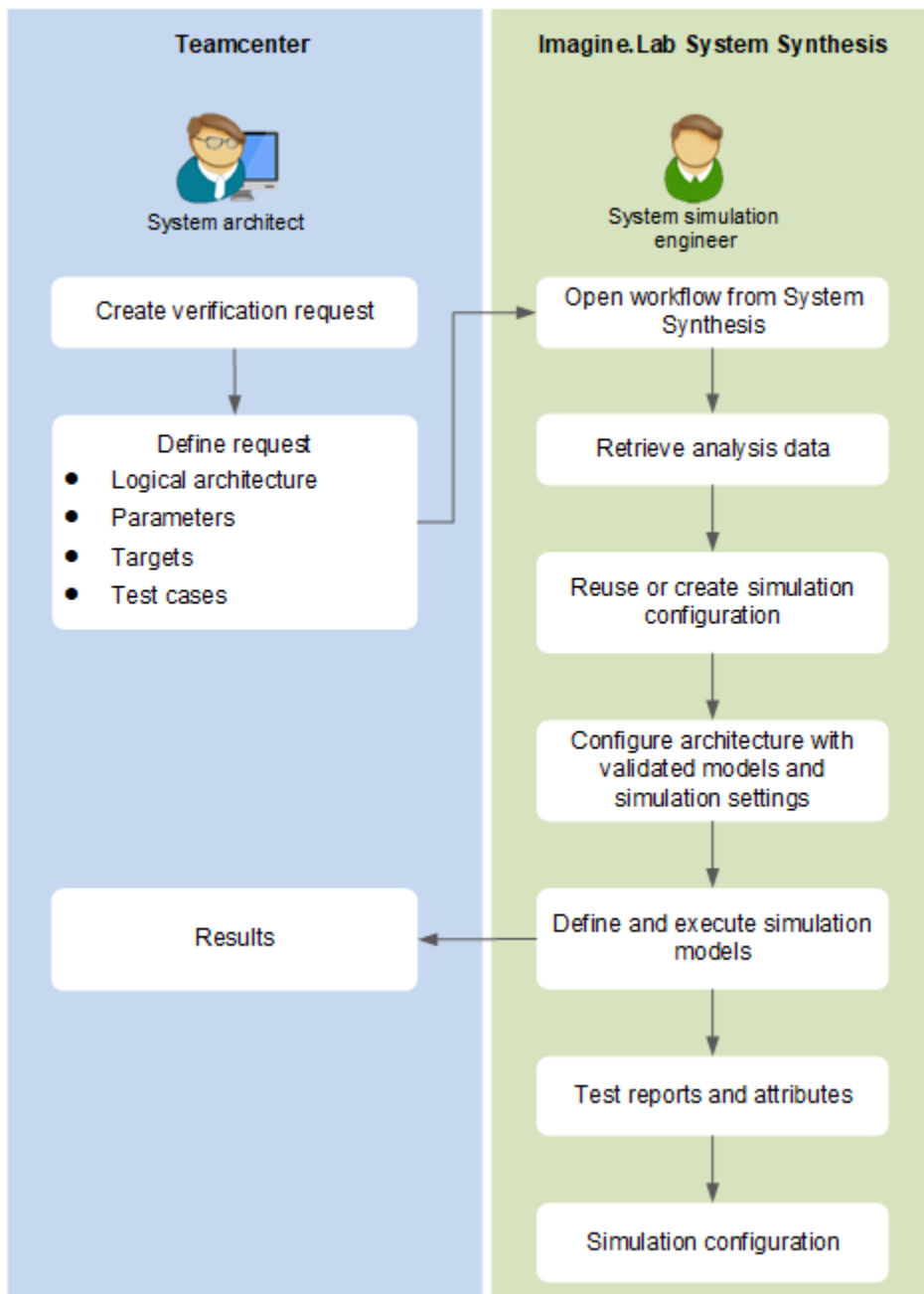
# 5. Managing tests

## Design verification and validation with MBSE

Test management (which may be referred to as verification management in your company) is part of verification and validation, which are similar but distinct processes as part of MBSE, that is used to check that a product or system meets requirements and specifications established to fulfill its intended purpose. Validation assures that the product meets the demands of the customer and any other identified stakeholders. Verification, on the other hand, is meant to evaluate whether a product, service, or system complies with a regulation, requirement, specification, or other imposed condition. Validation can be expressed as "Are you building the right thing?" and verification by "Are you building it right?". Building the right thing refers to the user's needs while building it right checks that the specifications are correctly implemented by the system.

Active Workspace provides a process that allows you to validate if you designed the correct system. For example, it may support the simulation of high-level driving use cases, an aero engine on a test rig being driven through a simulated flight profile, driving a car on a track, or similar use case testing.

The following diagram shows a typical test process. In this example, simulation results are captured from Imagine.Lab, but you can also capture them from other simulation tools or testing methods.



A test initiated and managed in Active Workspace fully describes a task (typically an analysis or test) that the tester performs. It references the related elements and also describes the purpose of the task in sufficient details such that the recipient understands:





- What they are asked to do.
- The context of the request.
- The location of the related data.
- Why they are asked to perform the task.
- The expected outputs of the task.

## Test vs Verification Request



Your administrator has set the use of the UI term as either **test** or **verification request**. Therefore, in Active Workspace you will encounter either term. This documentation uses **test** and **test management** as the default terms.

## Roles

The following table shows the high-level roles engaged with test management:

 Administrator	MBSE administrators includes those who install MBSE and those who configure the specific areas of MBSE such as requirements management or test management.
 System architect	Responsible for the high-level design of the product and owns the system model.
 System designer	Executes the direction provided by the system architect and owns the designated subsystem domains.
 Simulation engineer	Runs assigned tests or simulations and reports the results.

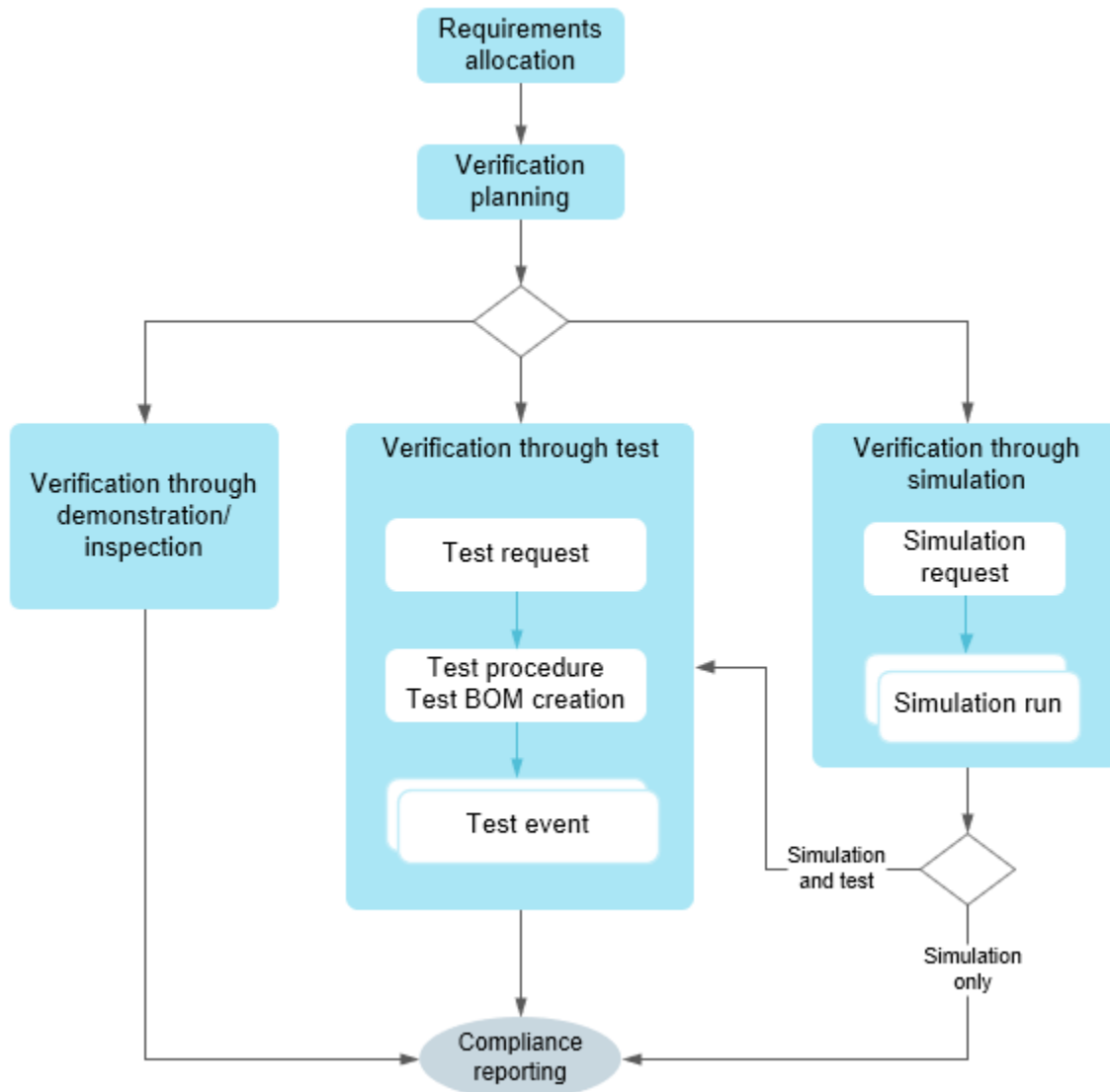
## Where do I go from here?

 Administrator	See Configuring test management.
 System architect, system designer, simulation engineer	
Learn more about how the MBSE features work together in Active Workspace.	See What is Model-Based Systems Engineering (MBSE).
Learn more about the validation process, roles, and validation objects.	Start with the topic Recommended design validation process.
Get started with parameters for use with your testing.	For information about creating and working with parameter dictionaries, see Managing dictionaries of parameter definitions.  For information about working with parameters and tests, see Test and parameter project configuration relationships
Get started with creating tests.	Start with the topic Test package objects.

## Recommended design validation process

### Design validation process through testing

Test management is part of a larger verification management process. The following graphic shows the recommended process using Active Workspace.



The test management process employs multiple Active Workspace features:

- Test management
- Program planning
- Teamcenter reporting and analytics
- Service asset manager
- Classification

This process provides the *proof of compliance* to help prove that you performed the necessary tests to validate the requirements:

1. Create and link requirements to systems, assemblies, and subassemblies.

In the requirements allocation process, you author the product requirements and allocate the requirements to the systems and assemblies that form the product engineering bill of materials (EBOM).

2. Plan the verification activities.

In the verification planning process:

- Plan the activities required to identify the method for verifying that the product requirements are met.
- Identify the different simulation models and test article configurations to perform the verification activity.
- Refine the test needs into test methods and procedures.
- Plan the timing needs for test execution.
- Establish the source systems relationship and definition through CAD or physical test BOMs.

3. Initiate the test to test a requirement collection.

This step includes defining the parameters to measure.

Active Workspace sends the test to the test engineer or operator.

Communication among team members throughout testing is managed through provided workflows.

4. Verify the requirements through demonstration or inspection.

You can verify requirements using a demonstration or an inspection, and then document the verification results in Active Workspace.

5. Verify the requirements through analysis and simulation.

In this method, you perform an engineering design analysis or simulation, and then determine if the design satisfies the product requirements.

6. Initiate and perform a test for requirement verification:

- a. Initiate the test request.

A test request describes the tasks you perform to initiate a test.

- b. Prepare the test article (part/system).

When verifying by test, you initiate a physical test on a test article. The test request process includes tasks to create a Test Engineering Bill of Material (Test EBOM) and a test request with specific measurement requirements, inspection requirements, and test conditions. Each required measurement must have an approved and calibrated test instrument assigned.

- c. Perform the test.

At the end of this process, an Individual Test Readiness Review (ITRR) is performed to ensure that all steps required to proceed to the execution step are complete.

7. Report the compliance and the verification status.

After completing the different processes to collect data for verifying the set of requirements, you create a compliance report, and then mark the requirements as either verified or not verified.

Continue to monitor the test collection status for a given product or project.

### Create and link requirements to systems, assemblies, and subassemblies

Requirements allocation is a set of tasks you perform to author and manage product requirements. You can author requirements in Active Workspace or you can author the requirements in an external authoring tool and import them.

Test requirements are derived from product requirements and provide documentation of explicit test requirement needs. They define a means of compliance, for example, simulation, inspection or physical test.

The following is an example of the requirements authoring and allocation business process for physical testing.

1. Create a program.

A *program* is a collection of resources and assets with a common objective, for example, the design and development of a new aircraft. Before you author the requirements for the product, create a program.

**Feature:** Program Planning

For more information about programs in Active Workspace, see [Creating and managing programs](#).

## 2. Create or import requirements.

*Requirements* describe the product or system that the customer will buy. They communicate the customer's specifications to the various disciplines involved in the development process. You start a new product development process by gathering requirements from customer specifications, standards, and other relevant sources. You then author the requirements in Active Workspace or import your existing requirements into Active Workspace.

**Feature:** Requirements management

For more information about creating or importing requirements in Active Workspace, see *Acquiring and authoring product requirements*.

## 3. Create a high-level product EBOM.

You define the high-level system architecture (system models) and preliminary EBOM for the product.

**Features:** Requirements management, Designs and structure management

For more information about system modeling and structure management in Active Workspace, see *Managing systems and Managing designs and product structures*.

## 4. Link requirements to systems, assemblies, and subassemblies.

As you develop the system architecture and preliminary design of the product, you also allocate requirements to systems, assemblies, and subassemblies to ensure that each requirement is traceable to a specific product structure. To do this, create trace links between a requirement and the corresponding product structure node.

**Features:** Requirements management, System modeling

For more information about trace links, see *Associating requirements with system model blocks or workspace objects using trace links*.

## Plan test activities

*Test planning* provides the tasks you perform to plan the requirements test activities. In the test planning process, you initiate the test activity by defining the test methods for each requirement.

During the planning phase of a program, one of the challenges is the identification, consolidation, and bidding of the test articles and simulation models required to verify the product requirements. Defining test articles and simulation model configurations and allocating them to appropriate requirements at an early stage of the program is critical for the success of the program. A Bill of Test provides a collection of test articles with their configurations and a Bill of Simulation provides a collection of simulation articles with their configurations for planning purposes.

### 1. Set test methods for requirements.

To verify a product requirement, first decide the method of test. You can verify a product requirement by conducting an analysis, performing a physical test, conducting an inspection, or through a demonstration. You can also consider a requirement to be verified if a similar requirement is verified using any of the earlier mentioned methods.

Methods of test are set as a property on the requirement. You can set more than one method of test on a requirement, one of which must be the primary verification method Method and has the highest priority. For example, if testing is done using a combination of simulation and test, set the primary verification method to **Test** and the secondary verification method to **Simulation**.

**Features:** Requirements management, Test management

### 2. Create a Bill of Test.

Create a top-level Bill of Test Article object to create a Bill of Test (test BOM) for the program. This top-level Bill of Test Article node acts as a collector for all the test articles planned for the program. The individual test article nodes are then created as children of this top-level node. The configuration of each test article is defined by creating a multi-level structure of constituent test parts. You can relate images or CAD files and other file formats, such as Microsoft Word or PowerPoint datasets under the test parts of the test articles.

**Feature:** Structure editing (using test objects)

### 3. Link test articles to requirements.

At the time of finalizing the Bill of Test Article for a program, you link the product requirements to the planned test articles to maintain traceability. Ensure that each requirement that is planned to be verified by a physical test has a corresponding test article in the Bill of Test.

**Features:** Requirements management, Test management

For more information about trace links, see [Associating requirements with system model blocks or workspace objects using trace links](#).

### 4. Create a Bill of Simulation.

The Bill of Simulation is useful to track the simulations planned for a given test requirements. Create a top-level Bill of Simulation Model object to create a Bill of Simulation for a program. This top-level Bill of Simulation Model node acts as a collector for all simulation models planned for the program. The individual simulation model nodes are then created as children of the top-level node. The configuration of each simulation model is defined by creating a multi-level structure of constituent simulation parts. You can relate images, CAD files, and other file formats.

**Feature:** Structure editing (using simulation objects)

5. Link simulation models to requirements.

At the time of finalizing the Bill of Simulation Models for the program, ensure that each requirement that is planned to be verified by analysis or simulation has a corresponding simulation model in the Bill of Simulation. Create a trace link from a product requirement revision to a simulation model revision in the Bill of Simulation.

**Feature:** Trace link creation (using trace links)

6. Create tests and link to requirements.

You create a test to collect multiple requirements for a specific test activity. Instead of creating trace links between individual requirements and its corresponding objects, you can link the requirements and the complying objects to the test to avoid creating multiple trace links.

**Feature:** Requirements management, Test management

For more information about trace links, see *Associating requirements with system model blocks or workspace objects using trace links*.

7. Create a detailed EBOM design.

Use the Active Workspace's Model-Based Systems Engineering, product structure, and CAD integration capabilities to accomplish this task. You can create additional derived requirements as part of the detailed design activity.

**Feature:** System modeling, Designs and structure management

8. Review requirements stability and allocation.

Before initiating test activities, ensure that all the program requirements, including any derived requirements are allocated to systems, assemblies and subassemblies, to ensure complete traceability. You should also ensure that the requirements are stable enough to start the test activity.

You can generate a Requirement Stability report with Teamcenter Reporting and Analytics. This report counts all the requirements that are released for a particular program over time. The number of requirements released during each month is plotted as a bar graph for the relevant period. The total count of the requirements released over this period is shown on a dial chart. The released requirement count includes the initial release and all subsequent releases after any requirement change.

**Feature:** Reporting and analytics

9. Initiate test.

You submit a test request in an Active Workspace workflow. A sample **Initiate Verification Activity** workflow is provided but you can create your own workflow to suit your business needs.

**Features:** Test, Workflow

## Verifying and testing against requirements

### Verify requirements through simulation

Simulation comprises tasks to verify requirements using either the Simulation, or the Simulation and Test method. Simulation Process and Data Management in Active Workspace contains a comprehensive set of capabilities to enable you to verify or improve a design in the early stages of the product lifecycle. It uses Teamcenter to manage CAE or simulation data on a centralized basis. You can integrate data from a CAE system of your choice into the PLM environment.

1. Create a simulation test.

Create a test for simulation and identify specific conditions for this simulation activity to initiate the simulation process.

**Feature:** Simulation process management

2. Use preprocessing to create CAE models.

You preprocess the part geometry to prepare data for performing the CAE analysis.

**Feature:** Simulation process management

3. Link the Bill of Simulation model to the CAE model

You must relate the CAE model to the corresponding analysis model identified in the bill of simulation. This ensures that all simulation models identified in the Bill of Simulation during the planning phase are traced to an actual CAE model.

**Feature:** Trace links

4. Set up load and boundary conditions.

You add load and boundary conditions and prepare the CAE model for an analysis using NX or any third-party tool.

**Feature:** Simulation process management

5. Solve the analysis.

Use Active Workspace to launch the solver of your choice. You can also use NX to prepare all the data and then launch the solver from NX.

**Feature:** Simulation process management

#### 6. Postprocessing CAE results.

After the solve is complete, you manage output files and share postprocessing displays with Active Workspace. You can generate JT files or other visualization files and create reports by using your postprocessor, and then import them in Active Workspace.

**Feature:** Simulation process management

#### 7. Update CAE models.

You can repeat the analysis after the results are peer reviewed by revising the CAE model. You can do one of the following:

- Revise the CAE model in Active Workspace by using the batch meshing tool to refine the mesh without entering a traditional meshing application.
- Update the CAE model in NX using Active Workspace integration for NX.

**Feature:** Simulation process management, NX

#### 8. Update load and boundary conditions.

After updating the CAE model, you can use NX or any third-party tool to update the load and boundary conditions and prepare the revised CAE model for analysis.

**Feature:** Simulation process management, NX

#### 9. Create a simulation report.

Relate the report to the simulation test object using a report trace link. You should also relate the data generated during the simulation object to the simulation test using a data trace link.

**Features:** Simulation process management, Trace links

## Verify requirements through demonstration or inspection

You can verify requirements using a demonstration or an inspection and then document the test results in Active Workspace.

## Initiate test request

You can create configurations of the test EBOM to verify the product requirements using a physical test. The test EBOM configurations are based on a specific configuration of the product EBOM. After you create a test EBOM, you can update it by incorporating specific product EBOM changes. You then identify the measurement and inspection requirements and the required test conditions for your test request. Use the result of the analysis and simulation process to specify the measurement locations for the test request.

### 1. Create a test EBOM.

You can create the EBOM definition of the test article using a specific production EBOM configuration. Your test article can be based on the top-level production end item, or any of its lower level child assemblies, sub-assemblies, or parts. Your production EBOM can use any of the configuration capabilities available in Active Workspace, such as options and variants, and occurrence or revision effectivity.

**Feature:** Structure editing and management

### 2. Link a test article to a test EBOM end item.

You can relate revisions of the test EBOM end item to the corresponding test article identified in the Bill of Test article. Ensure that all test articles identified in the Bill of Test article during the planning phase are traced to an actual test EBOM.

**Feature:** Trace links

### 3. Add a new EBOM configuration,

After you create a test EBOM, you can add new configurations for your testing needs. For example, you can add a part in the test EBOM for your initial set of tests and then replace that part with a different part for subsequent tests. To do this, in Active Workspace, apply the default unit occurrence effectivity to parts in the test EBOM to make it unique for a specific configuration of the test EBOM. You can now configure the test EBOM with unit effectivity to see only the parts required for a specific configuration. Apply unit effectivity on occurrences using the Product Structure capabilities in Active Workspace.

**Feature:** Designs and structure management, Trace links

### 4. Update the test EBOM with the product EBOM changes.

If you create a copy of a production part to include in your test EBOM, any change to the production part does not affect the test EBOM. However, as a design engineer or a test engineer, you know the changes and should selectively incorporate any changes made to the production part to the corresponding test part. In the change management process for the production EBOM, all impacted departments are notified of changes made to a production part. We recommend that you implement this process using Active Workspace workflows.

After you are notified that a production part with a corresponding copied test part is modified, you can use the compare capability in Active Workspace to determine the difference between the production EBOM and the test EBOM.

**Features:** Designs and structure management, Workflow

#### 5. Conduct analysis for test EBOM.

Ensure that each requirement that is planned for analysis or simulation test has a corresponding simulation model in the Bill of Simulation. Create a test trace link from a product requirement revision to a simulation model revision in the Bill of Simulation.

You conduct a CAE analysis of the test EBOM to:

- Analyze and predict test article performance to guide detailed test design.
- Help select the position of instruments for measurement requirements.
- Validate instrumentation and test cases to observe all critical phenomena.
- Define and de-risk detailed test plans.
- Prepare data to support test-setup validation.

Note:

A test case is a subtype of a requirement. In turn, a test procedure is associated with a test case. A test case can be consumed by tests and studies. A test case can be associated with requirement subtypes including inspection requirements and measurement requirements.

#### 6. Create a test request.

Initiate the test process by creating a test request and by identifying specific requirements for this activity. You can relate all the measurement and inspection requirements and test conditions to this test request.

**Feature:** Test management

#### 7. Define test requirements.

A key step in the test request process is to specify the measurement and the inspection requirements. You can review the data obtained during the analysis phase of the test process to determine the exact measurement locations and the specific conditions (such as temperature and pressure) to define the requirements for the test request. You can create the following requirement types to use in a test request in Active Workspace:

- Measurement requirement

You typically define minimum, maximum and nominal (goal) attribute values for each measurement.

- Inspection requirement

- Test condition

**Feature:** Test management

### Prepare a test

A test preparation process describes the tasks you perform to prepare the test article (EBOM).

1. Modify the test EBOM to add instrument models.

When you receive a test request with the related test EBOM configuration, add instrumentation at a specific location in the test EBOM to satisfy measurement requirements. Review the measurement requirements and, based on the specifications, search for the appropriate instruments in the instrument library, for example, by using Classification attributes.

2. Create test procedures.

After you finalize the engineering definition of the test article configuration for a specific test request, create a test procedure with detailed instructions for the operator to perform the test.

3. Generate a test article physical structure.

Build the test article for a specific configuration of the test EBOM, creating manufacturing plans and work instructions to build the test article.

Then store the physical configuration of the test article. Record the serial number or lot number for the serialized parts and lot parts for the test article. Use an approved deviation to use a part in place of a design part in the physical test article.

After you generate the test article physical structure, install physical instruments at the appropriate location by reviewing the calibration data. The calibration data is attached to each physical instrument.

4. Rebase test article physical structures.

After the initial build of your test article, you can update the test article to match a different configuration of the test EBOM based on the following conditions:

- A new configuration in the test EBOM for a new test request.
- Production EBOM changes to the test EBOM.

To update the physical test article, you should first compare and review the differences between the new configuration of the test EBOM and the physical test article.

5. Create a test event.

After you build the test article based on the original configuration or an updated configuration to reflect a new configuration of the test EBOM, you should initiate the test activity. To do this, create a test event.

6. Assign channel numbers to instruments.

You should assign channel numbers to all the instruments that are installed in the test article.

7. Create a conformance report.

Perform an inspection to ensure that the physical test article conforms to the requirements specified in the test request. You then compile a conformance report that is approved by the relevant authority. Create a trace link from the physical structure to the approved conformance report in Active Workspace.

8. Conduct an individual test readiness review.

Before conducting the test, perform a review to ensure that all data and assets required for the test request are available and approved.

## Perform a test

A test describes the tasks that you perform to execute a test request after obtaining the approval from the individual test readiness review. You initiate the test execution process by submitting the test event in a workflow. The test is performed in an external system. You process the raw data recorded by the measuring instruments using the calibration data stored in the system, and then attach both the raw and the processed data. If the test execution is successful, a quick look and final test report are generated and linked to the test request. This test report provides results of the tests to the test planner.

1. Initiate the test.

Initiate the test to record the raw test data in your test system. Use the calibration information for each instrument to generate the processed data from the raw data.

2. Link test data to a test event.

After completing the test, review the status of the test execution, set the state of the test event to the relevant state, typically, Success, Failure or Partial. If the test event state is Failure or Partial, run additional test events until all steps specified in the test procedure are completed successfully.

Create data trace links from the test event to the raw and the processed data.

**Feature:** Trace links

3. Generate a test report and link it to a test request.

Create data trace links from the test event to the raw and the processed data.

## Submit a test to the test engineer

Design engineers can use their company's workflow to manage all assignments given to test engineers.

Note:

Step 5 onward in this process are described later in *Capturing measurements*.

1. The design engineer creates and populates the test.
  - Adds and configures each domain as a 150% structure, for example, the system model, requirements, and test cases.
  - Adds elements to the test.
  - Sets the **input and output parameters**.
  - Creates studies, simulation requests, test requests, test events, and runs for the test. Adds elements, sets a variant rule on the domain and structures for the test package (if appropriate), and then assigns the input and output parameters.
  - Assigns one or more test (simulation) engineers to the test by opening the **Participants** tab and clicking ⊕. On initial creation, no test (simulation) engineers are assigned to the test.
2. The design engineer marks the test as complete in the workflow task.

The system locks the input data of the test and studies. It then assigns a review task to the test (simulation) engineer.

Caution:

Do not modify the domain structures once the test is complete. The structures must be static and released.

3. The test engineer reviews the configuration of the test and accepts it in the workflow task.

The system sets the test to an analyzing state.

Note:

If the test engineer rejects the test, the system returns it to the design engineer for rework.

4. The test engineer starts verification, performs the tests, and updates the outputs of the test and studies with measurements.

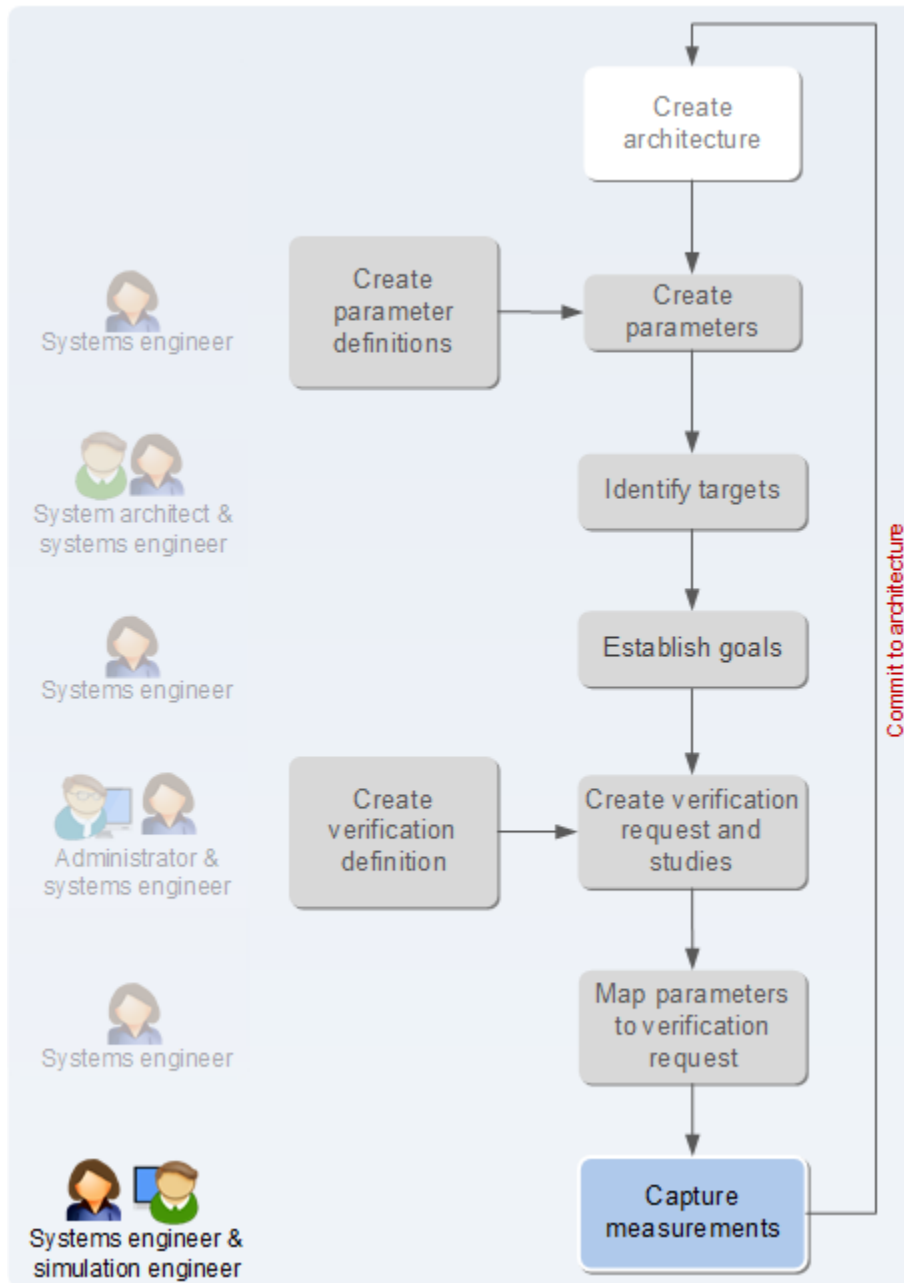
5. The test engineer marks the verification task as complete.

The system locks the output data and assigns a review results (analyzed) task to the design engineer.

6. The design engineer reviews and accepts the results.

The design engineer publishes the results from the test to the domain models. The system unlocks the domain structures, adds a baseline to the test, and then releases the test revision. The workflow is now complete.

## Capturing measurements



A test activity is typically executed by several test engineers. Each engineer is responsible for one or more studies and the associated measurements, and also for updating the corresponding test output attributes.


Once the verification activity is complete, the system architect commits the validated measurements to the system architecture. You cannot commit the verification results if the verification activity is not complete; this prevents commitment of inconsistent results.

The verification activity is coordinated by a workflow that distributes the verification tasks and updates the test release status. The workflow is customer specific and depends on your business process. Likewise, the test status that allows you to commit it is configurable.

Several tests may be run against a single product for a test. In such cases, the parameter captures the values in context. For example, a car may have a parameter for braking distance. The braking distance has multiple conditions to satisfy depending on environmental factors such as wet road or dry road. The parameter's measured values contain each condition-specific value in relation to the product once accepted by the design engineer from the test. That is, braking distance (wet) and braking distance (dry) are measured values of the parameter for the car in the given context.

## Commit measurements of verification to the system architecture

The design engineer commits the output measurements of the verification activity back to the system architecture, making them available to the enterprise after evaluation. Typically, you are notified by a workflow task when the verification activity is complete and that the results are entered.

1. Accept the workflow task.
2. Open the test in the content panel and click **Details > Parameters**.
3. Edit the measured value of each parameter and the method by which the value was obtained.
4. Commit the changes when all edits are complete by navigating to **Inputs**, selecting one or more parameters, and then clicking **Synchronize** .

The **Synchronize** icon appears only if you are the assigned design engineer, the test status is **Unpublished**, and measured values are captured for all output parameters.

Active Workspace displays a confirmation message stating that out-of-date input parameters will be refreshed and measured values of output parameters will be removed.

5. Click **Synchronize** to continue.

The system publishes the measurements to the model. The status bar on each relevant parameter changes from yellow to green, indicating that it is published to the product.

You can now view the updated output measurements in the domain.

**Tip:**

The **Results** tab of the test and study show a result property of **Pass** or **Fail**. You set the correct value, depending on the outcome of the task.

## Understanding the Status property

Use the **Status** property column in the parameters table to determine if the parameters need to be synchronized or published. This column displays one of the following values:

- **Unsynchronized** or **Synchronized**

Indicates whether one or more of the selected input parameters should be synchronized.

- **Publish** or **Unpublish**

Indicates whether one or more of the selected output parameters should be published.

- Blank indicates one of the following:

- This is an input parameter aligned with the source attribute data, that is, there are no changes from the initial data or the source parameter has not changed.
- This is a parameter with outputs that currently have no measurements.
- This parameter is currently unused.

The **Synchronize** command appears when one or more of the selected parameters are out of synchronization. When clicked, it updates the **Goal**, **Min**, **Max**, and **Measurement** fields.

The **Publish** command appears when one or more of the selected parameters is publishable and the test workflow is in the **Publish** state. When clicked, it updates the **Measured Value** property on the source parameter for each selected publishable measured value.

## Report compliance and test status

Compliance reporting is the last phase of the test activity for collecting requirements grouped in a test. Create compliance reports based on the test results gathered using any of the following methods for the sets of requirements:

- Demonstration
- Inspection
- Analysis and simulation
- Physical test

1. Generate a compliance report.

After you receive the result from all test activities, you examine the demonstration, inspection, analysis, and test results. Compile a compliance report and submit it to a workflow for review.

**Feature:** Test management

For more information about reporting, see Reporting test results.

2. Link the compliance report to the test.

Create a trace link to associate the compliance report results to the initial test.

**Features:** Requirements management, test

3. Set the test status for requirements.

Set the test status of the requirements at the end of the test management process. The test status represents the test activity result for a specific requirement, for example, pass or fail.

**Feature:** test

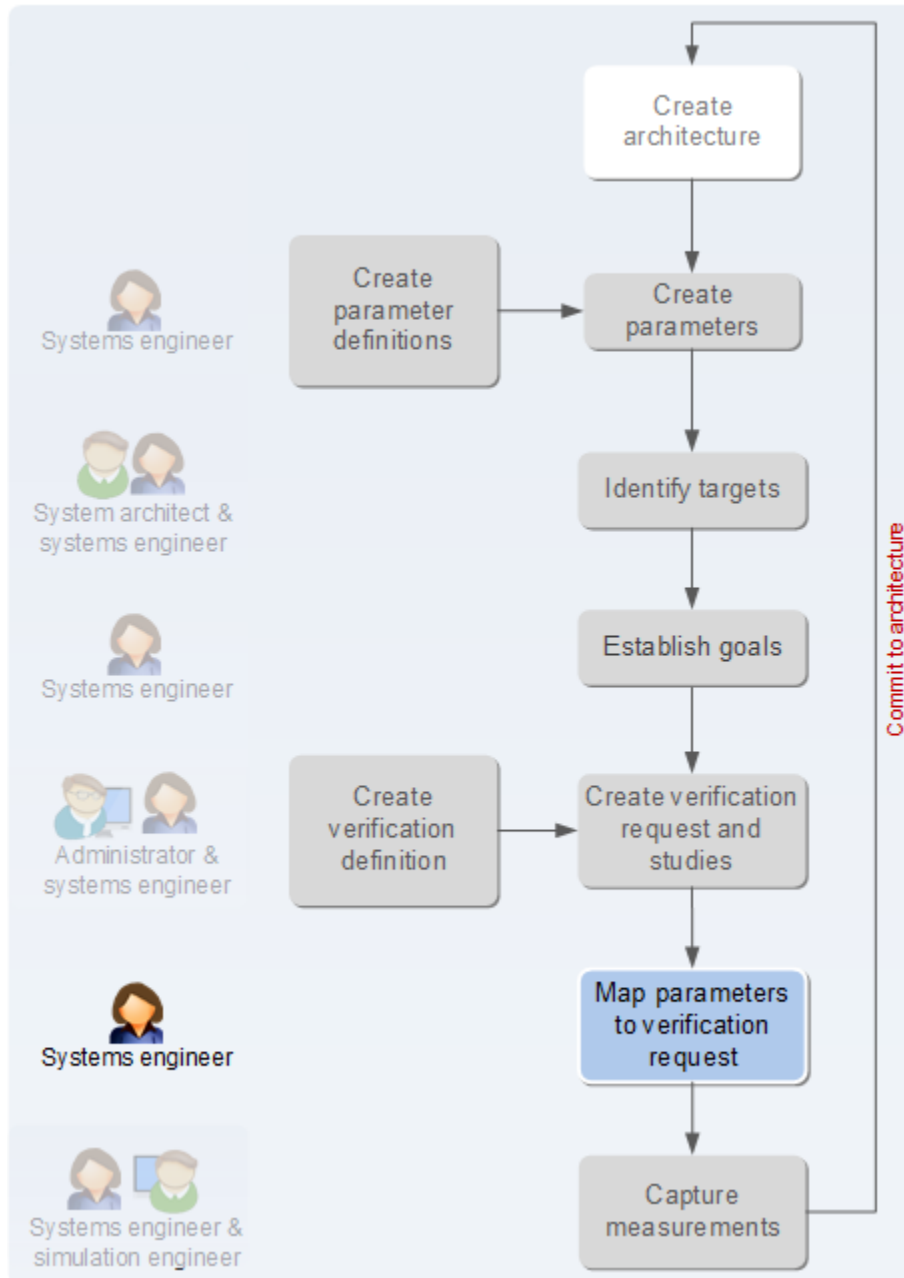
## Managing test parameters

### Parameter management overview

The topics in this section provide general information and procedures about managing parameters. For simulation-specific parameter information, see [Multidisciplinary Analysis and Optimization \(MDAO\) in Active Workspace](#).

## Associating target values with verification requests using parameters and tests

### Managing parameters associated with a test



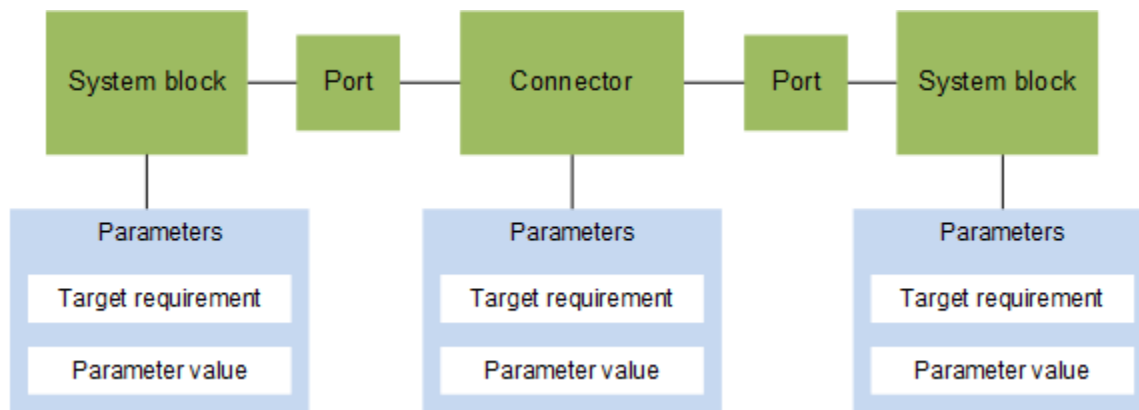
The parameters provided are:

## Parameters

A *parameter* is a characteristic of a model that provides a measurement of form, fit, and function in terms of expected behavior. In a vehicle product, examples of parameters include fuel economy, carbon monoxide output, particulate matter output, and drag coefficient. To meet the target, users can access parameters to manipulate the desired form, fit, and function.

- An input attribute that sets a parameter to configure a test activity.
- A goal (optimal value) for a test activity.
- An output attribute that holds the measurement resulting from the test activity. This is returned to Teamcenter for evaluation.

In a system architecture, parameters can be placed on a system block or a connector.



For example, when modeling a fuel economy scenario, you may define the following parameters:

- The alternator load of the product.
- The drag coefficient (a constant value).
- The gear shift pattern of the product.
- Gear ratios. These are input on an analysis request and are related to the alternator load to support calculations.
- Torque. Input to an analysis request and related to the shift pattern to support calculations.

### Table value parameters


A table value parameter is a multidimensional parameter represented as a value table (2D table) for the individual **Goal**, **Min**, **Max**, and **Measurement**.

For more information on table value parameters, see [Understanding multidimensional table value parameters](#).

### Refine parameters associated with a test


1. Open a test and click **Overview**.
2. On the **Parameters** table, click **Edit**  and double-click editable fields to edit values.

Note:

Click **Set Usage**  to change the parameter to **Input** or **Output**.

3. Optionally, import and export parameter values with Microsoft Excel.

This allows you to import test results from a previously exported Excel file.

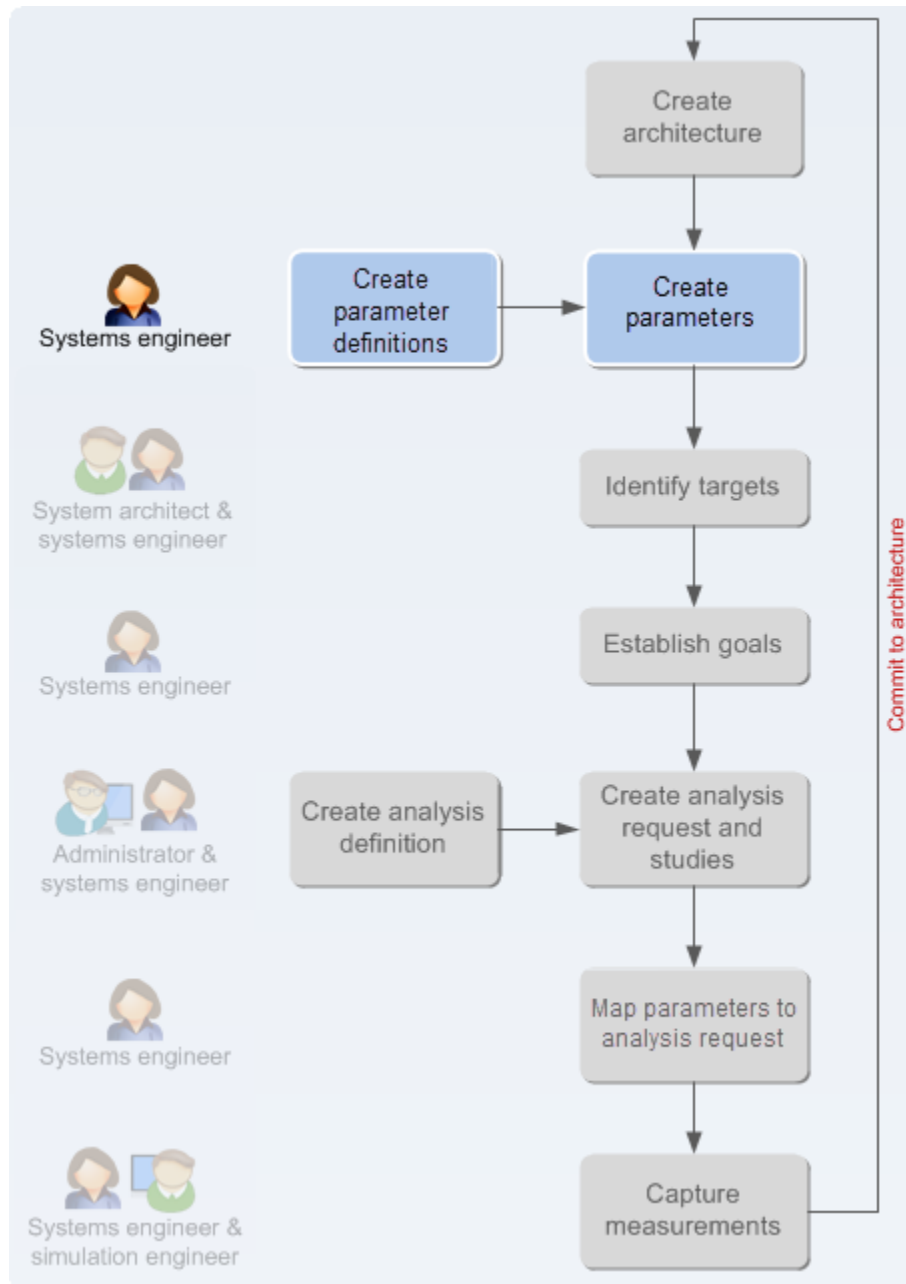
4. Click **Save** .

Note:

You can filter the parameters in the parameter table by clicking **Filter Parameters**. This command filters the table based on single or multiple selections in one or more of the test element tables.

## Defining and specifying target values using parameters

### Creating parameters



### What is a parameter?

A *parameter* is a characteristic of a system that you can observe externally. A parameter has a target value or value range that the product satisfies before you deliver the product to customers. The parameter comprises:

- A goal.

The goal is the result that you want to achieve from the analysis.

- A measured or validated value.

The measured value may be complex, for example, a chart or spreadsheet, rather than an integer. If so, it is stored in an external file or dataset, for example, the mass, cost, speed, pressure, or other constituents of energy flow.

A parameter is specified in a unit of measure (UOM). More than one UOM with the same name may exist, for example, a UOM called **Length** may have a unit of **meters** or **feet**.

A parameter has a unique name. However, you can rename the parameter later, including once it is already used inside an analysis request or study.

A parameter may be associated with more than one item revision or occurrence. You can copy and paste a parameter between item revisions or occurrences, and the goal and value are also copied (the value may be modified later if required).

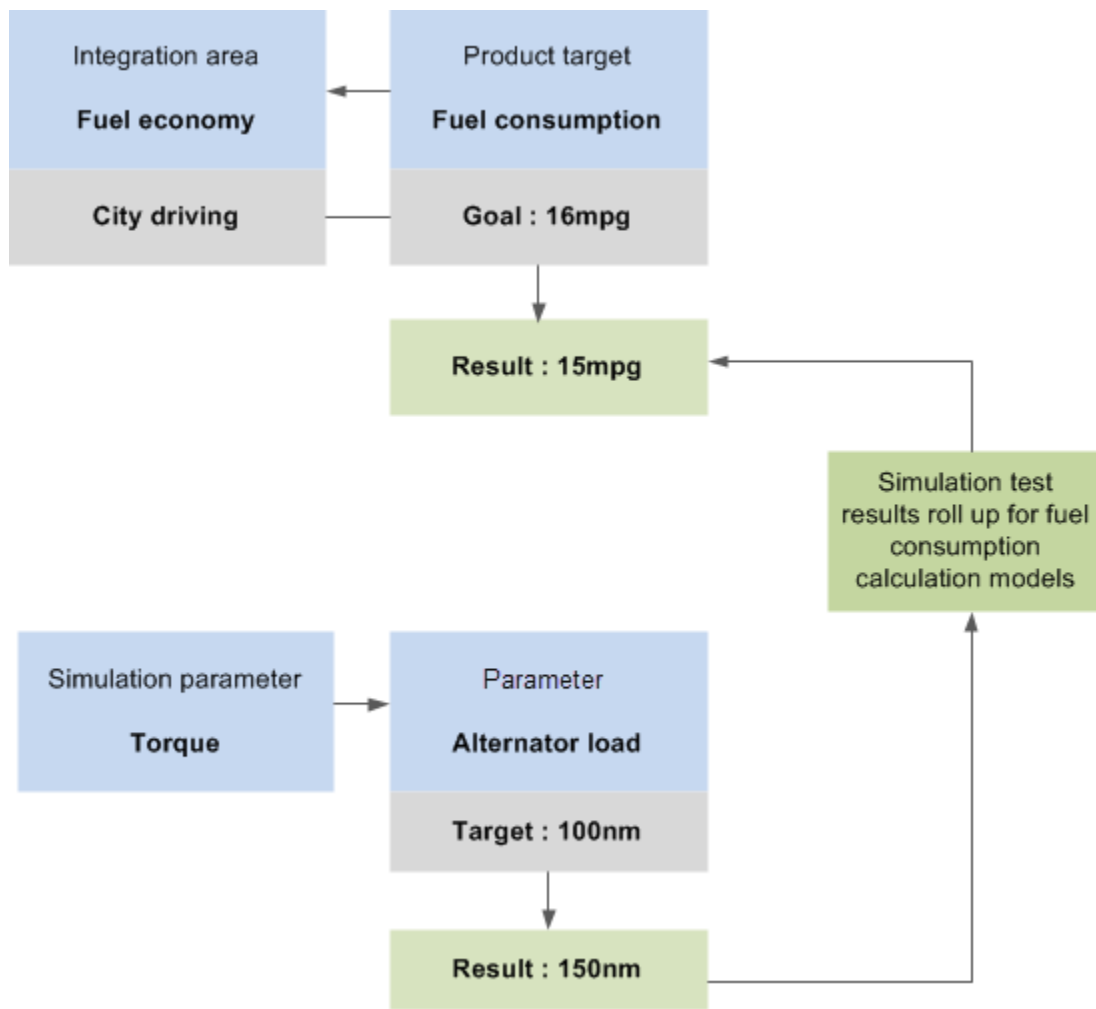
A parameter may also be associated with a particular modeling tool, for example, NX or other MCAD software. If so, you can optionally define the projected modeling tool when you create the parameter. If it is subsequently consumed by a different modeling tool, you can open the **Information** panel for the parameter and update the property to show the actual modeling tool. If the capability is supported by the modeling tool integration, the tool may set the **Complying** and **Fulfilling** properties on the parameter.

Parameters of the **Point** type are displayed as comma-separated values (for example, **1.55, 23.8, 65.1**) with the deviation in a separate column if one is defined.

A parameter may have more than one measured value over time. It may have several sources, for example, estimated, calculated, and measured. It may also be seen in context of more than one domain.

Parameters are always associated with an object, for example, a block, connection, target, or analysis request. Typically, parameters are specified in the context of a block, but the measured values are captured in the context of the product, for example, front and back wheels may have different values.

The following diagram shows how parameters and product targets are related.



## What is a parameter definition?



A *parameter definition* identifies the parameter, together with its unit of measure (UOM) and type. The rigorous use of definitions ensures that, when actual parameters are created for connections and system blocks, they reference a consistent terminology. This allows their life cycles to be managed and approved in a standard process. For example, you may have parameter definitions for mass, cost, speed, and pressure.

## Define a parameter definition

Tip:

If you prefer to create parameters without requiring a parameter definition, you can **set a preference** to turn off the parameter definition requirement.

You can define the type of measurement detail required to provide a consistent specification of target values or measured values. You can define complex characteristics in measurement and goal templates attached to the parameter definition.

1. Create a folder to hold parameter definitions of the relevant type or subtype, if you have not already done so. (If your site uses subtypes of parameters, you can create or modify them in the Business Modeler IDE.)
2. Open the parameter definition folder and then click **More Commands** **...** > **New**  > **Add** .

The **Add** panel appears.

3. Find and select the **Parameter Definition** type and enter the required properties.

The parameter definition name and unit of measure (UOM) must be unique in the system. You can define one value or a list of possible values for each parameter. The user can only select a value that you have defined.

In releases prior to 3.2, it was only necessary for the parameter definition name to be unique in the system.

4. (Optional) Perform any of the following:
  - To allow users to select from a defined value list, enter the values one line at a time in the **List of Values** field.  
  
If you want to allow users to enter their own values, then clear the **Restrict List Of Values** check box.
  - To allow parameter overrides by default, then select the **Default Allow Override** check box.


5. Click **Add**.

The system creates the parameter definition.

6. (Optional) If you intend to capture complex values, do the following to attach a goal template to the parameter definition.

- a. Click **Attachments**.

The **Goal Template** table appears.

- b. Click **Add** .

The **Add** panel appears.

- c. Click **Select File** to manually select the file, or browse to the file and drag and drop it in the **Add** panel.

**Note:**

Depending on your business conventions, you may be required to change the template file name before attaching it to the parameter definition.

- d. (Optional) Select a **Type** for the template.
- e. Click **Add**.

**Note:**

If the parameter definition is already released, you cannot attach a goal template and an error message appears. Instead, create a new revision of the parameter definition.

7. (Optional) Do the following to attach a measurement template to the parameter definition.

- a. Click **Attachments**.

The **Measurement Template** table appears.

- b. Click **Add** ⊕.

The **Add** panel appears.

- c. Click **Select File** to manually select the file, or browse to the file and drag and drop it in the **Add** panel.

**Note:**

Depending on your business conventions, you may be required to change the template file name before attaching it to the parameter definition.

- d. (Optional) Select a **Type** for the template.
- e. Click **Add**.

**Note:**


If the parameter definition is already released, you cannot attach a measurement template and an error message appears. Instead, create a new revision of the parameter definition.

8. Open the template files in turn in the appropriate external application, for example, Microsoft Excel, edit them until they contain the required values, and then save any changes.

9. Do one of the following:

- Click **More Commands** **...** > **Manage**  > **Submit to Workflow**.

The **Submit to Workflow** panel appears.

- Right-click the diagram in the structure and then click **Submit to Workflow** .

The **Submit to Workflow** page appears.

10. (Optional) Select the **Template**, and update the **Name** if necessary.

The name of the workflow is automatically populated. Workflow templates are specific to your enterprise as the tasks included are specific to your business needs. Depending on your business conventions, you may be required to change the name before submitting.

11. (Optional) Click **Assignments** to assign tasks for reviewing and signing off on the parameter definition.

12. Select a group of users in the **Assignment List** or add individual users or a resource pool to the **Assignee** list.

13. Click **Submit**.

### Update a parameter definition

You can modify parameter definitions to incorporate additional attributes as the process matures.

1. (Optional) Create a new revision of the parameter definition.
2. Edit the properties you want to change and then save the changes.




Note:

If the current revision is released, you cannot change the type (for example, an integer to a string) or the unit of measure.

### Submit a parameter definition to a workflow

To avoid possible training or consistency issues, you can direct users to select from approved parameter types. Once you create a parameter definition, you can then submit it to an approval workflow.

1. Open a parameter definition from the structure.
2. Do one of the following:

- Click **More Commands**  > **Manage**  > **Submit to Workflow**.
  - Right-click the diagram in the structure and then click **Submit to Workflow** 
3. (Optional) Select the **Template**, and update the **Name** if necessary.

The name of the workflow is automatically populated. Workflow templates are specific to your enterprise as the tasks included are specific to your business needs. Depending on your business conventions, you may be required to change the name before submitting.

4. (Optional) Click **Assignments** to assign tasks for reviewing and signing off on the parameter definition.
5. Select a group of users in the **Assignment List** or add individual users or a resource pool to the **Assignee** list.
6. Click **Submit**.

Active Workspace sends the parameter definition to the selected workflow.

When all approvers have signed off the parameter definition, the system sends a message to your inbox. You can also check the status of a parameter definition at any time by clicking the **Overview** tab and looking at the **Release Status** property; if the parameter definition is ready for use, this property contains **Approved** or other release status as defined by the system administrator.

## Set objectives on output parameters


You can set an objective option for output parameters:

- Minimize
- Maximize
- Minimize Difference
- Maximize Difference

### Prerequisites

A verification request or simulation request must exist to manage the objective for output parameters.

### Procedure

1. Select a verification request or simulation request in the **SCOPE** panel.
2. If the objective columns do not appear in the **OUTPUT PARAMETERS** table, then click **Table Settings**  > **Arrange** to add the columns.

3. Select an **Objective Option** value, and then complete the objective fields.

## Results

The object values are updated:

### Create parameters from a parameter definition

You can create a parameter from an existing parameter definition. The parameter definition must have been released using the relevant workflow.

Note:

A parameter must always be created under a parent, such as a system model block or requirement. You cannot create a standalone parameter.

1. In the diagram, select a block, requirement, target, or connection, and then click **Details > Parameters**.

The **Parameters** table appears.

2. Review the list of parameters already on the object, if any. If additional parameters are required, select the table and click **Add** ⊕.

The **Add** panel appears.

3. In the **Parameter Definition** panel, select a parameter definition, and then click **Add**.

A row showing the parameter and its properties appears.

4. Enter the necessary values of the properties of the new parameter, and then click **Add**.

A row to the table showing the parameter and its properties appears.

### Remove a parameter to adjust a measurement

You can remove parameters that are no longer required for the analysis task.

1. Select the parent analysis request.
2. Click **Details > Parameters**, select one or more parameters to remove, and click **Unuse** -. You must have write access to the parameter instance.

The system checks for impact to associated studies and, if it identifies impact, confirms if you still want to unuse the parameter.

If you proceed, it deletes the parameter and any associated measurements, and also removes them from the impacted studies. If there are no impacted studies, it deletes the parameter and any associated measurements without further notification.

Note:

If you delete the parent directly, any parameters and measurements associated with it are also deleted.

## Test and parameter project configuration relationships

When you create a test from a parameter project, the test uses the parameter project configuration instead of the test configuration. Therefore, if you change the parameter project configuration, this changes the test configuration as well. The reverse is true for changes in the test configuration.

For more information about parameter projects, see [Associating parameters with a project](#).

### Manage parameters for an item

1. Select the item revision (for example, the system block revision), and click **Details > Parameters**.

Active Workspace displays all the parameters associated with the item revision in tabular format.

The parameters shown depend on the properties of the selected item revision and how they are configured to display in the table.

2. (Optional) Use the toolbar to manage the parameters.

### Manage input and output parameters

You can review and update the input and output parameters for the test objects shown in the following table. The table also provides how the parameter displays for each test object.

Test object	Input and output parameters in separate tables?	Display parameter properties for optimization?
Verification request	No	No
Simulation request	Yes	Yes
Study	No	Yes
Run	No	No

Test object	Input and output parameters in separate tables?	Display parameter properties for optimization?
Test request	No	No
Test event	No	No

To view the input and output parameters, open a test package in **Overview** and select a test object from the results panel test package tree. You may need to show the table for the **PARAMETERS** section. The Usage value indicates the Input or Output setting for each parameter.

Note:

The default **Usage** value for a new parameter is **Output**.

Many sections and most tables provide similar actions. For more information, see **General test object panel, table, and chart actions**.

## Create trace links with parameters

You can map one or more parameters on a system block to one or more parameters on other blocks in the diagram.

1. Define an end point for the trace link, by clicking  $\oplus$  and choosing an object from the list or by dragging the end point from elsewhere in Active Workspace.
2. Click **Create** or use the keyboard shortcut Alt+L to create trace link.

Active Workspace creates the trace link and refreshes the **Create Trace Link** panel.

The **Create Trace Link** panel remains open, allowing you to create multiple trace links if required.

## Find where an object is used

You can see which items (such as parent assemblies, contexts, tests, and references) are associated with a selected element such as a system model block, requirement, or parameter.

1. Search for and open an item such as system model block, requirement, or parameter.
2. Click **Where Used**.

Active Workspace displays all objects that use the selected item.

## Synchronize and publish testing results parameters back to system model and requirement parameters

The verification request system allows you to synchronize and publish parameter value information between the verification request and the source parameter. This capability also supports verification requests created from parameter projects. When Active Workspace is embedded in a system modeling product such as System Modeling Workbench (source), you can publish and synchronize the simulation parameter values back to the system model and requirement parameters. Publishing allows you to redefine the existing parameter values with the best parameter values resulting through simulation testing. After defining an optimization study, you can persist the optimization properties, which define the baseline for the next optimization.

You can configure the **PLE\_Publish\_Measurement\_To\_Goal preference** to publish the measured values to the source parameter's goal value field instead of the measured value field. This feature is available for verification request types, not only parameter-project based types.

You can synchronize and publish both input and output parameters. The synchronization is based on the last date the parameter was modified.

The following graphic shows input and output parameters that show a **Status of Unpublished or Unsynchronized**.


Name	Revision	Usage	Result	Measurement	Goal	Min	Max	Units	Status
Brake Initiate Sp...	A	Output	Pass	54	40	10	60	each	Unpublished
Braking Distance	A	Output	Pass	192	200	175	201	each	Unpublished
Drum Diameter	A	Input			15	10	25	each	Unpublished
Final Distance	A	Output			40	15	55	each	Unsynchronized
Force	A	Input			17	11	21	each	Unpublished
Power	A	Output			15	5	25	each	Unsynchronized
Speed	A	Input			14	2	55	each	Unsynchronized

In the procedure, we use the term *source* to refer to the application that is synchronizing or publishing to from Active Workspace, such as System Modeling Workbench.

### Procedure

1. Select one or more parameter table rows that you want to either synchronize or publish.
2. Do one of the following:
  - Click **Synchronize**.

Active Workspace refreshes the selected parameters with the corresponding values from the source (outside system) parameters.

- Click **Publish**  and select one of the following options:
  - **To measured value with goal min/max:** Publish the measured value **Goal Min** and **Max** values to the source.
  - **To measured value only:** Publish the **Measurement** parameter value to the **Measurement** of the source.
  - **To goal with min/max:** Publish the **Min** and **Max** values to the source **Goal Min** and **Max** values.
  - **To goal only:** Publish the measured valued to the **Goal** of the source.

Active Workspace indicates that the outside system (source) parameters are updated.

## Results

Active Workspace either synchronizes or publishes the selected parameters back to the source. The **Status** changes from either **Unsynchronized** or **Unpublished** to blank.

## Managing and monitoring test packages

### Managing test management workflows

#### Understanding the test workflow

If you use the standard test workflow, the ability to edit it is restricted by the current state as follows:

	Requesting Engineer					Performing Engineer				
	Sync	Publish	Toggle I/O	Unuse	Add to VR	Sync	Publish	Toggle I/O	Unuse	Add to VR
Authoring	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Submitted	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Analyzing	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Analyzed	no	Yes	no	no	no	no	Yes	no	no	no
Publishing	no	Yes	no	no	no	no	Yes	no	no	no
Completed	no	Yes	no	no	no	no	Yes	no	no	no

By default, the system does not lock the domain structure. Customer sites may implement a locking mechanism that best fits their business process, for example, baselining or revision rule release status.

At most sites, the responsible role submits the test to a workflow, which moves the test to the next state. However, at some sites, you may have to move it to the next state manually.

If issues are found during the **Analyzing** state, the simulation engineer may reject the workflow. If this occurs, the test reverts to the **Authoring** state, allowing the simulation engineer to resolve the issues and then submit the test for another review.

**Note:**

Although access to edit the test is limited by the state, the general access rules for the user's group and role also apply. The user must *also* be in a group and role compatible with the author's group and role to change any values in the test.

## Running test management workflows

The solution provides the following workflows.

### Test management workflows

Workflow Name	Purpose	Participants	Initiator
Simplified Test Management Workflow	A three-step workflow with the following phases: <b>Start</b> , <b>Analyze</b> , and <b>Published</b> . The user needs to assign the "Simulation Engineer" before starting the Workflow. If assigned before starting the workflow, then the first task will go to "Simulation Engineer;" otherwise, the first task goes to the "Systems Engineer".	Any	
Verification Request State Workflow	Full verification workflow with unlocked editing for parameter and BOM elements until the <b>Published</b> state for the <b>Verification Request State Workflow</b> .		

	Requesting Engineer					Performing Engineer				
	Sync	Publish	Toggle I/O	Unuse	Add to VR	Sync	Publish	Toggle I/O	Unuse	Add to VR
Authoring	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Submitted	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Analyzing	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Analyzed	no	Yes	no	no	no	no	Yes	no	no	no
Publishing	no	Yes	no	no	no	no	Yes	no	no	no
Completed	no	Yes	no	no	no	no	Yes	no	no	no

Workflow Name	Purpose	Participants	Initiator
VM Initiate Verification Activity Workflow	Creates a test, such as a test request or an analysis request. The type of test is based on the verification method assigned to the related requirements.	Verification Planner, Design Engineer	Test Planner
Sample Structure Release Workflow	Provides a sample workflow to release the product structure of a product EBOM and the requirement structure.	Any	Any
VM Sample Document Release Workflow	Provides a sample workflow to release a technical document revision and its part list revisions.	Any	Any
VM Transfer Test EBOM Ownership Workflow	Transfers the ownership for all items of an occurrence when the occurrence type is marked as a test item. The associated workflow handler is <b>IAV_attach_test_item</b> .	-	-
VM Test EBOM Release Workflow	Releases all test nodes generated for a process test EBOM.	Design Engineer, Test Engineer	Design Engineer, Test Engineer
VM Test Procedure Approval WF	Invokes a <b>IAV_check_measurement_req_instrument_relation</b> rule handler that ensures that the test procedure is related to a test request. The measurement requirement is related to the test EBOM nodes. The released configuration of a test EBOM is specified in the structure context, and releases the test procedure and the structure. A <b>IAV_disallow_ctxt_revrule_clause</b> may also be used with this workflow.	-	-
VM Test Request Approval Workflow	Ensures that the collaboration context of a product EBOM and a test EBOM configuration are related to a test request. It also ensures the test request is related to a verification object.	Design Engineer, Test Engineer	Design Engineer

Workflow Name	Purpose	Participants	Initiator
	If you are a reviewer, you must approve the assigned tasks to set the release status. This workflow is linked to the <b>VM Test Request Signoff</b> workflow.		
VM Test Request Signoff Workflow	Assigns the approved test request to a test engineer. As a test engineer, you work on the test request and provide the resulting data to a design engineer for signoff. A subprocess from the <b>VM Test Request Approval</b> workflow activates this workflow.	Design Engineer, Test Engineer	Design Engineer

## Create a test or verification request

You can create a test or verification request from both BOM and non-BOM items such as requirements, PDFs, or system model blocks. You can also create a test or verification request from a parameter project.

You can create a test or a verification request with or without a predefined test definition. Using a test definition as a template in this way ensures that the system can enforce rules for permitted elements, modeling, parameterization, and the capture of results. A test is valid only in the context of a single domain. If you do not use a test definition, the system designer can include any element in the test without restriction.

### Restrictions and limitations

On the **Relations** tab, the children of occurrences do not appear as the endpoint when that occurrence is expanded.

### Procedure

1. Perform either of the following:
  - Open a model diagram and select one or more elements.
  - Select one or more elements in the structure.
2. Click **More commands ... > New > Create Test**.

The **Create Test** panel appears.

3. From the **TYPE** list, select **Test** or **Verification Request**.

4. Enter a name for the test or verification request.
5. (Optional) Select an **Inclusion Rule**.
6. Enter a name for the test.
7. (Optional) Select the **Open on create** check box to open the test or verification request upon creation.
8. Click **Create**.

Active Workspace creates the verification or test request object. For information on managing test requests, see [Manage test requests](#). For information on managing verification requests, see [Manage verification requests](#).

## Create a verification or test request in context

You can create a verification or test request directly against selected objects, such as requirements.

### Prerequisites

You must have existing objects, such as requirements, to create the requests against.

### Procedure

1. Locate and select one or more objects in the search results panel that you want to create the verification or test request against.
2. Click **More commands ...** > **New** ✨ > **Create Test**.

The **Create Test** panel appears.

3. In the **Type** field, select a request type, such as **Verification Request**.
4. Complete the dialog box, and then click **Create**.

### Results

Active Workspace creates the verification or test request with the selected context objects included in the request.

## Managing worksets

### Context navigation overview

The working context of a test consists of a collection of artifacts that verification management evaluates to assign a status of pass or fail. These artifacts may include items such as requirements, system designs, 3D designs, and so on.

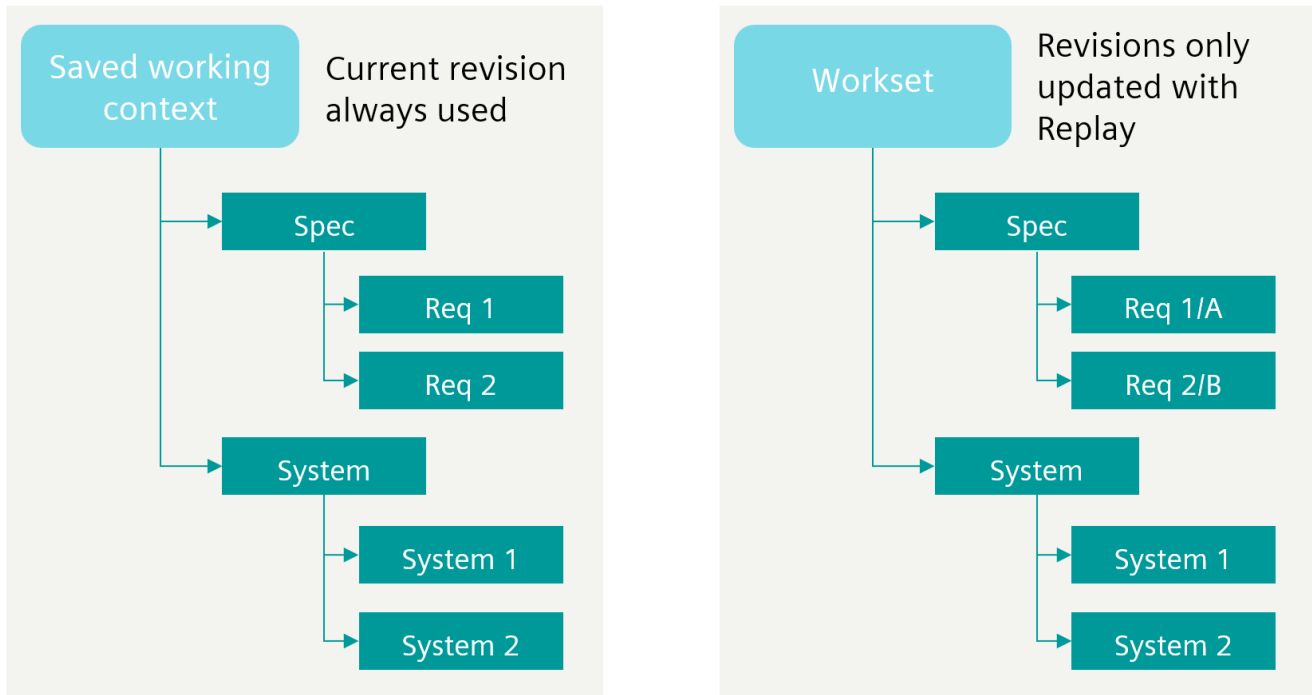
The background working context is your current environment. It is not necessary to save your current background working context before you exit, because you always return to your previous background context when you reopen the same content.

To preserve your environment for later retrieval or sharing, you must save the context.

After the context has been saved, you can navigate through artifacts contained in that context using the following functionalities:

- **Saved working context** - This functionality supports a dynamic state of verification where the current object revision is always used. The saved working context functionality is an Active Workspace construct where each of the artifacts added to the verification request is kept at the top line.
- **Workset** - This functionality supports a static state of verification where the same object revision is used unless a different revision is selected using the **Replay** button. The workset functionality is a platform construct where each of the artifacts added to the verification request is kept under a workset instance.

This image illustrates the differences between the saved working context and workset functionalities.



You can use the preference `PLE_Is_Create_VerificationRequest_With_Workset` to set which functionality you use to navigate the context of a test. If the preference is set to **false**, the verification request is created with the saved working context functionality. If the preference is set to **true**, the verification request is created with the workset functionality. See [Set preferences for test management](#) for information about the preferences used to configure test management.

### Navigate with the saved working context

The saved working context functionality supports a dynamic state of verification where the current object revision is always used when a verification request is processed.

### Restrictions and limitations

- The state of the information used for verification is dynamic. Verification is always performed against the latest version.

### Prerequisites

The preference `PLE_Is_Create_VerificationRequest_With_Workset` must be set to **false** to use the saved working context functionality. See [Set preferences for test management](#) for information on preferences used to configure test management.

### Procedure

1. Open the verification request and click **Manage Context**.

The **Element** panel opens.

2. Select one or more items. For example, select a system model, system block, or requirement. If you have nested items such as a child system block, expand the list until you reach the correct level.

The panel displays all items related to the selected context. If the context contains many items, you can search in context to find the item you require.

3. (Optional) Do either of the following:
  - a. Click **Details > Architecture** to view the diagram of the selected item. You can use many of the **standard commands** to navigate the diagram.
  - b. Click **Details > Documentation** to view specification information for the selected item.

## Navigate with the workset

The workset functionality supports a static state of verification where a verification request is always processed against the same revision of an artifact, unless a different revision is selected.

See the *Structure Management on Active Workspace – Usage* guide for information about worksets.

## Restrictions and limitations

- You must use **Replay** to select a new revision of the artifact for verification. See *Structure Management on Active Workspace — Usage* for information about using the **Replay** button to update a workset.
- Once a new artifact revision is selected for verification, you cannot return to the previously selected revision. Ensure that you want to make this change before proceeding.

## Prerequisites

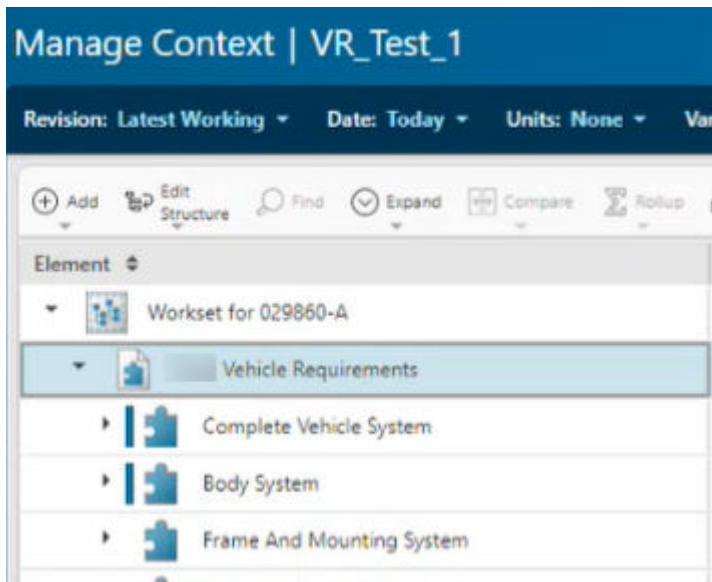
The preference `PLE_Is_Create_VerificationRequest_With_Workset` must be set to **true** to use the saved working context functionality.

See **Set preferences for test management** for information about preferences used to configure test management.

## Procedure

1. Open the verification request and click **Manage Context**.

The **Element** panel opens and the workset appears.



2. Select one or more items. For example, select a system model, system block, or requirement. If you have nested items, such as a child system block, expand the list until you reach the correct level.

The panel displays all items related to the selected context. If the context contains many items, you can search in context to find the item you require.

3. (Optional) If **Replay** appears for your context, one or more of the items has a newer revision. Click **Replay** to update your context to use the new revision.

Once a new artifact revision is selected for verification, you cannot go return to the previously selected revision. Ensure that you want to make this change before proceeding.

4. (Optional) Do either of the following:
  - a. Click **Details > Architecture** to view the diagram of the selected item. You can use many of the **standard commands** to navigate the diagram.
  - b. Click **Details > Documentation** to view specification information for the selected item.

## Manage test packages and test objects

### Test package objects

You can **create** test package components include the following:

**Verification Request** Verification requests provide a high level of detail for tracking testing.

- Simulation Request** Simulation requests are for cases where a team wants to plan and manage a group of simulation test types for a given system. These requests let you plan the simulation tests needed for a system under the test or verification request, document the multiple simulation tests, and even assign them to different teams. You can plan by simple assignment, like email, or controlled through a workflow. The request has a sample workflow that you can adapt to your needs. You can also include runs with simulation requests to allow for iterations.
- Run** Runs let you perform multiple testing iterations for a primary design to prove all scenarios are valid—whether with parameter value changes or with larger changes to the system, model, or design. The run keeps record of the changes as well as the test results, all in the scope of the larger test package. Use runs for scenarios such as:
- When there are many scenarios and breath-of-scope testing is required.
  - When the team is iterating to solve test failures.
  - When you need daily or periodic sanity test runs to ensure determine if changes impact the test.
- Study** Studies are for design trade off analysis and allows you to document the unique differences of each study. The study includes the test results and allows you to compare the results of different designs. You can also include runs with studies to allow for iterations.
- Test Request** Test requests are peer to verification requests, but do provide the detail data tracking that verification requests require.
- Test Event** The runs for a physical test request are called test events. Test events are designed to specifically support test instrument data, including channel pickups. These events also support physical part BOM management, which can include the instruments, fixtures, and resource parts.

## Create test package objects

You create the test package by adding test package objects. Consider the following when creating test package objects:

- This procedure assumes you have already **created a test request or verification request**.
- You can create test and verification requests with or without a predefined test definition. Using a test definition as a template in this way ensures that the system can enforce rules for permitted elements, modeling, parameterization, and results. A test is valid only in the context of a single domain. If you do not use a test definition, the system designer can include any element in the test without restriction.
- You can create tests from both BOM and non-BOM items such as requirements, PDFs, or system model blocks. You can also create a test from a **parameter project**.

- By creating an object as a **Child** or **Sibling**, Active Workspace automatically selects the appropriate **TYPE** in the **Add Child** dialog box. For example, if you add a child to a **Test Request**, the **TYPE** defaults to **TEST EVENT**.

1. Select one or more elements in the Results panel.
2. Click **Add** ⊕ **Child** or **Sibling**.

The **Add Child** or **Add Sibling** panel displays, showing the appropriate **TYPE** as a child for the selected parent object.

3. Enter the remaining properties and any owning project, and then click **Create**.

Active Workspace creates the test object and displays the **Overview**.

### Add elements to a test

You can add elements to the test package. In most cases, you can add non-structure elements such as generic items, revisions, datasets, PDFs, and Microsoft files. This allows the test engineer to understand what to simulate and assess during testing. You cannot add elements to the test if it is already in the workflow, that is, not in the **Author** state.

You can add elements from the context from within the test. However, you cannot add or remove *attributes* from an element from within the test.

### Add non-occurrence elements to a test

1. Open the test in the **Overview**. In a panel such as **REQUIREMENTS** or **TEST CASES**, click **Add** ⊕ and then select **Add**.

Active Workspace displays the **Add** panel. The panel defaults to the specific **Type**, such as a requirement or a test case.

2. Complete the dialog box, and then click **Add**.

### Add occurrences from content

1. Open the test and, in a panel such as **REQUIREMENTS** or **TEST CASES**, click **Add From Content** ⊕.

Active Workspace automatically changes to **Content** and displays the **Add From Content** panel.

2. Drag content objects to the dialog box for inclusion in the test, and then click **Add**.

**Note:**

To avoid duplicate objects, if an object already exists in the test, then Active Workspace does not add the element to the panel.

## Add parameters to a test

You can copy and paste parameters directly from the **PARAMETERS** panel to the **TEST CASES** panel.


1. In the **PARAMETERS** table, select one or more parameters, and then click **Copy**.
2. Click **Paste Parameters**.

## Remove an element from a test

You can remove elements, such as requirements or test cases, that are associated with a test.

**Note:**

- If the test is already in the workflow, that is, not in the **Author** state, you cannot remove elements from it.
- In this procedure you are only removing elements from the test. You are not deleting the element from Active Workspace.

1. Open the test in **Overview**.
2. In a panel, such as **REQUIREMENTS** or **TEST CASES**, select one or more elements. and then click **Remove from Verification Request** .

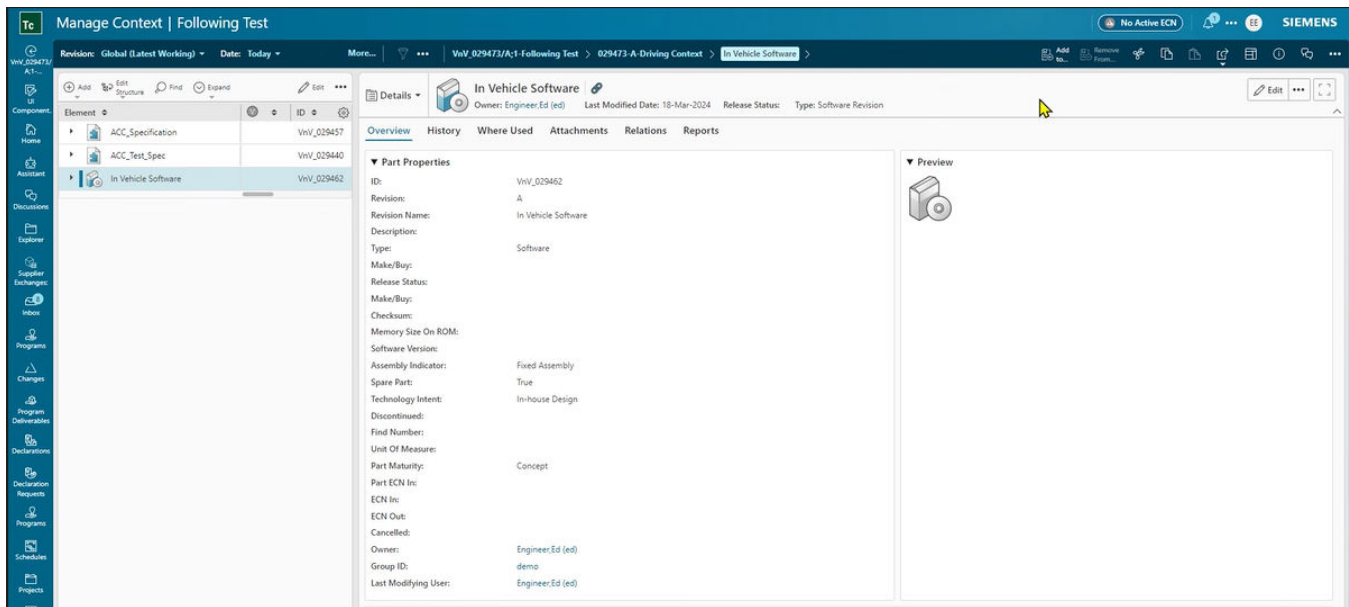
## Navigate the context of a test

You can navigate to the saved context of a test to understand what information to apply to the model related to the test.

**Note:**

If you do not have access to the saved context, you cannot view the content.

The following graphic shows an example of saved context for a selected test.



1. Open the test request and click **Manage Context**.

The **Element** panel appears.

2. Select one or more items in the **Element** panel. For example, select a system model, system block, or requirement. If you have nested items such as a child system block, expand the list until you reach the correct level.

The list includes all items related to the selected context. If the context contains many items, you can search in context to find the item you require.

3. (Optional) Do either of the following:
  - Click **Details > Architecture** to view the diagram of the selected item. You can use many of the **standard commands** to navigate the diagram.
  - Click **Details > Documentation** to view specification information for the selected item.

### Create a test package against a requirement

You can create and associate a test package directly against a requirement.

1. Select a requirement in the results panel.
2. Click **More commands ... > New > Create Test**.

The **Create Test** panel displays.

3. Complete the panel and click **Create**.

The test opens in the **Overview** and the associated requirement appears in the **REQUIREMENTS** panel.

## Create a test package against a test request

You can create and associate a test package directly against selected test requests.

1. Select one or more test requests in the results panel or in **Architecture**.
2. Click **More commands ... > New > Create Test**.

The **Create Test** panel appears.

3. Complete the panel and click **Create**.

The test opens in the **Overview** and the associated test package appears in the **SCOPE** panel.

## Manage verification requests

You can review and update the verification request details in the **Overview**. The page provides a snapshot of the test package data.

Open a test request and then click the **Overview**.

The following sections provide an overview of the actions you can perform and descriptions of the panels available on the **Overview**.

### Overview

The following are panels available in the **Overview** tab.

#### General panel and table actions

Many sections and most tables provide similar actions. For more information, see [General test object panel, table, and chart actions](#).

#### Results panel test package tree

The results panel displays the test package object tree, where you can select the scope of your data focus. For more information, see [Results panel test package tree](#).

## SUMMARY

Provides an overview of the verification request that you selected in the results panel test package tree.

Click **Edit Localization** to open the **Edit Localization** dialog box and update the **Name** and **Description** field text into alternate languages.

### TEST RESULTS

Provides a summary of the **Pass** or **Fail** test results. Test results propagate from child to parent. If all children have a **Pass** result, then the parent results in a **Pass** as well. If any child has a **Fail** result, then the parent results in a **Fail**.

The underlying data on which the test results are based is provided in the **REQUIREMENTS** panel table. Click the graphic to filter the **REQUIREMENTS** panel table based on **No Result**, **Requirements Fail**, or **Requirements Pass** result.

Click **Add** ⊕ to add a study, test request, run, or simulation request. For more information, see Create test package objects.

For more information, see [General test object panel, table, and chart actions](#).

### PROGRAM EVENTS

Provides the event milestones to which the verification request is due.

Click **Add** ⊕ to add a milestone.

For more information about programs and milestones, see Program Planning .

### REQUIREMENTS

Provides a manageable list of requirements related to the verification request, the **Result** status, and a chart to filter the data.

For more information, see [General test object panel, table, and chart actions](#).

### TEST CASES

Provides a manageable list of test cases associated with the verification request.

### TEST PROCEDURES

Provides a manageable list of test procedures associated with the verification request.

### FUNCTIONS

Provides a manageable list of functions associated with the verification request.

## SYSTEMS

Provides a manageable list of systems associated with the verification request.

## PARTS

Provides a manageable list of parts associated with the verification request.

## SIMULATION MODELS

Provides a manageable list of simulation models associated with the verification request. For more information, see [MDAO simulation in Active Workspace](#).

## PRODUCT AND TEST EBOMS

Provides a manageable list of product and test engineering BOMs (EBOMs) associated with the verification request. For more information, see Engineering BOM Management on Active Workspace.

## SOFTWARE

Provides a manageable list of software used for testing.

## PARAMETERS

Provides a manageable list of parameters associated with the verification request.

For more information about parameter tables, see [Parameter table actions](#).

## OTHERS

Provides a manageable list of any objects added to the test that are not included in other tables.

## REPORTS

Provides a manageable list of reports.

### Note:

If there are custom objects such as sub-types created by the end user, a new preference with appropriate values must also be created for each sub-type. The new preference name must then be added as a value to the preference called **AWC\_StartupPreferences**.

For example, if the end user has created a sub-type called *LHM3TestRun* for the preference **IAV0TestRun**, then a new preference called *AWC\_LHM3TestRun\_DisplayTables* must be created and added to **AWC\_StartupPreferences**.

See Enabling your custom preferences in Active Workspace for information on the **AWC\_StartupPreferences** preference.

### Trends

For more information, see [Trends tab](#).

### Participants

For more information, see [Manage test participants](#).

### Manage studies

You can review and update the test events in the **Overview**, add instruments and calibration status in the **Test Setup**, and add test event users in the **Participants**.

Open a test request, select an **existing test event**, and then click the **Overview**.

The following provides an overview of the actions you can perform and descriptions of the panels available on the test event.

### General panel and table actions



Many sections and most tables provide similar actions. For more information, see [General test object panel and table actions](#).

### Results panel test package tree

The results panel displays the test package object tree, where you can select the scope of your data focus. For more information, see [Results panel test package tree](#).

### SUMMARY

Provides an overview of the study or test event that you selected in the Test package tree.

Click **Edit**  to edit the values such as **Result**, **Pass**, or **Fail**. When you are done, click **Save** .

### TEST RESULTS

Provides a summary of the **Pass** or **Fail** test results. Test results propagate from child to parent. If all children have a **Pass** result, then the parent results in a **Pass** as well. If any child has a **Fail** result, then the parent results in a **Fail**.

The underlying data on which the test results are based is provided in the **REQUIREMENTS** panel table. Click on the graphic to filter the **REQUIREMENTS** panel table based on **No Result**, **Requirements Fail**, or **Requirements Pass** result.

Click **Add** ⊕ to add a study, test request, run, or simulation request. For more information, see Add elements to a test.

## PROGRAM EVENTS

Provides the event milestones for which the test is due.

Click **Add** ⊕ to add a milestone.

For more information about programs and milestones, see Program Planning .

## Execute Test

Click **Execute Test** to run the test and open the [Execute Test page](#).

## REQUIREMENTS

Provides a manageable list of requirements related to the study, the **Result** status, and a **chart** to filter the data.

For more information, see [General test object panel, table, and chart actions](#).

## TEST CASES

Provides a manageable list of test cases related to the study.

## PARTS

Provides a manageable list of parts related to the study.

## SIMULATION MODELS

Provides a manageable list of simulation models associated with the verification request. For more information, see [MDAO simulation in Active Workspace](#).

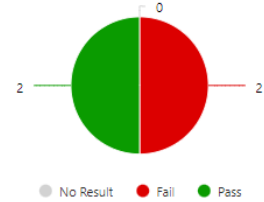
## SOFTWARE

Provides a manageable list of software used for testing.

## PARAMETERS

Provides a manageable list of parameters associated with the study.

Name	Revision	Usage	Result	Measure...	Goal	Min	Max	Units	Source	Status
Ambient Humidit...	A	Input			20		100		Test Condi...	Unpublish...
Ambient Humidity (%)	A	Input			20		100		Boundary ...	
Ambient Temp (F)	A	Input			10		85		Test Condi...	Unpublish...
Ambient Temp (F)	A	Input			10		85		Boundary ...	
Do Not Exceed Max N...	A	Output	Fail	91			90	Db	NVH Brak...	Unpublish...
Drag Percentage	A	Input					78		Test Condi...	Unpublish...
Drag Percentage	A	Input					78		Boundary ...	Unpublish...
Fade Front	A	Output	Fail	565			450		NVH Brak...	Unpublish...
Fade Rear	A	Output	Pass	238			250		NVH Brak...	Unpublish...
Hex element average ...	A	Input			10				Mesh:	Unpublish...
Operational Max Noise	A	Output	Pass	68			70	Db	NVH Brak...	Unpublish...
Pressure Range	A	Input				0	51		Test Condi...	
Pressure Range	A	Input				0	51		Boundary ...	



For more information about parameter tables, see [Parameter table actions](#).

## Runs

This area provides information about the runs related to the study. For more information, see [Execute a test \(run\)](#).

### Create a study from an existing study

You can duplicate an existing study into a new independent study. Parameter values and mappings are copied into the new study, except for the following values, which are cleared:

- Measured value
- Multiple result table
- Min measured and Max measured

### Prerequisites

You must have an [existing study](#) to copy from.

### Procedure

1. Open a test package and select the test at the top of the tree.

2. Click **Add**, and select the study **Type**.
3. Enter a **Name**.
4. The **POPULATION** field populates automatically with the selected study, but you can select a different study from the dropdown.

## Results

Active Workspace creates the new study that is populated with parameter values and mappings from the source study.

## Manage test requests

You can review and update the test requests shown in the **Overview** with panels that provide a snapshot of the test request data.

Open a test request and then click the **Overview**.

## Overview

The following are panels available in the **Overview**.

### General panel and table actions

Many sections and most tables provide similar actions. For more information, see [General test object panel and table actions](#).

### Results panel test package tree

The results panel displays the test package object tree, where you can select the scope of your data focus. For more information, see [Results panel test package tree](#).

## SUMMARY

Provides an overview of the study, test, run, and simulation request that you selected in the **SCOPE** panel.

Click **Edit**  to edit the values such as **Result**, **Pass**, or **Fail**. When you are done, click **Save** .

## TEST RESULTS

Provides a summary of the **Pass** or **Fail** test results. Test results propagate up from child to parent. If all children have a **Pass** result, then the parent results in a **Pass** as well. If any child has a **Fail** result, then the parent results in a **Fail**.

The underlying data on which the test results are based are provided in the **REQUIREMENTS** panel table. Click on the graphic to filter the **REQUIREMENTS** panel table based on **No Result**, **Requirements Fail**, or **Requirements Pass** result.

Click **Add** ⊕ to add a study, test request, run, or simulation request. For more information, see Add elements to a test.

### PROGRAM EVENTS

Provides the event milestones to which the test for which the test is due.

Click **Add** ⊕ to add a milestone.

For more information about programs and milestones, see Program Planning.

### REQUIREMENTS

Provides a manageable list of requirements related to the test request, the **Result** status, and a **chart** to filter the data.

For more information, see [General test object panel, table, and chart actions](#).

### TEST PROCEDURES

Provides a manageable list of test procedures.

### PARTS

Provides a manageable list of parts associated with the test request.

### TEST METHODS

Provides a manageable list of test methods associated with the test request.

### PRODUCT AND TEST EBOMS

Provides a manageable list of product and test engineering BOMs (EBOMs) associated with the test request. For more information, see Engineering BOM Management on Active Workspace.

### PHYSICAL BOM

Provides a manageable list of physical BOMs related to the test request.

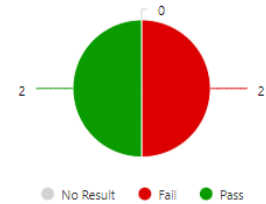
## PHYSICAL TEST RESULTS

Provides a manageable list of test results related to the test request.

## PARAMETERS

Provides a manageable list of parameters associated with the test request.

PARAMETERS										
Name	Revision	Usage	Result	Measure...	Goal	Min	Max	Units	Source	Status
Ambient Humidit...	A	Input			20		100		Test Condi...	Unpublish...
Ambient Humidity (%)	A	Input			20		100		Boundary ...	
Ambient Temp (F)	A	Input			10		85		Test Condi...	Unpublish...
Ambient Temp (F)	A	Input			10		85		Boundary ...	
Do Not Exceed Max N...	A	Output	Fail	91			90	Db	NVH Brak...	Unpublish...
Drag Percentage	A	Input					78		Test Condi...	Unpublish...
Drag Percentage	A	Input					78		Boundary ...	Unpublish...
Fade Front	A	Output	Fail	565			450		NVH Brak...	Unpublish...
Fade Rear	A	Output	Pass	238			250		NVH Brak...	Unpublish...
Hex element average ...	A	Input			10				Mesh:	Unpublish...
Operational Max Noise	A	Output	Pass	68			70	Db	NVH Brak...	Unpublish...
Pressure Range	A	Input				0	51		Test Condi...	
Pressure Range	A	Input				0	51		Boundary ...	



For more information about parameter tables, see [Parameter table actions](#).

## Test Events

For more information, see [Manage test events](#).

## Participants

For more information, see [Manage test participants](#).

## Manage test events

You can review and update the test events in the **Overview**, add instruments and calibration status in the **Test Setup**, and add test event users in the **Participants**.

Open a test request, select an **existing test event**, and then click the **Overview**.

## Overview

The following are panels available in the **Overview**.

## General panel and table actions



Many sections and most tables provide similar actions. For more information, see [General test object panel and table actions](#).

## Results panel test package tree

The results panel displays the test package object tree, where you can select the scope of your data focus. For more information, see [Results panel test package tree](#).

## SUMMARY


Provides an overview of the study or test event that you selected in the Test package tree.

Click **Edit**  to edit the values, such as **Result**, **Pass**, or **Fail**. When you are done, click **Save** .

## TEST RESULTS


Provides a summary of the **Pass** or **Fail** test results. Test results propagate from child to parent. If all children have a **Pass** result, then the parent results in a **Pass** as well. If any child has a **Fail** result, then the parent results in a **Fail**.

The underlying data on which the test results is based are provided in the **REQUIREMENTS** panel table. Click the graphic to filter the **REQUIREMENTS** panel table based on **No Result**, **Requirements Fail**, or **Requirements Pass** result.

Click **Add**  to add a study, test request, run, or simulation request. For more information, see [Add elements to a test](#).

## PROGRAM EVENTS

Provides the event milestones for which the test is due.

Click **Add**  to add a milestone.

For more information about programs and milestones, see [Program Planning](#).

## TEST PROCEDURES

Provides a manageable list of test procedures.

## PARTS

Provides a manageable list of parts associated with the test event.

## TEST METHODS

Provides a manageable list of test methods associated with the test event.

## PRODUCT AND TEST EBOMS

Provides a manageable list of product and test engineering BOMs (EBOMs) associated with the test event. For more information, see [Engineering BOM Management on Active Workspace](#).

## PHYSICAL BOM

Provides a manageable list of physical BOMs related to the test event.

## PHYSICAL TEST RESULTS

Provides a manageable list of test results related to the test event.

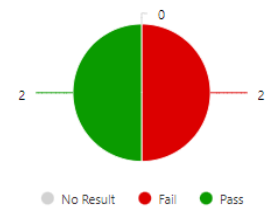
## SOFTWARE

Provides a manageable list of software used for testing.

## PARAMETERS

Provides a manageable list of parameters associated with the test event.

PARAMETERS										
Name	Revision	Usage	Result	Measure...	Goal	Min	Max	Units	Source	Status
Ambient Humidit...	A	Input		20	100				Test Condi...	Unpublish...
Ambient Humidity (%)	A	Input		20	100				Boundary ...	
Ambient Temp (F)	A	Input		10	85				Test Condi...	Unpublish...
Ambient Temp (F)	A	Input		10	85				Boundary ...	
Do Not Exceed Max N...	A	Output	Fail	91	90		Db		NVH Brak...	Unpublish...
Drag Percentage	A	Input			78				Test Condi...	Unpublish...
Drag Percentage	A	Input			78				Boundary ...	Unpublish...
Fade Front	A	Output	Fail	565	450				NVH Brak...	Unpublish...
Fade Rear	A	Output	Pass	238	250				NVH Brak...	Unpublish...
Hex element average ...	A	Input		10					Mesh:	Unpublish...
Operational Max Noise	A	Output	Pass	68	70		Db		NVH Brak...	Unpublish...
Pressure Range	A	Input			0	51			Test Condi...	
Pressure Range	A	Input			0	51			Boundary ...	



For more information about parameter tables, see [Parameter table actions](#).

## Test Setup

Provides the **INSTRUMENT STATUS** table that allows you to manage the physical testing instruments and track their calibration date and status.

## Participants

For more information, see [Manage test participants](#).

## Add or remove requirements from a test package

You can add or delete a requirement from a test package.

### Add requirements to a test package

1. Open the test package in the **Overview**.
2. In the **REQUIREMENTS** panel, click **Add > Add from Content**.

A task-based activity starts and you are returned to **Content**. The **Add From Content** panel appears.

This panel allows you to drag requirements from the specification to the panel.

3. Drag one or more requirements from the Results panel to the **Add From Content** panel.
4. Click **Add**.

▼ TEST CASES

Element Name	Revision	Description	Result	Related Objects	Target	Owner	Type	Test Ste...
▼ Acc System Target Vehicl...	A			REQ-001055/A;1-Maintain Following Di...	True	Ed Engineer (ed)	Test Case Revisi...	
Step 1- Gradual Slow...	A				True	Ed Engineer (ed)	Test Step Revisi...	
Step 2- Soft Braking	A				True	Ed Engineer (ed)	Test Step Revisi...	
▶ Step 3- Maintain slow...	A				True	Ed Engineer (ed)	Test Step Revisi...	

You are returned to the test package **Overview** and the requirements appear as **Related Objects** in the **TEST CASES** panel.

### Remove requirements from a test package

1. Open the test package in the **Overview**.

- In the **REQUIREMENTS** panel, select one or more requirements, and then click **Remove From Verification Request**.

A confirmation appears and indicates that associated parameters will also be removed.

- Click **Remove** to confirm.

## Common test management screen actions, panels, tables, and charts

### Results panel test package tree

Provides a dynamic tree hierarchy of the verification and test requests, studies, runs, and simulation requests for which the details are compiled and provided in other panels on the page. The objects are color-coded to indicate **Pass** (green) or **Fail** (red) status.

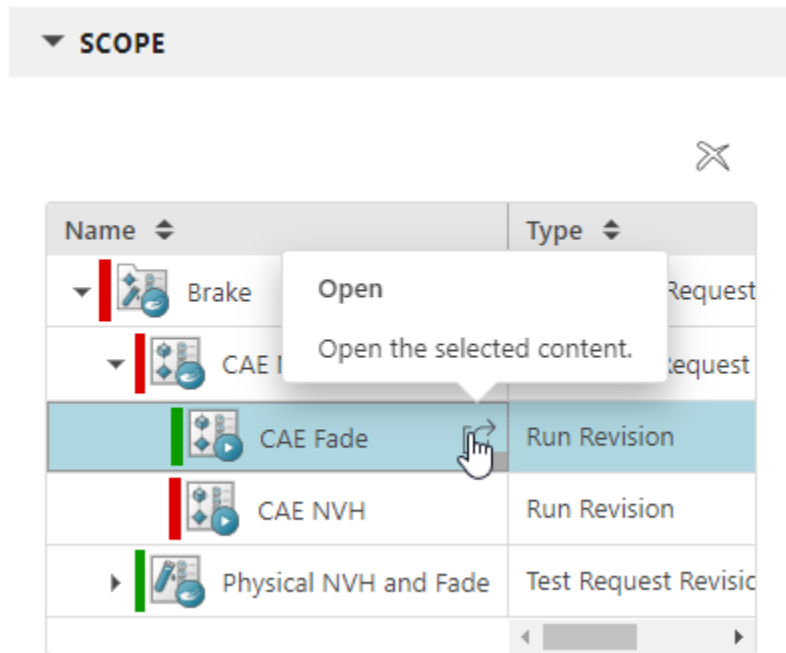
Click **Add** (+) to add a study, test request, test event, or simulation request. For more information, see Create test package objects.

The following graphic shows a typical **SCOPE** panel showing the test package tree hierarchy

Name	Type
▼ Brake	Verification Request Revision
▼ CAE NVH and Fade	Study Revision
CAE Fade	Run Revision
CAE NVH	Run Revision
▶ Physical NVH and Fade	Test Request Revision

To open a specific element with any tab (such as **Content**, **Trends**, and so on), open the item, then click the tab that you want to view.



The following graphic shows opening the **CAE Fade Run Revision** item.





The **CAE Fade** item opens in the **Overview** tab with the scope adjusted to the selection. You can now select any tab to show data related to directly to the **CAE Fade** item.

### General test object panel, table, and chart actions


Many test package object tables provide similar features and actions. If available, you can perform the following:

- For general information about working with tables, see Arrange, wrap text, and save the order of columns .
- Click **Preview**  to display the object associated with the table in a pop-up window. For example, preview the requirement, simulation model, or reports.
- Click **Add**  or **Add from Content** to either add elements, or to add content from occurrences. For more information, see Add elements to a test.

If you try to add invalid objects to a test, a pop-up message lists the invalid objects which were not added.

- Click **Start Edit**  to edit table fields. If a field is not already editable, double-click to edit. Click the field again to display a list of values (if available). When you are done, click **Save Edits** .
- You can adjust the dividers between panels to minimize or maximize graphs, tables, and other areas.

## Tables


Click **Show Tables**  to toggle a list of available to tables to add or remove to the page.

Tip:

To manage the table columns, click **Table Settings**  > **Arrange**. For more information, see [Modifying table and tree appearances](#).

## Charts

You can perform the following actions with pie charts:

- Click **Show Charts**  to toggle chart display.
- Set a **Target** table row value to **True** to include that row data the pie chart.
- Click the pie chart **TEST RESULTS** pie chart at the top to filter *all* table data by test result status **No Result**, **Fail**, or **Pass**.
- Click the pie chart to the right of a table to filter that table list.
- Click a pie chart label to toggle its display in the pie chart.

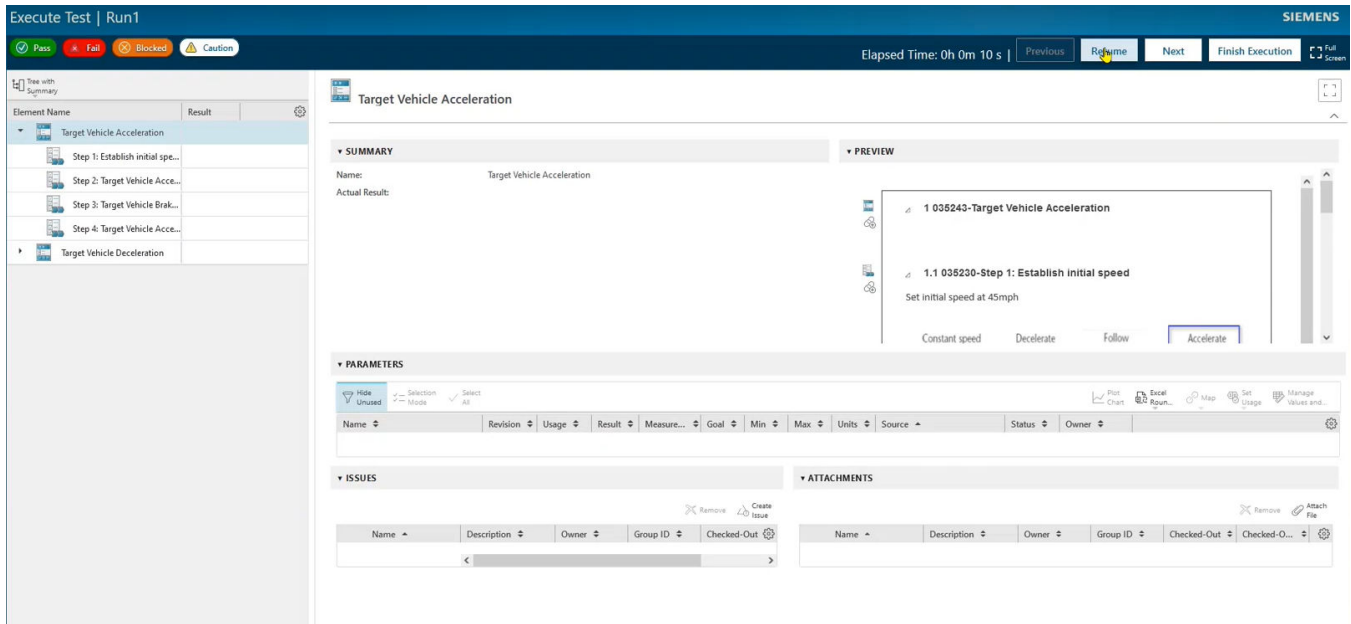
## Execute Test

Click **Execute Test**  to open the [Execute Test page](#) to start a run.

## Manage issues

You can create and manage issues when a test fails. The issue table is available with failed test runs, test cases, and test steps.


The following graphic shows the **ISSUES** table on the **Execute Test** page.



Click **Create Issue** to begin opening the change management issue.

## Parameter table actions

You can perform the following actions (if available) with the **PARAMETERS** table (which includes the **INPUT PARAMETERS** and **OUTPUT PARAMETERS** tables):

- Click **Quick Add**  to create a parameter in the table.

Note:

You can set the default **Usage** value with the **PLE\_Parameter\_Create\_With\_Default\_Usage preference**.

- Click **Propagate** to update one or more mapped parameter propagation directions: **Primary to Secondary** or **Secondary to Primary**.

**Select propagation direction**

▼ INPUT PARAMETERS

Name	Units	Revision	Source	Value Usage	Initial	Goal	Min	Max	Resc
Aircraft_Mass	kg	A	Empty weight	Constant	11700	11700	10000	12000	
Aircraft_Mass	kg	A	032355/A;1-Study_...	Constant	11700	11700	10000	12000	
Hybridation_Level	Each	A	REQ-000049/A;1-H...	Continuous	0.3	0.5	0.2	1	201
Hybridation_Level	Each	A	032355/A;1-Study_...	Continuous	0.3	0.5	0.2	1	9
Payload_Mass	kg	A	Payload	Continuous	7000	7000	5000	8000	201
Payload_Mass	kg	A	032355/A;1-Study_...	Continuous	7000	7000	5000	7000	101

**Select parameters**

▼ INPUT PARAMETERS

Name	Units	Revision	Source	Value Usage	Initial	Goal	Min	Max	Resc
Aircraft_Mass	kg	A	Empty weight	Constant	11700	11700	10000	12000	
Aircraft_Mass	kg	A	032355/A;1-Study_...	Constant	11700	11700	10000	12000	
Hybridation_Level	Each	A	REQ-000049/A;1-H...	Continuous	0.3	0.5	0.2	1	201
Hybridation_Level	Each	A	032355/A;1-Study_...	Continuous	0.3	0.5	0.2	1	201
Payload_Mass	kg	A	Payload	Continuous	7000	7000	5000	8000	201
Payload_Mass	kg	A	032355/A;1-Study_...	Continuous	7000	7000	5000	8000	201

**Secondary parameter properties are updated**

- Click **Start Edit** to update fields. Changes automatically update related fields. For example, if you update a parameter that is now out of the Min and Max range, the **Result** field automatically updates to **Fail**.
- Select a parameter, and then click **Manage Values and Measurements** to manage **multidimensional complex values** for the parameter. Perform any of the following actions:
  - Click **Table Value from Excel** to select a Microsoft Excel file to import multidimensional complex values.
  - Click **New Table Value** to **create** and **manage** values and measurements for the parameter. This table is useful for managing parameters that vary over time or frequency.
  - Select a parameter, and then click **Measurements** to manage measurement values for the selected parameter.
- The default **Usage** direction value is **Output**.

- The **Result** field **Pass** and **Fail** status automatically roll up from child to parent test pack objects.
- You can copy and paste parameter rows, including the parameter values.

For other information about parameters, see the following topics.

- For general parameter information, see [Managing global parameters](#).
- For information about input and output parameter usage, see [Manage input and output parameters](#).
- For information about verification and test request parameters, see [Managing parameters associated with a test](#) and [multidimensional table value parameters](#).
- For information about updating parameters, see [Add parameters to a verification request](#).
- For more information about mapping parameters, see [Map parameters from system models and requirements to simulation model parameters](#)


## Generate parameter value line charts and tables

You can display a chart, table, or a combination of both for selected parameter values.

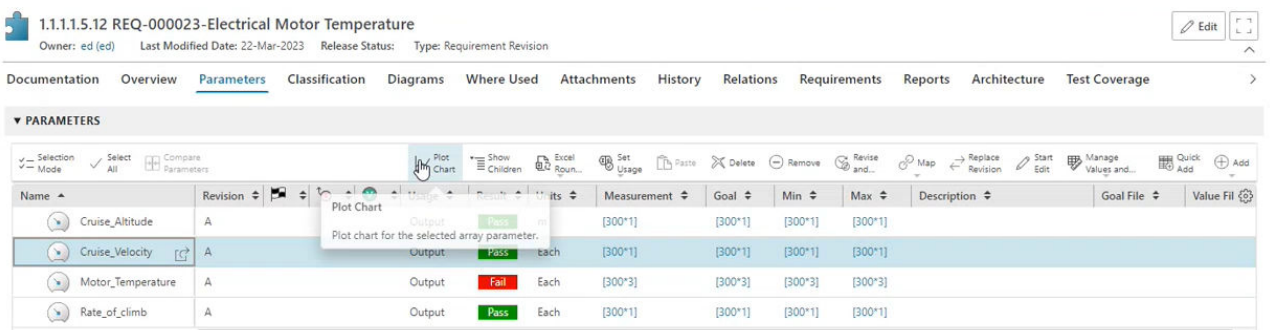
### Prerequisites

You must have existing [multidimensional complex parameter](#) data that includes **Goal** values.

### Procedure

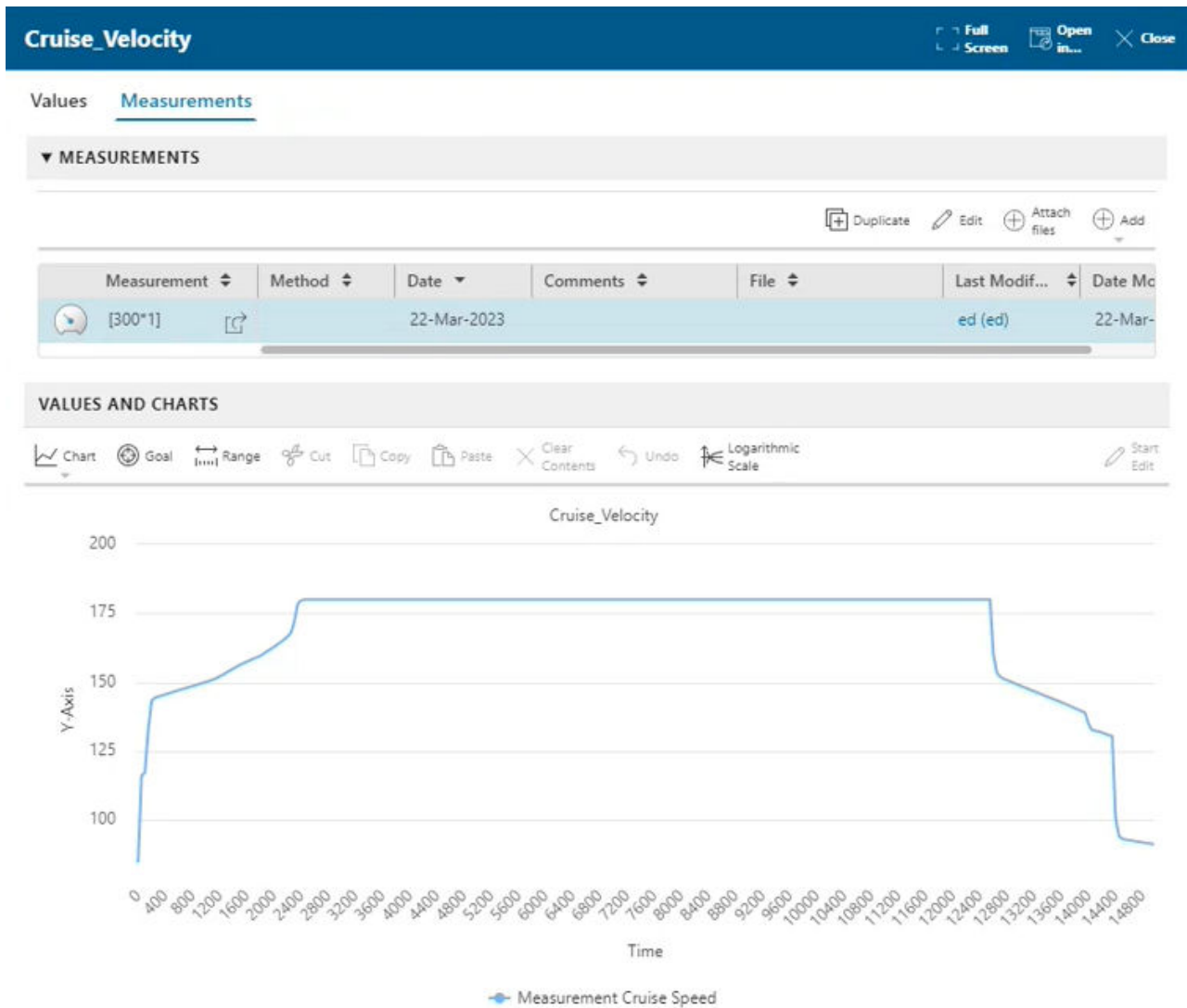
1. Locate and view the **PARAMETERS** section for a test package object.
2. Select a complex parameter, and then click **Plot Chart** .

In the following graphic, the user has selected a complex parameter from a requirement **PARAMETERS** table.



Name	Revision	Output	Units	Measurement	Goal	Min	Max	Description	Goal File	Value Fill
Cruise_Altitude	A	Output	each	[300*1]	[300*1]	[300*1]	[300*1]			
Cruise_Velocity	A	Output	each	[300*1]	[300*1]	[300*1]	[300*1]			
Motor_Temperature	A	Output	Each	[300*3]	[300*3]	[300*3]	[300*3]			
Rate_of_climb	A	Output	Each	[300*1]	[300*1]	[300*1]	[300*1]			

A dialog appears with the name of the selected parameter in the **Measurements** tab. The **VALUES AND CHARTS** section shows the line chart of the selected complex parameter.



- Perform any of the following on the line chart.
  - Hover over the line to see the underlying plotted values.
  - Click **Chart** and select either **All** or **Table** to display the chart and table, or the table only.
  - Click **Goal** to display the goal value line.
  - Click **Range** to show the range values.
  - Click **Logarithmic Scale** to change the scale.
  - Click on a label at the bottom of the chart to toggle the displayed data.



4. Perform any of the following on the table.
  - Click **Chart** and select either **All** or **Table** to display the chart and table, or the table only.
  - Update the table values and the chart updates automatically. The underlying **PARAMETERS** table updates automatically also, including **Status**.
  - Copy and paste data rows.

### Manage runs

As an alternative to selecting individual runs in the **results panel test package tree**, you can manage runs and related parameters for a selected study or test request.

Select a test object and then click **Runs**.

### Manage test participants

You can manage the users who take part in testing.

Select a **test object** and then click **Participants**.

## Run tests, monitor test coverage, and view test results

### Create and view test case coverage information against requirements

You can view the test case coverage status and other test information for selected requirements. You can also create a test case and add it directly to a requirement.

1. Open a requirement specification and then click the **Test Coverage** tab.
2. Select one or more requirements to update the test coverage information.

- **TEST CASE COVERAGE**

Provides a pie chart overview of the selected requirements that either have or do not have test cases associated with them.

Click **Requirements with Test Cases** or **Requirements without Test Cases** to filter the **TEST CASE COVERAGE AND STATUS** list.

- **TEST CASE STATUS**

Provides the **Released** or **Not Release** test case status for the selected requirements.

- **TEST COVERAGE AND STATUS**

Provides a sortable list of test cases with related information for the selected requirements.

Select a single requirement and click **Add Test Case** ⊕ to create a test case with a trace link directly to the selected requirement.

Tip:

To pick specific branches from which to expand or collapse all children, select the elements and then click **Expand/Collapse Below**.


## Manage simulation requests, test requests, and results in the Trends tab

You can manage the simulation and test request, the test results, the associated parameters, and the simulation models.

Open a test request and then click **Trends**.

### TEST ITERATIONS

Defines the simulation and test requests.

- Open  a test to display related test information in the **Overview**.
- Click **Add** ⊕ to add another simulation or test request.
- Select one or more elements to update the scope of all other panels.
- Double-click a field to edit it.

### RESULTS

Provides a summary of the test results selected in the **TEST ITERATIONS** panel.

- Click **No Results**, **Pass**, or **Fail** to filter the data in the other panels.
- Hover over a bar to see the underlying data counts.

## PARAMETERS

Provides a summary of parameters associated with the selected tests in the **TEST ITERATIONS** panel. Also provides parameter-related data and test results.

- Measurements that are above or below the **Min** and **Max** values and outside of the **Deviation** are highlighted in red.
- Double-click a field to edit it.
- Parameters with multidimensional table values display its values as table dimensions in the format **[rows\*columns]**. The following example graphic shows that the **Braking Distance** parameter has a value table with 17 rows and 3 columns, which displays as **[17\*3]** in each row **Measurement**, **Min**, **Max**, or **Goal** value cell.

▼ Parameters

Name	Deviation	Usage	Units	Measurement	Goal	Min	Max	Initial
Test_Param_1		Output	cm	[2*2]	[2*2]	[2*2]	[2*2]	
Test Param 2		Output	m		5	1	10	

For more information on table value parameters, see [Understanding multidimensional table value parameters](#).

## SIMULATION MODEL

Provides the simulation models associated with the selected tests in the **TEST ITERATIONS** panel.

- Open a simulation model to open in its related application.
- Double-click a field to edit it.

## TEST ARTICLES

Provides the test articles associated with the selected tests in the **TEST ITERATIONS** panel.

### Perform a simulation after updating test request

You can automatically perform a simulation after updating your test request and then saving.

### Prerequisites

- You must have an existing test to execute.

- The test request must already be associated with a simulation tool.
- The parameters associated with the test request are properly mapped with the simulation tool parameters.

### Procedure

1. Open a test request.
2. Update the test request.
3. Save the verification request.

The **Open in Simulation Tool** dialog appears.

4. Select a simulation tool and then run it.

### Results

Active Workspace executes the simulation with the updated data.

### Execute a test

You can execute a test, which is called a run.

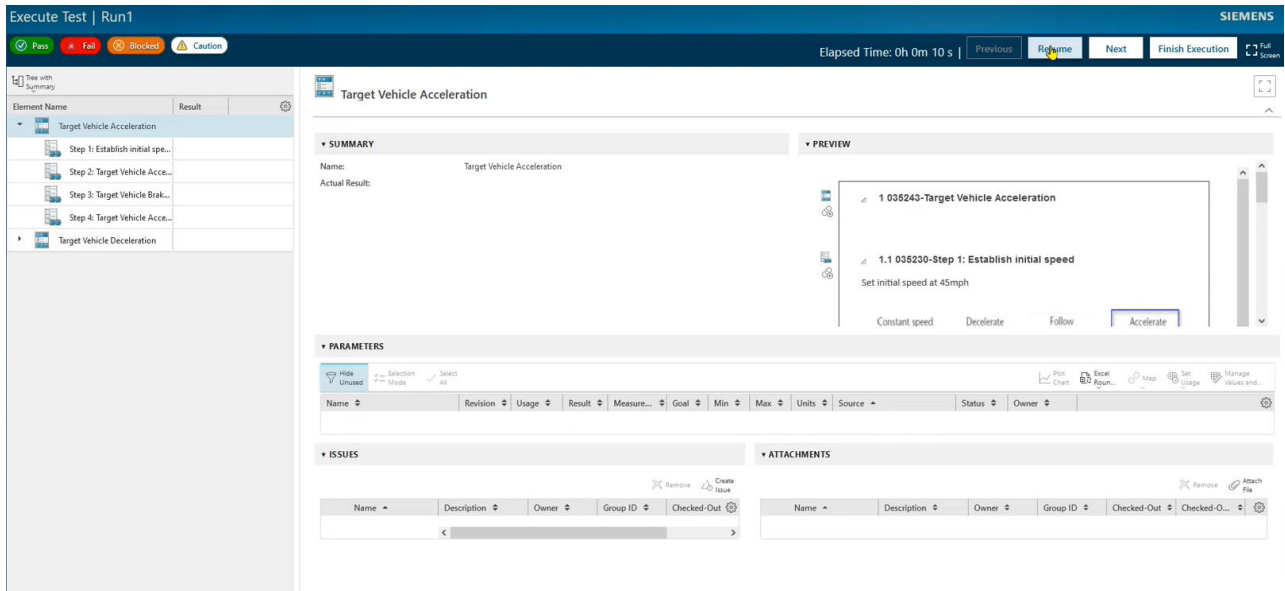
### Prerequisites

You must have an existing test to execute.

### Procedure

1. In the test request, select a test object from the tree.
2. Click **Add** ⊕ and then select **Run**.
3. Complete the **Add Child** dialog, and then click **Create**.
4. The run appears as a child of the selected test.

The following graphic shows a typical **Execute Test** screen.

**Note:**

Deployed software releases associated with each test appears at the top of the page.

**Tip:**

Click **Previous** or **Next** to move through the test and steps.


5. Perform any of the following with an test **Element Name** selected:

- Many sections and most tables provide similar actions. For more information, see [General test object panel, table, and chart actions](#).
- To add a test result icon, select the test, and then click a result button: **Pass**, **Fail**, **Blocked** or **Caution**. The corresponding icon appears in the **Result** column. The **Result** status rolls up from child to parent; therefore, if a child fails, the parent **Result** is also **Fail**.

**Note:**

If you add a **Fail** result, you can click the icon and create a change management issue.

- To update the test name, click in the **Name** field.
- To add notes to explain the results, click in the **Actual Result** field.
- To add software release related to a run, select the run **Element Name**, and then in the **Software** panel, click **Add > Add From Content** to select a software package.

- To record parameter results, double-click the **PARAMETERS** table, and then enter values. The **Result** column automatically populates as a **Pass** or **Fail**.
  - To create an issue, click **Create Issue** in the **ISSUES table**.
  - To view or add supporting documents or other files, perform the following in the **ATTACHMENTS** panel:
    - To view an attachment, click the attachment.
    - To add an attachment, click **Attach File** .
6. Click **Save**.
  7. Click **Finish Execution**.

## Results

Active Workspace completes the run and returns to the **Overview tab**.

### View test results

You can view the pass/fail test results information for selected requirements.

1. Open a requirement specification and then click the **Test Results** tab.

Because a requirement can have multiple tests associate with it, the table shows multiple instances of each requirement with the associated **Parent test**.

The *Result* column *Pass, Fail, Blocked, and Caution*

The *Element Name* column shows requirements, tests, and the runs. The **Result** status rolls up from child to parent; therefore, if a child fails, the parent **Result** is also **Fail**.

2. Perform any of the following:
  - Multi-select elements to change the scope of the test results.
  - Click pie pieces to filter by test results.
  - Click a parent test link to display the test details in the Overview tab.

## Managing simulation analyses with Multi-disciplinary Analysis and Optimization (MDAO)

### MDAO simulation in Active Workspace

Multidisciplinary Analysis and Optimization (MDAO) is a software-driven methodology that lets you optimize systems through simulation and across engineering disciplines.

When you are designing your system models with software such as System Modeling Workbench (SMW), you simulate your models to ensure they are valid.

You can open an embedded Active Workspace window within your system modeling software and create a test to define the simulation and analysis that you want to perform. You can perform MDAO through the following high-level process:

1. Launch through Teamcenter SimCenter to the HEEDS analysis tool.
2. Run the simulation.
3. Receive the results.
4. Decide on the changes to optimize your system model.

**Siemens Simcenter** A flexible, open, and scalable portfolio of the best predictive simulation and test applications. Simcenter optimizes the performance of complex products throughout the lifecycle. Starting from the early stages. Simcenter allows engineers to generate a set of ultrarealistic, multi-physics models and data that can predict real product behavior.

**HEEDS** A design space exploration and optimization software package that interfaces with all commercial computer-aided design (CAD) and computer-aided engineering (CAE) tools to drive product innovation. HEEDS automates analysis workflows (process automation), maximizes the available computational hardware and software resources (distributed execution), and explores the design space for solutions (efficient search), while assessing the new concepts to ensure performance requirements are met (insight and discovery).

The topics in this section are specific to managing parameters with simulations. For general parameter topics, see [Parameter Management Overview](#).

### Manage simulation requests

You can review and update the test object (verification request, test request, studies, runs, and so on) shown in the **Overview** tab, panels which provide a snapshot of the test package data.

Open a test request and then click the **Overview** tab:

The following sections provide an overview of the actions you can perform and descriptions of the panels available on the **Overview** tab.

## General panel and table actions


Many sections and most tables provide similar actions. For more information, see [General test object panel, table, and chart actions](#).

## Results panel test package tree

The results panel displays the test package object tree, where you can select the scope of your data focus. For more information, see [Results panel test package tree](#).

## SUMMARY


Provides an overview of the verification request that you selected in the results panel test package tree.

Click **Edit Localization**  to open the **Edit Localization** dialog to update the **Name** and **Description** field text into alternate languages.

## TEST RESULTS


Provides a summary of the **Pass** or **Fail** test results. Test results propagate up from child to parent. If all children have a **Pass** result, then the parent results in a **Pass** as well. If any child has a **Fail** result, then the parent results in a **Fail**.

The underlying data on which the test results are based are provided in the **REQUIREMENTS** panel table. Click on the graphic to filter the **REQUIREMENTS** panel table based on **No Result**, **Requirements Fail**, or **Requirements Pass** result.

Click **Add**  to add a study, test request, run, or simulation request. For more information, see [Add elements to a test](#).

## PROGRAM EVENTS

Provides the event milestones to which the test for which the test is due.

Click **Add**  to add a milestone.

## REQUIREMENTS

Provides a manageable list of requirements related to the simulation request, the **Result** status, and a **chart** to filter the data.

For more information, see [General test object panel, table, and chart actions](#).


## TEST CASES

Provides a manageable list of test cases associated with the verification request.

## SIMULATION MODELS

Provides a manageable list of simulation models to run.

Note:

- If you have a large number of simulation models, when you add a simulation model using the **Add** dialog, click **Search** and filter  by classification criteria to select a model.
- When you add a simulation model, each parameter within the simulation model that matches the name of a parameter in the parameter table is automatically mapped. For more information about mapping parameters, see [Map parameters from system models and requirements to simulation model parameters](#).

## INPUT/OUTPUT PARAMETERS

Provides a manageable list of simulation request parameters with **Usage** values of **Input** or **Output**.

For more information, see the following:

- Parameter tables, see [Parameter table actions](#).
- Automatic mapping of parameters with integrated products, see [Automap parameters](#).
- Manual parameter mapping, see [Manually map parameters from system models, requirements, and integrated simulation products to simulation model parameters](#).

The parameter table for simulations provides the following fields:

- **As Used**

Acts as a temporary override to the **Goal** value.

- **Formula**

Allows entering of text string formulas, such as **x+22** or **speed(2u\*t)**.

- **Minimum Measured**

Simulation measured value.

- **Maximum Measured**

Simulation measured value.

- **Value Usage**

Defines how the simulation tool analyzes the value throughout the simulations:

- **Continuous**

Allows the value to change.

- **Dependent**

Use the value as a reference which the text formula in the **Formula** column defines.

- **Constant**

Use the same value.

- **Discrete Values**

Discrete list of values. Entered in the **Discrete Values** column.

- **Ordered**

Dictates if the order of the values in the **Discrete Values** column must be followed.

Values: **True** or **False**

Note:

For more information about input and output parameters, see [Manage input and output parameters](#).

- **Objective**

For output parameters, allows you to set the value: **Minimize**, **Maximize**, **Minimize difference**, **Maximize difference**.

- **Target**

For output parameters and the **Objective** value is set to **Minimize difference** or **Maximize difference**, allows you to define the target value.

## OTHERS

Provides a manageable list of any objects added to the test that are not included in other tables.

## REPORTS

Provides a manageable list of reports.

### Runs tab

For more information, see [Manage runs](#).

### Participants tab

For more information, see [Manage test participants](#).

## Manage multidisciplinary model and analysis creation and execution from the verification request

You can run test analyses through Teamcenter while creating a verification request for the management of the request, and also launch a related HEEDS analysis. For more information about HEEDS, refer to the HEEDS documentation.

In the procedure, the term *source* refers to the application that is synchronizing or publishing to from Active Workspace, such as System Modeling Workbench.

### Procedure

1. Select a study in the **Scope** panel.
2. In the **TABLES** panel, select **Simulation Models**.
3. In the **SIMULATION MODELS** panel, select a simulation model.

Note:

You can also execute an analysis without selecting a simulation model.

4. Click **Open > Open in Simulation Tool**.

Active Workspace loads the configuration for the simulation tool and displays the results in the **Open in Simulation Tool** dialog box.

**Note:**

All options require a HEEDS license and a Teamcenter for Simulation license.

## 5. Select an option to run:

- **HEEDS – Create Model (Verification Scope)**

Creates a HEEDS project file, with input parameters transformed as HEEDS variables, and output parameters transformed as HEEDS responses. The relevant parameters properties are conveyed to HEEDS. Once you finish the HEEDS project authoring, click **Save** or **Final save**. **Save** allows you to save the model, which you can modify later. **Final save** also creates the underlying analysis object, allowing you to execute the analysis.

- **HEEDS – Execute Analysis (Verification Scope)**

Opens the HEEDS project with the user interface on the your local machine. You then manually execute the analysis and save the file once the analysis completes. This option works only when there is an MDAO analysis object in the verification request.


- **HEEDS – Execute Analysis Batch (VR Scope)**

Executes the HEES project silently and then display results. This option works only when there is an MDAO analysis object in the verification request.

- **HEEDS – Modify (VR Scope)**

Opens the HEEDS model (if the model exists) and allows you to modify the model. This option works only when there is an MDAO analysis object in the verification request.

## Results

Use the **TOOL PROGRESS MONITOR**  tile on your home page to follow the simulation.

## Include MDAO model and MDAO analysis object in a verification request

You can include the MDAO analysis and MDAO model in the verification request object in the following ways:

- Create an MDAO model, and **perform a Final save**.
- Add an MDAO model and an MDAO model analysis. If you choose this option, note that MDAO model must relate to MDAO analysis.

If you manually add the MDAO model and analysis, you must **map the parameters** coming from the system or requirement parameters.

## Automap parameters


When you add a simulation model to a simulation or a verification request, select the **Automap Parameters** check box. Each simulation model parameter that matches (case-sensitive, exact name, data type, and **Input** or **Output Usage**) an existing simulation request parameter maps automatically. Updated parameters are automatically refreshed between the integrated products and Active Workspace.

## Manually map parameters from system models, requirements, and integrated simulation products to simulation model parameters


You can manually associate parameters within a requirement to their counterpart parameters in a simulation model to ensure that the correct requirement parameters run when the simulation is run. Mapped parameters are automatically refreshed between the integrated products and Active Workspace. The following graphic depicts the parameter table that shows the source of the parameters as both from requirements called **Braking System** and from simulation analysis results **TCSIM\_BrakeAnalysis1**. In this case, you want to map the requirement parameters to the simulation results parameters.

The following graphic provides the **PARAMETERS** table that shows the source of the parameters as both from requirements called **Braking System** and simulation analysis results **TCSIM\_BrakeAnalysis1**. In this case, you want to map the requirement parameters to the counterpart simulation results parameters.

### Procedure

1. In the **results panel test package tree**, select a **test package object**.
2. In the **PARAMETERS** panel, select a parameter to map.
3. Click **Map** .
4. In the **Map** dialog, select the parameter you want to map to. For our example, we select **Sim\_BD\_D-30**.

**Tip:**

If you want to perform multiple mappings, then **Pin**  the **Map** dialog box to keep the dialog open.

5. Click **Map**.

## Results


Active Workspace displays the mapped parameter as a child of the source parameter.

Note:

- In Teamcenter 14.2, if the parameter name contains a space, or if it starts with a digit, you must manually map the parameters after selecting **Open in Simulation Tool > Create Model (Verification request scope)**.
- For display consistency, regardless of mapping order (simulation to non-simulation parameter or non-simulation parameter to simulation parameter) the simulation parameter value always appears as a child of the source (non-simulation) parameter value.

## Perform automated simulations and record test results

You can launch a simulation directly from a test package in **Overview**, and then add the test results.

1. Open the test package in **Overview**.
2. Ensure that **SIMULATION ANALYSES** and **PARAMETERS** tables are displayed. If not, toggle them in the **TABLES** section list.
3. Select a test run.
4. In the **SIMULATION ANALYSES** pane, select a simulation.
5. Select  **Open > Open in the Simulation Tool**.

The **Open in Simulation Tool** dialog displays with a list of available simulation tools.

6. Select  **Open > Open in the Simulation Tool**.

The **Open in Simulation Tool** dialog displays with a list of available simulation tools.

7. Select a simulation tool.

The **Launch Inputs** tab opens. Use this tab to enter the launch properties, input objects, and to set the processing options.

8. Select the **Auto Process** option and the **Upload File/Update URL** drop-down. These options automatically process the results, and then upload the these results and their related files.
9. Click **Open** to start the simulation and to run the test.

The **Simulation Tool Progress Monitor** page opens and displays the progress status.

10. Once the testing completes, return to the test **Overview** tab to see the simulation test results data.

## Display MDAO optimization results

Once an MDAO analysis has completed, the optimized results are automatically sent back to the verification request object (verification request, simulation request, and so on) that was used to execute the analysis. You can view the simulation results for a selected study on a verification request.

Tip:

Click **Preview** to view to display the analysis results graphically.

## Restrictions and limitations

Consider the following:

- After a final save, you can change the parameters values, but you cannot add parameters to the verification request or to HEEDS.
- To support the execution of commands, only one MDAO analysis and only one MDAO model can be present in a simulation request. The MDAO analysis must relate to the MDAO model.
- The following considerations are specific to Teamcenter 14.2 and later:
  - If the parameter name contains a space, or if it starts with a digit, HEEDS changes the name of the parameter. You must manually map the parameters after selecting **Open in Simulation Tool > Create Model (Verification request scope)**.
  - Discrete variables are supported only for parameters that are doubles.
  - Only an MDAO analysis that produces one best result is supported. We recommend the usage of SHERPA: Weighted sum of all objectives.
  - Only HEEDS 2210 is supported.

## Prerequisites

An MDAO analysis has completed and is available.

## Procedure

1. Open a verification object for which an MDAO analysis has completed.
2. Click the **Overview** tab.

The report section tab lists the additional file that HEEDS produced during its execution. For more information about creating reports or adding files, see the HEEDS documentation.

## Results

The simulation results display in the **OUTPUT PARAMETERS** panel in the **Analysis Value** column. The results are color-coded green for pass and red for fail.

## Reporting test results

### Create a test report

You can generate reports of a test package or its components such as a simulation request, study, or run.

Note:

By default, Active Workspace provides several standard reports. For information about creating your own report templates, the system administrator should refer to the Teamcenter Reporting and Analytics documentation.

1. Open a test package in the **Overview** tab or the **Content** tab.

Active Workspace displays the **Generate Report** dialog box.

2. Select the Select a test package or component, and then select **New** ✨ > **Generate Report**.

Active Workspace displays the **Generate Report** dialog.

3. Select a report, complete the **FORMAT** dialog (if available), and then click **Generate**.

Active Workspace generates the report and downloads a local copy to your machine.

## View completion status and test results

### Overview

You can use the **Overview** to see a dashboard view of the ongoing results. You can also manage associated objects such as events, requests, and parameters. You can view associated requirements and BOM elements as well.

### Test results report

You can show a status report for one or more selected items in a BOM. This report includes items such as: pass or fail results, milestones, and the associated tests.

1. Open a BOM structure.
2. Select one more BOM elements.
3. Click **Details > Test Results**.

Note:

You can change your BOM selections and the test results update automatically.

Note:

The types of items for which the **Test Results** is shown is configured by your administrator with the **PLE\_Plan\_Table\_Allowed\_Child\_Types** preference.

# 6. Managing global parameters

## About parameter management

The global parameter management solution allows you to define and manage different types of parameters, including variables, characteristics, measurements, calibration, configuration, and requirements. You can maintain parameters in a single source dictionary that is shared across domains, applications, and life cycles.

Parameters manipulate a model using its defined variability (this may be a CAD model, functional model, behavioral model, and so on). You can modify the parameters of a model to achieve the desired form, fit, function, and behavior, which is typically expressed in terms of an attribute and one or more targets.

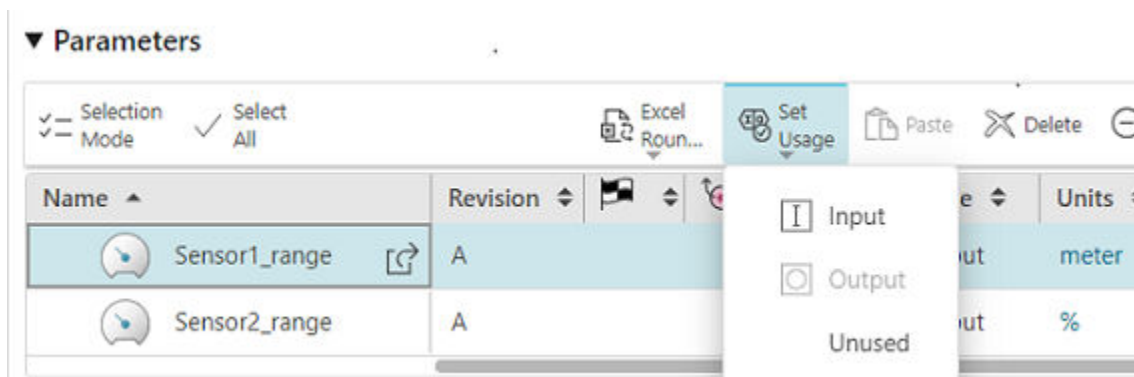
Parameters are standalone objects, and a single parameter may be associated with multiple contexts. Parameters may be used in requirements, system models, CAD design, and simulation analysis, for example, Teamcenter Simulation Process Management.

Working with common definitions allows you to avoid parameter duplication across different engineering domains. You can maximize reuse and consistency of parameter values across development stages and multiple programs. You can also track the history of parameters, assess change impact, capture the rationale of value changes, specify value variations, and support parameter definition standards (ASAM).

Note:

By default, parameter definitions are not required to create parameters. However, if the **PLE\_Parameter\_Create\_With\_Definition\_Ux** preference is set to **true** on the **Parameters** panel, then a parameter definition must be specified when creating a parameter. See **Make parameter definitions required** for more information.

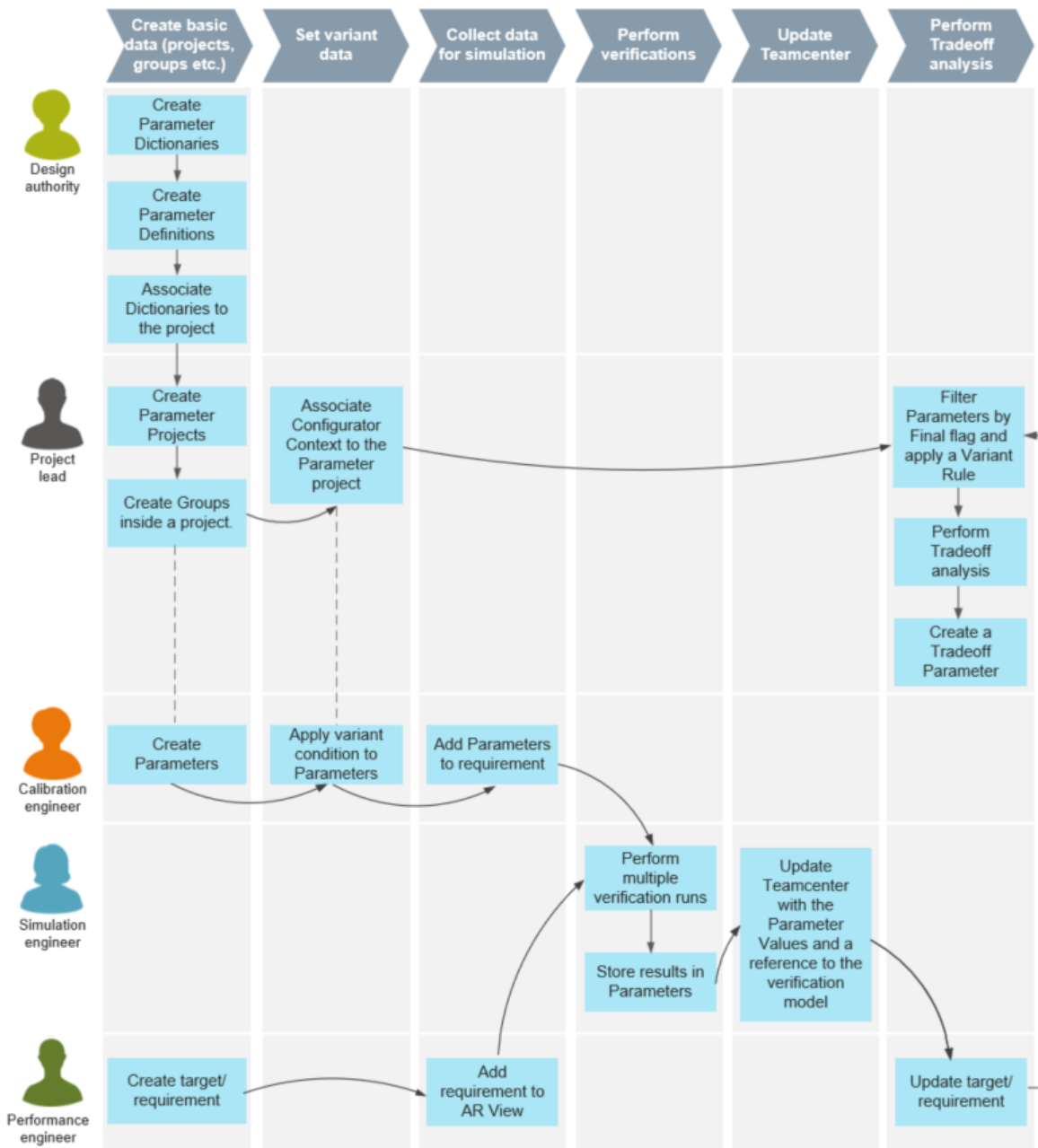
Each parameter has an *intent* in a particular context, which may be input (Ⓜ), output (Ⓞ), or unused. The intent is important when running simulations. You can specify parameter intent in a context (for example, an analysis revision or CAD design revision with the **Set Usage** command).



A parameter may be referenced by multiple item revisions as an input, but only one can use it as output. Revising an item revision carries forward its parameters with their intents to the new revision except for the output intent.

The solution allows you to:

- Create a **dictionary of parameter definitions** and the operations related to them.
- Capture **parameters values** and related actions from the parameter definitions.
- Create a **parameter project** to manage parameter values, rather than associating them to a design.
- Use **parameter variants** to configure groups of parameter values or parameter values in a parameter project.



## Creating and populating parameter dictionaries

### Manage parameter dictionaries

A parameter dictionary provides a common folder or repository for storing and managing parameter definitions. A parameter dictionary:

- Contains parameter definitions of a particular type, for example, performance or comfort. However, dictionaries are not mandatory.

- Supports standard editing commands on parameter definitions within a dictionary, for example, cut, copy, and paste.
- Allows different status to be applied to the parameter dictionary and the parameter definitions within it, for example, active, approved, or obsolete.
- Allows separate access rights to be defined on the dictionary and the parameter definitions within it.

A parameter definition:

- Is created by the design authority to set an organization goal and allowed range of a key parameter, used by engineers across different disciplines and products.
- Is not required by default to create a parameter. However, if the **PLE\_Parameter\_Create\_With\_Definition\_Ux** preference is set to **true** on the **Parameters** panel, then a parameter definition must be specified when creating a parameter. See **Make parameter definitions required** for more information.
- May participate in more than one dictionary.
- Is unique depending on fields and properties. Uniqueness is not enforced by name.
- Is assigned to a category. The system is provided with predefined categories but further categories may be defined at your site.
- Includes a data type, for example, boolean or integer. The data type cannot be changed or customized. However, the data type allows a downstream user to change range, for example, when performing *what if* scenarios that may require temporary modifications of the original parameter definition.
- Can be revised. You can also create a new parameter definition, rather than revising an existing parameter definition.

### Create a parameter dictionary

You can create a parameter dictionary to contain all the parameter definitions owned by your team. Parameter dictionaries may be nested, that is, you can create a parameter dictionary within another parameter dictionary.

1. Navigate to the folder where you want to create the dictionary, and go to **More commands ... > Add**.

The **Add** panel appears.

2. In the **Type** field, filter and select **Parameter Dictionary**.

The **Parameter Dictionary** fields appear.

3. Enter a name for the parameter dictionary, and then click **Add**.

## Creating parameter definitions

### About parameter definitions

A *parameter definition* sets the organizational goal and allowed range for a parameter used by different engineers across different disciplines and products. A parameter definition includes the unit of measure, type (for example, integer or string), complexity, default (starting) value, target value (goal), and the acceptable range expressed as minimum and maximum. For example, a basic parameter definition may be *weight* and then you can have specific definitions of how the weight parameter is applied in different contexts.

Parameter definitions can be attached to several parameter projects that share the unit of measure, goal, range, and potentially other properties. If you require a different goal, you must create a new parameter definition, if the range cannot be extended without compromising the organizational goals.

The goal is a target that may or may not be attainable. The default is the nominal starting point, an achievable value based on history. This is the value inherited when you create a parameter value.

Tip:

Parameter definitions are not required by default to create parameters. If you want to require a parameter definition, you can **set a preference** to establish the requirement.

### Create a parameter definition within a dictionary

You can create a parameter definition within a parameter dictionary.

Tip:

Parameter definitions are not required by default. To make them mandatory, you must **set a preference** specifying this.

1. Open the parameter dictionary in the work area.

The **Overview** panel appears, showing the table with parameter definitions.

2. Click **Add** ⊕
3. In the **Type** field of the **Add** panel, filter and select **Parameter Definition**.
4. Enter a **Name** for the definition.

5. (Optional) Select an **Application**.

This is the domain where the parameter definition is used. This selection determines the available data types.

Note:


If your administrator has enabled the **Application** field when creating a parameter definition, the **Quantity** option is currently not supported and should not be selected.

6. Select a **Data Type**.
7. (Optional) Select a **Unit of Measure**.

You can filter the list of units of measure based on the quantity name (for example *length*) or the unit name (for example *centimeter*). This unit of measure becomes the As Authored unit. It can be converted to a different unit by changing the unit display set on the user profile. See Manage your User Profile for information on changing the unit display set.


8. (Optional) Enter a **List of Values** from which the user can select a value.

The list of values depends on the selected data type.

9. (Optional) Select the **Restrict List of Values** check box to restrict entries to the defined list. If the check box is not selected, the user can enter free-form values.
10. (Optional) Select an **Owning Project**.
11. (Optional) Click **Add Project**  to add a new project.
12. Click **Add**.

The system creates the parameter definition.

Note:

You must release the parameter definition with the appropriate workflow before you can use it to create the parameter value. Release a parameter by selecting one or multiple parameter definitions and clicking **Release** .

## Revise a parameter definition in a parameter dictionary

To revise a parameter definition in a parameter dictionary, you must first update the AttOParamDictionarySummary stylesheet to enable the application to show revised definitions.

## Restrictions and limitations

- The system administrator should be responsible for updating the AttOParamDictionarySummary.

## Procedure

1. Click the **XRT Editor** icon on the Home page.

The **XRTEditor** page appears.

2. Select **Parameter Dictionary** in the **Object Type** field.

3. Click **Load**.

The **AttOParamDictionarySummary** stylesheet appears.

4. Click **Start Edit** .

A dialog box appears with a message indicating the AttOParamDictionarySummary value will be overridden at the user scope.

5. Click **Cancel**.

The dialog box closes.

6. Navigate to the user profile and set the **Group** field to **dba** and the **Role** field to **DBA**.

7. Select **Parameter Dictionary** in the **Object Type** field and click **Load** again.

8. Click **Start Edit** .

9. Navigate to the **tc\_xrt\_ParamDefinitions** section.

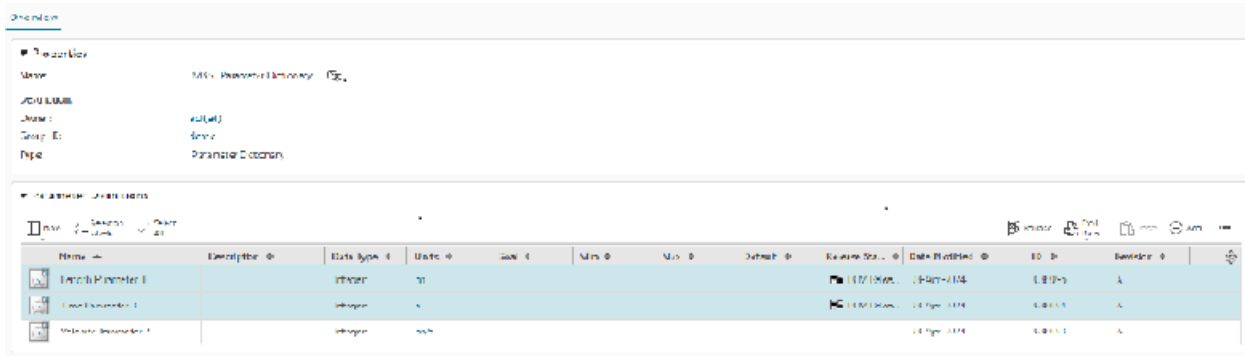
10. In the **objectSet source** row, add the property **showConfiguredRev="true."**

11. Click **Save Edits**.


## Release parameter definitions

You can release one or more parameter definitions with a single click.

1. Open the parameter dictionary, and then select one or more parameter definitions.



## 2. Click **Release** .

The parameter definitions are released and a flag icon  appears in the **Release Status** column.

## Creating parameters

### About parameters

A parameter is created by an engineer working in a parameter project or on a design to store results. It can be associated with and derived from a parameter definition. Parameters have many associations with or consumption to a variety of domains, systems, products, and projects.

A parameter has a single value that lies between the minimum and maximum values specified in the parameter definition. Parameter values can be configured with the **variant options** available on the project or the product.

A parameter may be created in your home folder or in the appropriate working context. The parameter may subsequently be assigned to an item or an element in a product structure, which may represent a requirement, simulation data, or CAD data.

The final parameter is released with a workflow and a release status is it at the last stage in the design evolution. You can use the release (effective) date to view a parameter at any time.

You can subsequently revise the parameter, for example if the goal changes, to create a new parameter revision. Each parameter revision has its own release status, and you can view details of each parameter revision in the **History** tab of the parameter.

### Create a parameter in a project, set, or other location

You can create a parameter within a parameter project, parameter set, or another location, such as within a product item. This procedure creates the parameter at the root level of the project or group.

You can add parameters by using the **Add Parameter** panel or the **Quick Add** feature.

You can also get notified on any changes to a parameter by using the **Follow** functionality. You can choose the frequency and priority of notifications. All notifications appear under the **Alerts** icon.

**Note:**

Parameter definitions are not required by default. If you prefer to require a parameter definition, you can **set a preference** to establish the requirement. The parameter inherits all available properties on the parameter definition.

1. Open the parameter project, or set and then open **Parameters**.

The **Parameters** table appears.

2. Use the **Add Parameter** panel or the **Quick Add** feature and follow the corresponding steps for your choice.

To use	Do this
The <b>Add Parameter</b> panel	Click <b>Add</b> ⊕. The <b>Add Parameter</b> panel appears.
The <b>Quick Add</b> feature	Click <b>Quick Add</b> . A row appears in the <b>Parameters</b> table.

3. Enter a name for the parameter.
4. Select an **Application**.

This is the domain where the parameter definition is used. This selection determines the available data types.

**Note:**

If your administrator has enabled the **Application** field when creating a parameter definition, the **Quantity** option is currently not supported and should not be selected.

5. Select a **Parameters Definition** if required.

The parameter inherits the values for the remaining fields from the parameter definition.

6. Select a **Data Type**.
7. (Optional) Select a **Unit of Measure**.

You can filter the list of units of measure based on the quantity name (for example *length*) or the unit name (for example *centimeter*). This unit of measure becomes the As Authored unit. It can be

converted to a different unit by changing the unit display set on the User Profile. See [Manage your User Profile](#) for information on changing the unit display set.

8. (Optional) Set values for the remaining parameter fields.
9. Perform one of the following steps:
  - If using the **Add Parameters** panel, click **Add** ⊕.
  - If using the **Quick Add** feature, click **Save** 📄.

The new parameter appears in the parameter table.

10. (Optional) To be notified of parameter notifications, do the following:
  - Click **More commands** ⋮ > **Share** > **Follow**.

The **Follow** panel appears.

- Click **Edit** to select changes for which you will receive a notification.
- Select a **Frequency** and a **Priority**.
- Click **Follow**.

Note:

You cannot create a baseline for a parameter value.

Note:

You can also create:

- A new parameter by copying an existing parameter and then pasting it with a new name within the same parameter project. If you do this, any associated variant condition is also save with the parameter. This capability may be useful when performing iterative processing of parameters in runs, to avoid creating a new one each time.
- A reference copy (using **Reference**) of a parameter.

## Create a parameter directly in the product

You can create a parameter directly in the product by using the **Add Parameter** panel or the **Quick Add** feature.

**Note:**

Parameter definitions are not required by default. If you prefer to require a parameter definition, you can **set a preference** to establish the requirement. The parameter inherits all of the available properties on the parameter definition.

1. Open the product structure where you want to add a parameter, and click **Details > Parameters**.

The **Parameters** table appears.

2. Use the **Add Parameter** panel or the **Quick Add** feature and follow the corresponding steps for your choice.

To use	Do this
The <b>Add Parameter</b> panel	Click <b>Add</b> ⊕. The <b>Add Parameter</b> panel appears.
The <b>Quick Add</b> feature	Click <b>Quick Add</b> . A row appears in the <b>Parameters</b> table.

3. Enter a name for the parameter.
4. Select a **Parameter Definition** if required.

The parameter inherits the values for the remaining fields from the parameter definition.

5. Select a **Data Type**.
6. (Optional) Select a **Unit of Measure**.


You can filter the list of units of measure based on the quantity name (for example *length*) or the unit name (for example *centimeter*). This unit of measure becomes the As Authored unit. It can be converted to a different unit by changing the unit display set on the user profile. See *Manage your User Profile* for information on changing the unit display set.

7. (Optional) Set values for the remaining parameter fields.
8. Perform one of the following steps:
  - If using the **Add Parameters** panel, click **Add** ⊕.
  - If using the **Quick Add** feature, click **Save** 📄.



The new parameter appears in the parameter table.


## Override parameter value in context

You can override a parameter value in the context of the product structure root node, the immediate parent, or any selected level in the hierarchy. If you do not select a context, the override is applied to the BOM element associated with the parameter revision.

1. In the left-hand panel, expand the tree, and select the level at which you want to make an in-context edit of parameter values.
2. Click **Edit Structure** , and then choose **In-Context**.

The system makes the selected level in the tree the context for your parameter edits.

3. Click the **Parameters** tab, select a parameter below the context level, and enter edit mode by clicking .
4. Click the cell containing the value you want to edit in context, for example, **Measured**, **Goal**, **Min** or **Max**, and then enter the new value.
5. Click  again to save your changes and exit edit mode.

Active Workspace adds a  symbol on the line containing to the in-context edit.

If you hover the cursor over this symbol, information about the context of the edit displays.

If the parameter is reused in another project or product, the in-context value is not seen, rather the original value is used instead.

## Create a weight parameter

You can create weight parameters for use from the conceptual design phase through the design and production phases of the product. Typically, a weight parameter is created by a weight analyst who assigns budgeted weights to different parts and assemblies in the product. The weight analyst may not have write access to the BOM elements which consume those parameters. Consequently, the access check on each such BOM element is bypassed just for the weight parameter and access rules are enforced for other type of parameters. The engineer or designer who owns the BOM element may not have access the weight parameter, but can add new values (measurements) by editing the property cells in the table or through the CAD system integration.

Each weight parameter can have multiple maturity values, for example, measured, calculated, estimated, and budgeted. When you add a weight parameter to a BOM element, the parameter and its values are associated with the BOM item revision and cannot be overridden in the same context. The system records the **modifiedBy** and **Last Modified Date** for each measurement.

**Note:**

Your administrator must add **Weight** to the **PLE\_MeasurableAttrSubTypes** preference to enable the creation of weight parameters.

To create a weight parameter:

1. Create a **parameter definition** for the weight parameter.
2. In the **Add** panel, select **Weight** from the **Application** list.

The **Data Type** field is set to **Double** for this value and you cannot change it.

3. (Optional) Set values for the following fields:

- **Unit of Measure**

You can filter the list of units of measure based on the quantity name (for example *length*) or the unit name (for example *centimeter*). This unit of measure becomes the As Authored unit. It can be converted to a different unit by changing the unit display set on the user profile. See Manage your User Profile for information on changing the unit display set.

- **List of Values**




This field allows you to define a list of values from which the user can select. The list of values depends on the selected data type.

- **Restrict List of Values**

This check box allows you to restrict entries to the defined list of values. If the check box is not selected, the user can enter free-form values.

4. Click **Add**.

The application creates the weight parameter.

5. Open the product BOM, and select the line representing the part or assembly to which you want to add the weight parameter.
6. Click **Edit** , click in one of the mass (weight) property cells, and then enter the parameter value. Typically, budgeted (goal) or estimated values are entered at this point in the life cycle; from design and actual values will be entered later.
7. Repeat the previous step for any other weight parameters, and then click **Save**  > **Save Edits**.
8. Optionally, when you have entered all the necessary parameter values, choose **New**  > **Rollup**.

Active Workspace displays the **Rollup** dialog box.

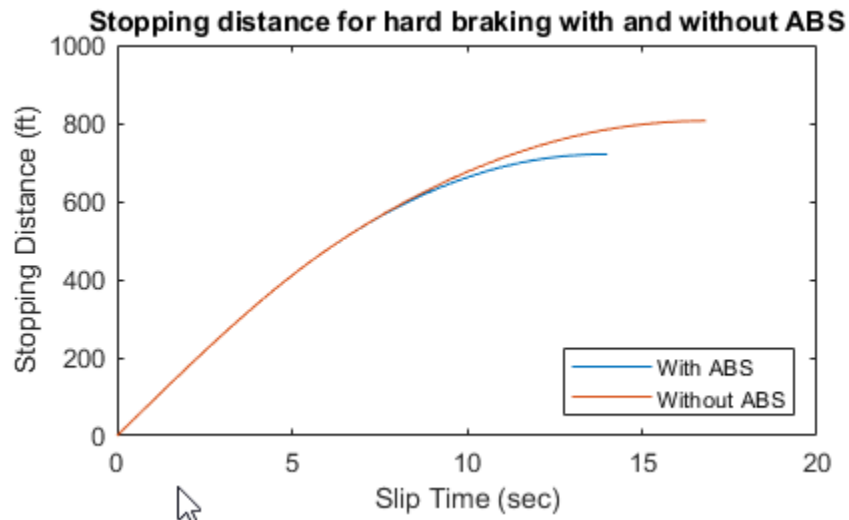
9. Enter a name for the rollup, select the template to use, and then click **Rollup**.

Active Workspace displays a rollup report based on the selected template. For each parameter, the most mature value is shown.

## Managing table value parameters


### Understanding multidimensional table value parameters

In engineering disciplines like MCAD design or CAE analysis, parameter values are often curves and maps with multiple values. Managing curves and maps is required for a multidomain parameter management solution. Therefore, parameter values can be a 2D or 3D curve with discrete values captured in an array table. For example, the parameter capturing the stopping distance for hard braking with ABS and without ABS is shown in the following graphic.

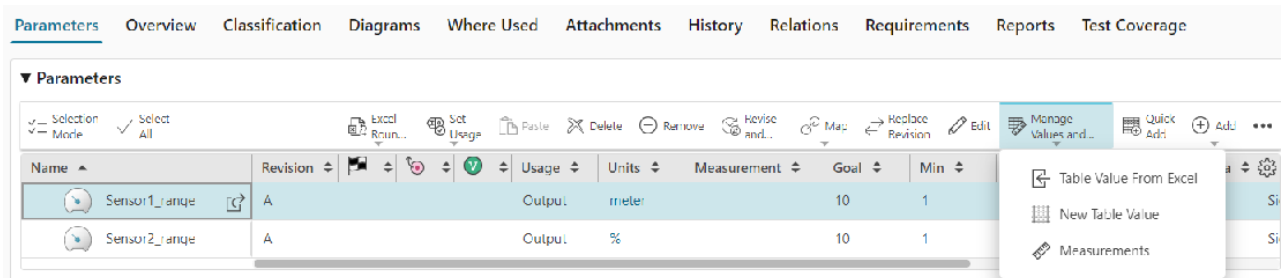


### Create a table value parameter

You can manually create an array value table for a selected parameter. For information about importing parameter table values, see the section starting with **Importing parameters**.

1. Select a **parameter table row** for which you want to add the multidimensional table values.
2. Click **Manage Values and Measurements**  > **New Table Value**.

The following example graphic shows a user preparing to add a value table to a selected parameter.



The Values panel opens showing the **Table Dimensions** dialog box.

3. Enter the number of rows and columns for the new value table, and then click **OK**.

An empty value table displays.

4. Double click a cell to enter a value, including the column and row title cells.

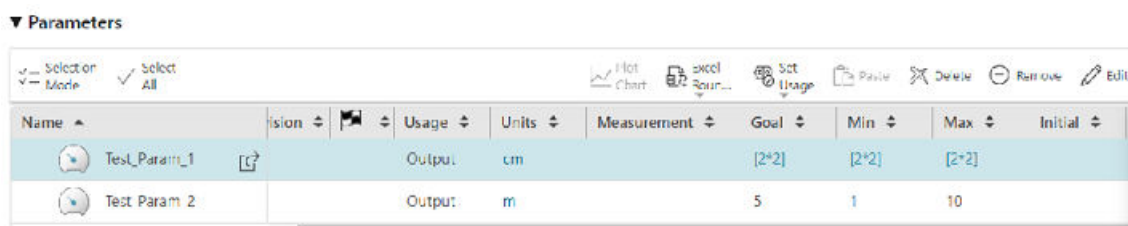
For information about editing the **Values** or **Measurements** panel tables, see [Edit multidimensional value and measurement tables](#).

## Edit parameter value and measurement tables

You can edit the array parameter values and measurements in a tabular format. This table is helpful for managing parameters that vary over frequency or time.

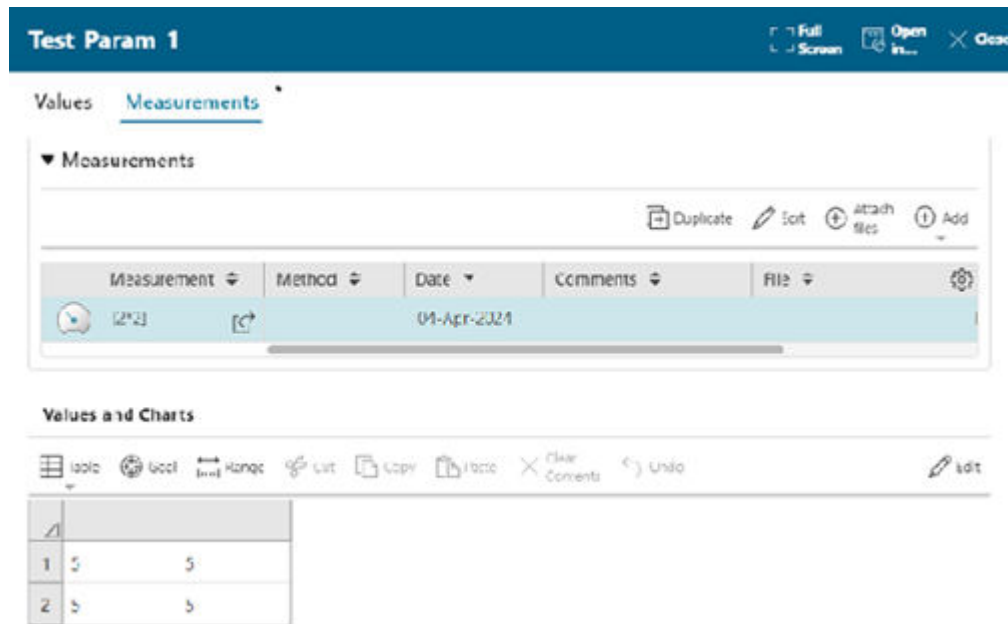
### Open the value and measurement table

1. Select the **parameter table row** for which you want to edit the values or measurements. Parameters with multidimensional table values display its values as table dimensions in the format **[rows\*columns]**. The following example graphic shows that the selected parameter has a value table with 2 rows and 2 columns, which displays as **[2\*2]** in each **Goal**, **Min**, or **Max** value cell.




2. Click a dimension to open the values and measurements dialog box.

The name of the dialog box is the name of the parameter. Following the previous example, in the selected row, the user clicks the **[2\*2]** value and the dialog box opens.






## Edit values

1. Click the **Values** tab.
2. Perform any of the following with the value table:
  - To modify table values, click **Edit** .
  - To modify the column header name or unit type, right-click the header, then click **Update Column Header and Units**.
  - By default, the table displays the Goal values. To see the Min and Max values as well, click **Range**.
  - To modify the row or column structure, select a row or column and then select **Insert Row Above** or **Insert Column Left** from the menu.
  - To delete a row or column, select the row or column, and then click **Delete**.
  - To clear row or column values, select the row or column, and then click **Clear Contents**.
  - To copy and paste multiple values, multiselect the row and column values, and then click **Copy**. Select one cell as a starting point to paste the values, and then click **Paste**.

## Edit measurements

1. Click the **Measurements** tab.

2. Click a table dimension table value (for example, [2\*2]) to display its value table.
3. Perform any of the following in the **Measurements** section:
  - To edit measurement values, select the row, and then click **Edit** .
  - To add a measurement, select the row after which you want to create the measurement, and then click **Add** .
  - To add files to a measurement, select the row, and then click **Attach Files** .
  - To duplicate measurements, select the row, and then click **Duplicate**.

## Managing parameters with parameter projects

### Parameter project lifecycle

You can use parameter projects to manage product parameters. Parameter values evolve from product target definitions to systems optimizations through several cycles of simulations. Therefore, at various product program milestones, you need to release and freeze the content of parameter values.

Products often require variant-specific parameter values. The parameter project through product configurator supports setting variant conditions on parameters.

You can also manage variance at the *parameter set* level to reduce the configuration complexity. You can therefore set variant conditions on parameter sets.

Because product systems such as engines or braking are reused across multiple product platforms, such as compacts or sedans, you can reuse parameters sets across parameter projects.

### Associating parameters with a project


You can capture parameters within a parameter project. Within the parameter project, you apply values to those parameter definitions. These parameter values are shared across the simulations and models within the parameter project. As the parameter project progresses, you can create new parameter definitions or evolve existing ones. At the end of the parameter project, you can compare and merge changes into the master location where the parameter definitions are managed.

Parameter projects can be used where there is no product structure or BOM context. In this case, the purpose of the parameter project is to manage parameter definitions, values, and configurations during their initial development phase.

A parameter project may also contain *sets*, which are used to collect and assign parameters to multiple users or teams. Users can only work in a project to which they are assigned.

## Create a parameter project

You can create an empty parameter project that will be used to contain all parameter sets and parameter values needed for your work.

1. Navigate to the folder where you want to store the parameter and click **More commands ... > New**  **Add**.

The **Add** panel appears.

2. In the **Type** field, filter and select **Parameter Project**.
3. Enter a name for the new parameter project, and click **Add**.

The new parameter project appears in the structure.

Note:

You can also create a new project by saving an existing project with a different name. Any child sets and referenced parameter dictionaries are carried over to the new project.

## Create a parameter project from another parameter project

You can duplicate an existing parameter project to create another project. By default, parameters are *not* duplicated along with their parameter sets. To change this behavior, set the deep copy rule in the BMIDE.

1. Open the parameter project, and then select the highest element in the structure that you want to duplicate.
2. Click **More commands ...**, and then select **Duplicate**.
3. To create duplicates of shared elements with new **ID** values, select the parent elements that you want to duplicate, and then right-click the element and select **Save As And Copy Children**.

All child elements duplicate as well with new **ID** values.

4. Click **Save**.

The **Save** panel appears.

5. Choose an **ID Naming Rule** option, and then click **Save**.

Active Workspace updates the element **ID** values based on the naming rule option you selected, and the parameter project is saved.

## Open a parameter project

You can open a parameter project to navigate its contents. If the project contains other projects, you can view the hierarchy in a tree format.

1. Select the parameter project in the navigation panel.

The Overview appears, displaying the object properties and the Parameter Dictionaries table.

2. (Optional) Open any of the following tabs to view information.
  - **Parameters** - Displays information about associated parameters.
  - **Attachments** - Displays information about objects attached to the parameter project.
  - **Where Used** - Displays information about where the selected parameter project is used.
  - **History** - Displays revision and change information about the selected parameter project.
  - **Relations** - Displays a diagram of relationships between the selected parameter project and other objects.
  - **Reports** - Displays a list of reports generated about the selected parameter project.
  - **Test Results** - Displays results and execution status of tests run using the selected parameter project.

## Create a new parameter set in a parameter project

You can create a parameter set within a parameter project to contain parameter values and (potentially) other parameter sets. A parameter set allows you to organize the parameter values of a large project into smaller sets. For example, one set may work on performance, while another set works on comfort.

1. Open the parameter project in the work area, and then click **Add** ⊕ > **Child** or **Add** ⊕ > **Sibling** in the navigation bar.

The **Add Child** or **Add Sibling** panel appears. The **Type** field is automatically populated with **Parameter Set**.

2. Enter a name for the new parameter set.

The parameter set name need not be unique within the parameter project.

3. Enter a **Number of Elements**. The default number is 1.
4. Click **Add**.

The new parameter set appears in the structure.

- (Optional) Open the parameter set and add parameters by clicking **Add** or **Quick Add**. For information on adding a parameter, see **Create a parameter in a project, set, or other location**.

Tip:

You can edit parameter names and values in the table.

Note:

You can also create a new set by saving an existing set with a different name. Any child sets are carried over to the new set

You cannot copy and paste sets. However, you can populate a set by copying parameters from elsewhere and pasting them into the set.

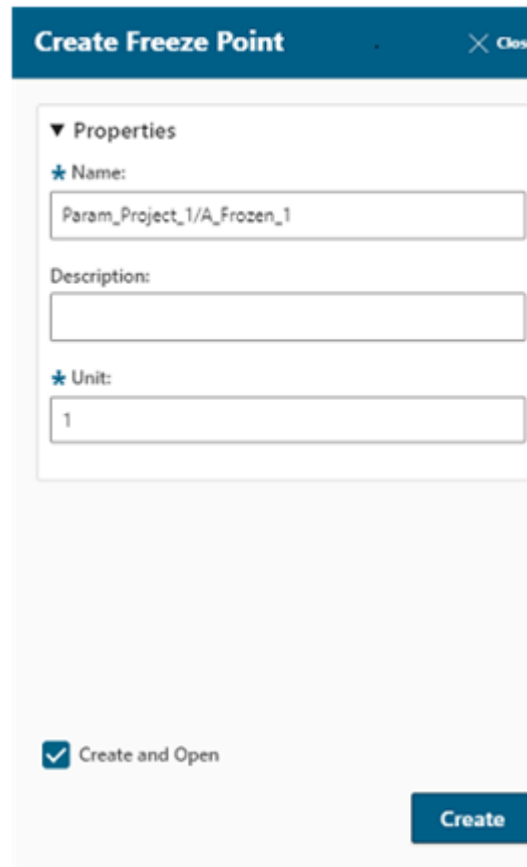
## Freeze parameter projects or elements

A *freeze point* represents a set of parameter values that you can restore at a later time. For example, you can use a freeze point to compare with other freeze point values at a later time. A freeze point is defined by a name, its end item, and a unit effectivity number.

Active Workspace freezes the entire structure from your selected element downwards, sets its release status to **Frozen**, and sets the shared effectivity to the selected value on the applicable BOM lines. It also creates a recipe that stores the frozen structure that you can later use to restore the frozen structure data. If you subsequently revise any elements or parameter values in the structure, Active Workspace unfreezes the structure (or substructure) and, if appropriate, you should create a new freeze point with a new name and unit number.

- Open the parameter project.
- In the panel, select an element that has associated parameters, and then click **More commands ...** > **New** > **Create Freeze Point**.

The **Create Freeze Point** panel appears.



**Create Freeze Point** Close

▼ Properties

★ Name:  
Param\_Project\_1/A\_Frozen\_1

Description:

★ Unit:  
1

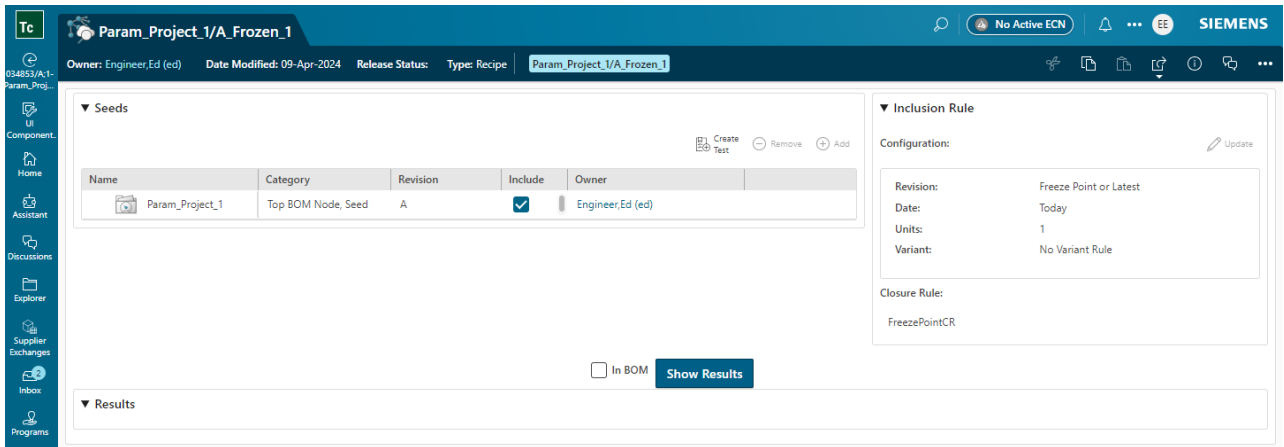
Create and Open

**Create**

3. (Optional) Update the name of the freeze point and the unit.
4. (Optional) Select the **Create and Open** check box if you want the freeze point to open immediately after it is created.
5. Click **Create**.

Active Workspace freezes the entire structure from your selected element and throughout all its child elements.

The following graphic shows a freeze point created in a selected parameter project.



## Associate a parameter dictionary with a parameter project

You can add one or more existing parameter dictionaries within a parameter project to provide a context when creating parameter values. A parameter dictionary can restrict the types of parameters that can be used in the project to filter unnecessary parameters.

1. Open the parameter project and then click **Add** ⊕ above the Parameter Dictionaries table.

The **Add** panel appears.

2. Find the parameter dictionary to add by searching or selecting it in the palette, and then click **Add**.

Active Workspace adds the parameter dictionary to the table and refreshes the work area.

## Comparing project parameters

### Performing tradeoff analysis

You can use the Parameters table to perform a tradeoff analysis and balance multiple attributes. You can compare:

- Different revision configurations of a project.
- Product parameters and verification requests.
- Product parameters and product freeze points.

For example, you can compare the parameter values on the latest item revision in a requirements structure with the values on parameter values on a verification request. Alternatively, you can compare it with previous revisions identified by a shared effectivity on their release status.

## Compare parameters

You can compare parameters from the **Parameters** table. Active Workspace displays a comparison table containing a preconfigured set of columns to compare. If necessary, you can add or remove columns by clicking **Settings** ⚙️ and using the **Hide/Show Columns** panel to select the required columns.

The parameters in the table are highlighted according to the following rules:

- The source parameter row is highlighted in grey.
- Parameters that do not have any values are highlighted with a yellow bar.
- Parameters that have changes in values are highlighted with a red bar.
- Parameters that have no changes are not highlighted.

1. In a specification, click **Details** > **Parameters** and click **Compare Parameters** 🗉.

The **Compare Parameters** page and the **Add Comparison** panel appear.

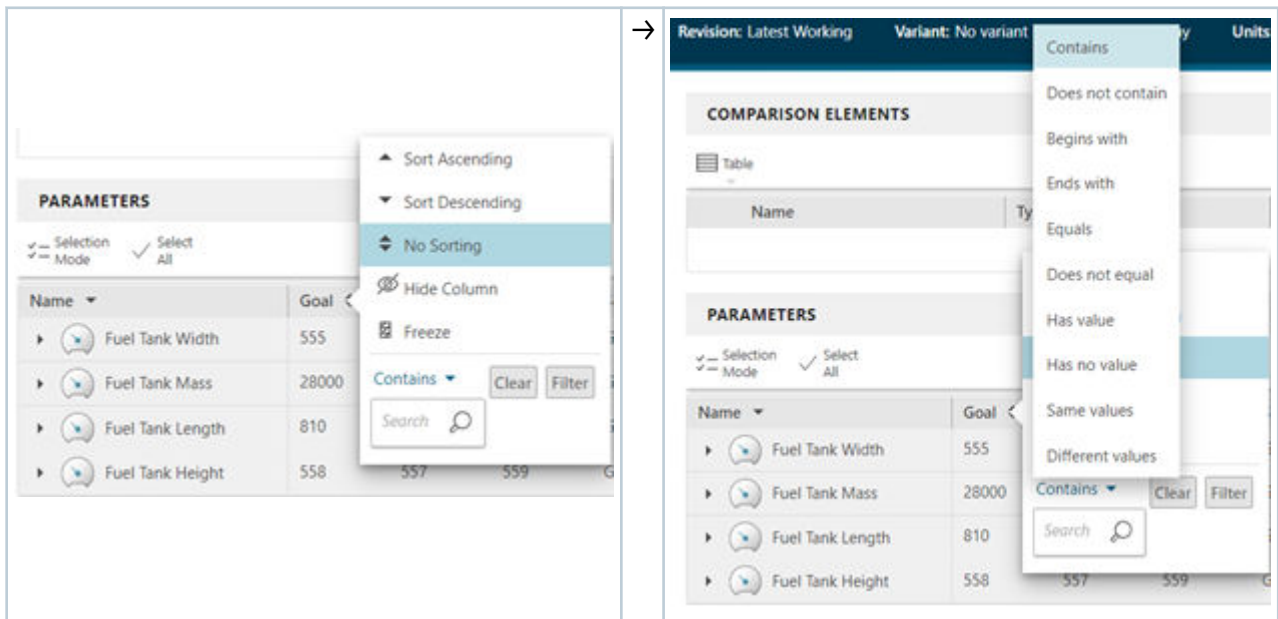
The screenshot shows the 'Compare Parameters' interface for 'Test\_Spec\_1'. The main table has the following data:

Name	Revision	Units	Measurement	Goal	Min	Max	Source	Description
Param1	A	meter		5	1	10	038890/A:1-Test_Spec_1	
Param2	A	m		10	5	15	038890/A:1-Test_Spec_1	

The 'Add Comparison' panel on the right shows the following options:

- Verification Request and Studies
- Recipe
- Revision Rule

2. (Optional) Click a comparison type on the **Add Comparison** panel or click **Add Comparison** ⊕ in the **Comparison Elements** section and choose from the following comparison elements:
  - **Verification Request and Studies.** After selecting this option, choose a verification request or study from the given list.
  - **Recipe.** After selecting this option, choose a recipe from the given list.
  - **Revision Rule.** After selecting this option, choose a revision rule from the given list.
3. (Optional) You can filter the results by clicking the arrow in a column and selecting a filter option.



The filters are of two types—regular filters and comparison filters.

By default, the comparison filters are available in the **Measurement**, **Goal**, **Min**, and **Max** columns. You can add or remove the columns that have comparison filters by updating the **PLE\_Parameters\_ComparisonColumns** preference.

#### Regular filters—only applied to source parameters on demand

- **Contains**
- **Does not contain**
- **Begins with**
- **Ends with**
- **Equals**
- **Does not equal**

#### Comparison filters

- **Has value**

Only show the child parameters with value in that column. The source parameters are displayed.

- **Has no value**

Only show child parameters with empty values in that column. The source parameters are displayed.

- **Same value**

Only show child parameters and source parameters whose values are the same. If the parameter values are different, no parameters are displayed.


- **Different values**

Only show child parameters and source parameters whose values are different. If the parameter values are the same, no parameters are displayed.


4. (Optional) Edit any of the values for the source parameter.

## Compare multidimensional table value parameters

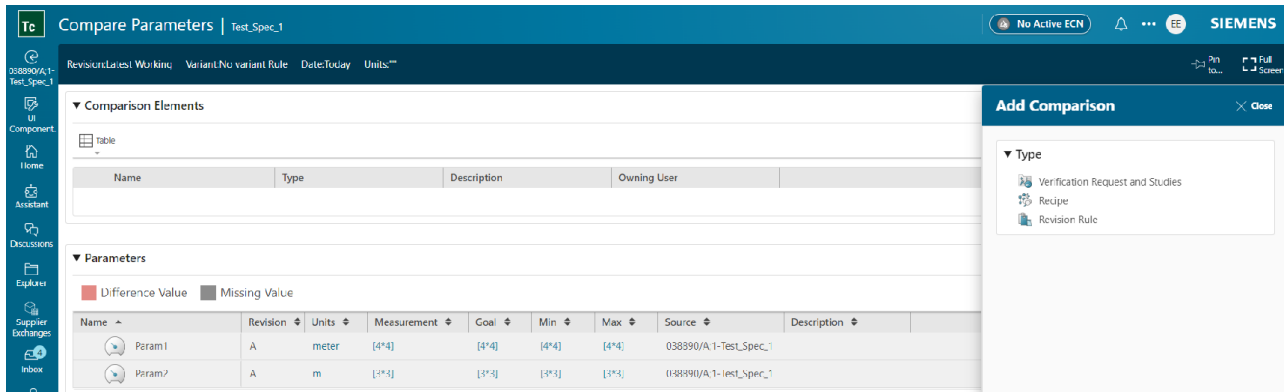
You can compare multidimensional table value parameters from the **Parameters** table. Parameters with multidimensional table values display these values as table dimensions in the format `[rows*columns]`.

When comparing multidimensional table value parameters, Active Workspace displays a comparison table containing a preconfigured set of columns to compare. If necessary, you can add or remove columns by clicking **Settings**  and using the **Hide/Show Columns** panel to select the required columns.

The parameters in the table are highlighted according to the following rules:

- The source parameter row is highlighted in grey.
  - Parameters that do not have any values are highlighted with a yellow bar.
  - Parameters that have changes in values are highlighted with a red bar.
  - Parameters that have no changes are not highlighted.
1. In a specification, click **Details** > **Parameters** and click **Compare Parameters** .

The **Compare Parameters** page and the **Add Comparison** panel appear.



2. (Optional) Click a comparison type on the **Add Comparison** panel or click **Add Comparison** ⊕ in the **Comparison Elements** section and choosing from the following comparison elements:
  - **Verification Request and Studies.** After selecting this option, choose a verification request or study from the given list.
  - **Recipe.** After selecting this option, choose a recipe from the given list.
  - **Revision Rule.** After selecting this option, choose a revision rule from the given list.
3. (Optional) You can filter the results by clicking the arrow in a column and selecting a filter option.

The filters are of two types—regular filters and comparison filters.

By default, the comparison filters are available in the **Goal**, **Min**, **Max**, and **Measurement** columns. You can add or remove the columns that have comparison filters by updating the `PLE_Parameters_ComparisonColumns` preference.

#### Regular filters—only applied to source parameters on demand

- **Contains**
- **Does not contain**
- **Begins with**
- **Ends with**
- **Equals**
- **Does not equal**

#### Comparison filters

- **Has value**

Only show the child parameters with value in that column. The source parameters are displayed.

- **Has no value**

Only show child parameters with empty values in that column. The source parameters are displayed.

- **Same value**

Only show child parameters and source parameters whose values are the same. If the parameter values are different, no parameters are displayed.

- **Different values**

Only show child parameters and source parameters whose values are different. If the parameter values are the same, no parameters are displayed.

4. Click a table value such as **[10\*6]** in the earlier example.

The value table dialog displays. The name of the dialog is the name of the parameter, such as **Ignition Map** in the earlier example.

For information about editing the **Values** or **Measurements** panel tables, see [Edit value and measurement tables](#).

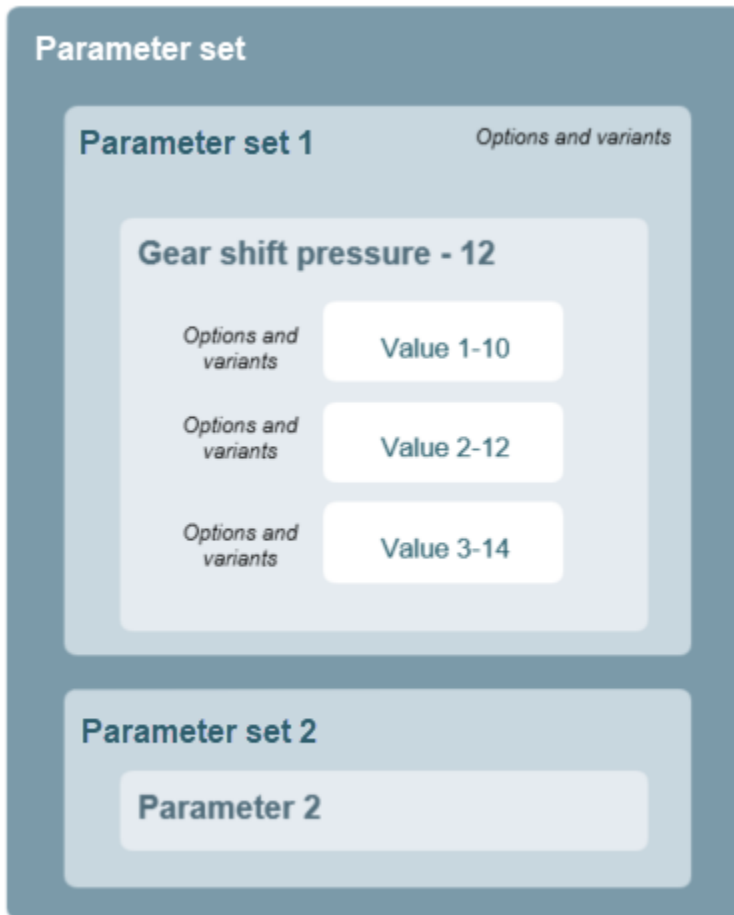
## Configuring parameters with variants

### Managing variants outside the product or program

#### Managing parameter variants outside the product or program

You can create variant expressions and resolve them against a parameter set or parameter project to obtain the parameter values. Parameter values are not always configured from the product perspective. They may be configured within a set or project of parameters where there is a common definition, but there are multiple parameter values associated to a different configuration. This allows you to manage parameters outside of the context of the product. The set or project of parameter values is resolved once a specific configuration (variant rule) is applied to the collection.

The following example shows how variants can be used within a parameter set.



Engineers typically perform studies to analyze the optimal parameter values for different values of the product. The typical process is:

1. Associate the project with a variant configurator context, which is created in the Product Configurator.

2. Run tests

Create parameter values under the appropriate set in the project. A set maps to a subsystem in the product.

3. Apply the required variant configuration from the Configuration panel.
4. Perform the test run and record the value.
5. Repeat steps 2 and 3 until you can declare one of the values for each variant to be final. The final value is considered during tradeoff analysis.

## Apply variant configuration to the parameter project

You can set a variant rule on a parameter project if the project is associated with a configurator context.

1. Open the parameter project you want to configure.
2. Click **More commands ... > Configure > Configuration**, and select the variant rule you want to apply.
3. Select **Parameter Variants**.

Active Workspace shows only those variants that are configured in the applied variant rule.

When you log off, the variant rule is saved to the project. When you reopen the project in a new session, the same configuration settings are applied. If you open a different project, its own variant rule is applied.

## Display and edit parameter variants in a parameter project

You can view and edit the parameter variants in a table within an existing parameter project.

1. Select the parameter project or one of its sets in the left-hand navigation panel.
2. Click the **Parameter Variants** tab.

Active Workspace displays the table of variants.

You can view the available variants in the table, and if appropriate, associate additional variants with the project.

Note:

You can also view multiple parameter variants by selecting them in the left-hand navigation panel. Each check mark shows a variant that is associated with a particular parameter.

## Managing parameter variants in the product or program


### Create parameter variant values

A parameter may not be valid for all configurations of a product. A parameter that is not valid for the current configuration is not shown in the parameter table. For example:


Variant Configuration	Family	Station Wagon	Sport
Parameter			
Torsional Stiffness	[Blank]	X	X
Tire Pressure	X	X	X

1. In the left-hand panel, select the BOM line context, and apply the relevant variant configuration.
2. Click **Details > Parameters**, and verify if the **Parameters** table shows the necessary parameters.
3. To add a new parameter in the current configuration, select an existing parameter, and then click **Add ⊕ > Add > Add Parameter** .

Active Workspace adds a new parameter to the parameter table, with the same values as the selected parameter.

4. Update the values (typically, **Goal**, **Min**, and **Max**) of the new parameter by clicking the relevant cell and entering the new value.
5. Click **Save Edits**  to save the new values for the parameters.


### Apply and save variant conditions on parameters

1. In the left-hand panel, select the parameters on which you wish to apply a variant condition.
2. Click **Variants** and then select the parameters to which you want to apply each variant condition. If you click in a table cell until a check mark ✓ shows, the variant condition is applied to the parameter. You can click again and clear the cell to remove the variant condition.
3. Click **Save**  to save the variant conditions on the parameters.

### View variant expressions for variant parameters

1. In the left-hand panel, select one or more occurrences and then click the **Variant Conditions** tab.


Active Workspace shows the variants matrix.

2. Click **Show Parameters**  to display the list of parameters in the variants table.

Active Workspace shows columns for each parameter across the top of the matrix, together with the occurrence columns.


3. (Optional) Associate the parameters with the relevant product model families by clicking in the intersecting cells until check marks ✓ are shown.

If you make changes, click **Save** .

Variant conditions are indicated by a  symbol in the relevant rows. If you hover the mouse over this symbol, information about the relevant variant condition displays.

## Add parameters to a verification request

You can add parameters to a verification request to determine if the parameter values are calibrated.

1. Navigate to the parameter project containing the parameters of interest.
2. Select the parameters whose values you want to verify and then choose **New**  **> Create Test**. The parameters must be set to **Output** and the **Goal**, **Min**, and **Max** values defined.

Active Workspace displays the **Test** panel.

3. Select the type of verification request to create.

Active Workspace refreshes the **Create Test** panel with fields for the required and optional properties.

4. Enter the remaining properties and the owning project (if any), and then click **Create**.

The inclusion rule specifies the elements and parameters to include on the verification request. The available inclusion rules are defined by your administrator. If you select **No Rule**, the verification request contains only the objects you selected.

Active Workspace creates the defined verification request.

5. Navigate to the **Overview** panel of the verification request and set the parameters to **Output**.
6. Run the necessary verifications and then set the **Result** property on the **Overview** panel to **Pass** or **Fail**.

Active Workspace shows a roll-up chart of the fail/pass status in the **Parameter Result** panel.

7. Select one of the sections (**No Result**, **Pass** or **Fail**) of the **Result** chart.

The **Parameters** list is updated to show only those parameters that have passed verification.

8. Select the **Test Requests** panel of the verification request.

Active Workspace shows a list of test requests included in the verification request and a chart of their current pass/fail status.

9. Select one of the test requests in the table.


(You can also select multiple test requests but the results will load more slowly.)

Active Workspace shows the parameters for the selected test request or test requests in the **Test Request Parameters** panel at the bottom of the page.

10. Select one of the blocks in the **Test Request Result Chart**, for example, the **Fail** block for a particular test.

Active Workspace filters the contents of the **Test Request Parameters** parameters panel to show only parameters associated with the selected block.

Note:

To remove a selected parameter from a verification request, choose **Edit**  > **Remove from Verification Request**.

## Revising parameters

### Configuring revisions of parameters

To track, manage, and configure the correct set of parameters, you must understand their revision history. Different revisions of parameters may have different revision rules, release status, effectivity, and states, for example, open, closed, or released. You can revise parameters in projects, groups and products.

A specific revision of a common parameter is typically associated with a product context, system block, item revision or occurrence. You can create a parameter in any of those locations or the **Home** folder, but can use it elsewhere if appropriate. For example, different revisions of a weight parameter may have different values, depending on which block it is associated with. When you select a revision rule in the **Configuration** panel, the correct revision of each parameter is configured in the parameter project.

Note:

Not all revision rules can be used when configuring a parameter project or compare parameters. Revision rules for parameters must be:

- Visible to users.
- Not suppressed.
- Contain at least one of the following entries: **Working()**, **Has Status()**, and **Latest()**.

Changes made to a particular revision do not affect the corresponding parameter definition in the dictionary.

A revision of a parameter does not inherit the access rights of its parent, that is, you can modify the revision even if the parent is released.

## Revise a parameter value

1. Select an individual parameter in the project or in an assembly, and then click **More commands ...** > **New** ✨ > **Save As or Revise**.

The **Save As** panel appears.

2. (Optional) Select the **Baseline** check box if you want create a baseline.

Select the **Precise Baseline** check box if you want the baseline to be precise.

From the **Template** list, select a baseline template.

3. (Optional) Select **Open New Revision** to display it in location once created. If you do not select this check box, the new revision is added to the parameters table where you can edit values as appropriate.

4. Click **Save**.

The system creates a new revision of the selected parameter and displays its **Overview** panel.

5. If you are in an assembly, from the **Parameters** tab, select the parameter and click **Replace Revision** ⇄.

In the **Replace Revision** panel, choose a revision and click **Replace Revision**.


The system replaces the previous revision. You can change this manually by applying a different configuration (revision rule and variant rule) to the project.

6. Edit the values of the new parameter revision as appropriate.

Note that all previous revisions of the parameter remain in the system and you can retrieve them at any time, for example, to compare two revisions.

To view all revisions of a parameter and their attributes, select the **History** panel.

### Revise and replace

You can also revise a selected parameter and replace the existing revision in the product with the new revision by clicking . This action does not affect other occurrences of the existing revision. However, you can click ⇄ to display the **Replace Revision** dialog box where you can select other revisions to replace with the latest.

You can also revise multiple selected parameters in the same way but must confirm this action by clicking **Revise and Replace** in the confirmation dialog box.

## Importing and exporting parameters for a project

### Sharing parameters with external applications

Parameter definitions and values can be exported to or imported from external applications such as Mentor Capital Harness. The export and import processes follow a common pattern for all external applications. The data can be exported and imported in a variety of standard formats, for example, Fibex, Microsoft Excel, or A2L. You can also provide your own conversion utility to exchange information in a custom format such as that required by Simulink.

Many applications consume parameters in Excel format. In this case, the parameters contain the name, value, units, and properties necessary for the consuming application to read the parameters and then associate them with the appropriate model, simulation, calculation or other analysis process. You can export parameters from Active Workspace in Excel format and then import them into the consuming application. Likewise, you can export parameters from the application, and then import them in Excel format into Active Workspace. You typically only export or import a selected number of parameters, typically ones associated with a validation request, project or group.

If the application has its own method of managing parameters or a library or dictionary of parameters, there are special considerations. For example, Mentor Capital Harness has the concept of an OTI (Object Type Information) which is a set of properties defined against an object type such as a component or device. Once the object is instantiated within a Capital Harness design it inherits the OTI and you can then assign a value to the design object subject to any constraints from the OTI. To maintain a consistent set of parameter definitions across systems, the definition must be common, shared, or synchronized between Active Workspace and the application.

By default, attached goal and measurement files are not exported with parameters, nor are measurement files imported during a round-trip process. The administrator can configure export and import of attached files by adding an **Include Attachments** entry in the **Excel Template Rules** field of the **Excel Template Properties** dialog box during installation. The following two lines should also be added to the **Parameter\_in\_product\_Excel\_CR** closure rule:

Primary	Primary Object	Secondary	Secondary Object	Relation Type	Related Property or Object	Action Type
CLASS	Att0MeasureableAttribute	CLASS	Dataset	RELATIONP2S	Att0HasGoalFile	PROCESS
CLASS	Att0MeasureValue	CLASS	Dataset	RELATIONP2S	Att0HasMeasurementFile	PROCESS

The dataset object must also be present in the list of objects to export (through selection or closure rule parsing) for attachments to be exported.

Note:

When importing parameters in a package:

- The package may contain multiple Excel files, one main file and as many goal and measurement files as necessary.
- The package may contain Excel files in XSLX and XSLM formats, including the main file, and the goal and measurement files.
- The root of the package may contain only one Excel file, the main file. Each goal and measurement file must be in a sub folder.


## Importing parameter definitions

### Importing parameter definitions from Microsoft Excel

As a parameter designer, you can import parameter definitions from a Microsoft Excel file into a parameter dictionary. If the parameter definitions are new, they are added to the dictionary. If the parameter definitions already exist, for example, because the Excel file was exported with the round-trip export option, they are updated from the contents of the Excel file.

#### Import new parameter definitions

Before performing this procedure, ensure the **import template for parameter definitions is correctly configured** for your data. When you import parameter definitions, if a parameter definition name does not already exist in your parameter dictionary, Active Workspace automatically names the parameter definition abased on the **Parameter Definition Name** cell in your spreadsheet.

1. Select a parameter dictionary and then click **Excel Round Trip**  > **Import**.

Active Workspace displays the **Import** panel.

2. Click the **XLS template** hyperlink on the **Import** panel if you want to download the template and add parameter definitions to it.
3. Add the new parameter definitions to the Excel file exported in the previous step and then save the file.
4. Drag the Excel file from Windows Explorer into the **Choose File** dialog box in the **Import** panel, click **Open**.
5. Select the **Release Parameter Definitions on Creation** check box to import the parameter definitions, and then automatically release them without having to manually start a workflow.
6. Click **Import**

The system imports the Excel file and the parameter definitions in it are added to the parameter dictionary. If you selected the auto-release option, then the parameter definitions are released as well.


If the system cannot recognize or read the Excel file (for example, because it is in an incorrect format), it shows an error message, otherwise it shows a success message.

### Import parameter definitions with round trip

Before performing this procedure, ensure that the **import template for parameters is correctly configured** for your data.

1. Export the parameter definitions from a parameter dictionary into an Excel file.
2. Update the existing parameter definitions in the exported Excel file and then save the file.

To add a new parameter definition, right-click the row containing the definition and click **Insert...** to add the new definition.

3. Select the parameter dictionary and then click **Excel Round Trip**  > **Import**.

Active Workspace displays the **Import** panel.

4. Drag the Excel file from Windows Explorer into the **Choose File** dialog box in the **Import** panel and click **Open**.
5. Select the **Auto-create and Release Parameter Definitions** check box if you want to create and release the parameter definitions.
6. Click **Import**.

The system imports the Excel file and the parameter definitions in it are added to the parameter dictionary.

### Importing parameters

#### Importing parameters

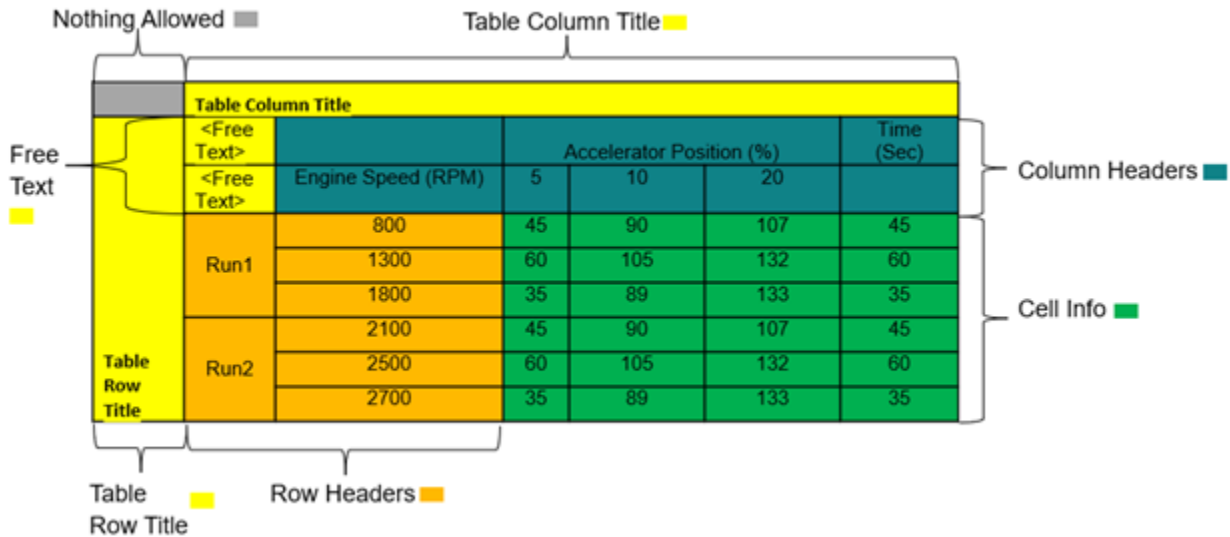
You can import parameter values from a Microsoft Excel file into a parameter project.

Consider the following when importing parameters:

- If the imported parameters already exist, their values are updated accordingly; otherwise, Active Workspace creates the parameters.
- If an imported parameter does not have a corresponding parameter definition, Active Workspace automatically creates the parameter definition after import. The parameter definition is given the same name as the imported parameter.

## Formatting Microsoft Excel files for importing parameters

You must have a properly formatted Microsoft Excel file before starting the import process.



The following graphic shows a simple braking goal value table.

Speed (mph)	Braking Distance (m)	Reaction Distance (m)	Stopping Distance (m)
10	5	20	25
15	11	31	42
20	19	42	61
25	30	53	83
30	43	64	107
35	59	75	134
40	76	86	162
45	97	97	194
50	119	108	227
55	144	119	263
60	172	130	302
65	202	141	343
70	234	152	386
75	268	163	431
80	305	174	479
85	345	185	530

You can also define values such as **Goal**, **Min**, and **Max** values in the table as shown in the following graphic.

Note:

When importing **multidimensional parameter values**, you must use the headings **Goal**, **Min**, and **Max** in the imported file.

Speed (mph)	Braking Distance (m)			Reaction Distance (m)			Stopping Distance (m)		
	Goal	Min	Max	Goal	Min	Max	Goal	Min	Max
10	5	1	8	20	5	22	25	9	30
15	11	2	20	31	6	33	42	10	53
20	19	3	25	42	7	44	61	11	69
25	30	4	35	53	8	55	83	12	90
30	43	5	45	64	9	66	107	13	111
35	59	6	60	75	10	77	134	14	137
40	76	7	85	86	11	88	162	15	173
45	97	8	100	97	12	99	194	16	199
50	119	9	200	108	13	110	227	17	310
55	144	10	150	119	14	121	263	18	271
60	172	11	180	130	15	132	302	19	312
65	202	12	210	141	16	143	343	20	353
70	234	13	235	152	17	154	386	21	389
75	268	14	270	163	18	165	431	22	435
80	305	15	310	174	19	176	479	23	486

Separate files can be used to store associated goal and value information. The goal and value files must be listed in the appropriate column of the Excel file. This graphic shows an example of the files *goal.txt* and *value.txt* listed in the Excel file to be imported.


Measured Value	Goal File	Value File
Value	File Name	File Name
	goal.txt	value.txt

Prior to import, the goal and value files must be packaged in a .zip file, along with the Excel file to be imported. This graphic shows an example of a .zip file to be imported into Teamcenter.


Name	Date modified	Type	Size
value.txt	4/22/2024 3:26 PM	Text Document	1 KB
Test_Engine_Parameters.zip	4/22/2024 3:28 PM	Compressed (zipp...	18 KB
Test_Engine_Parameters_with_Assoc_files.xlsx	4/22/2024 4:29 PM	Microsoft Excel W...	22 KB
goal.txt	4/22/2024 3:26 PM	Text Document	1 KB

## Import new parameters

Consider the following:

- For regular parameters, ensure that the **import template for parameters is correctly configured** for your data.
  - For table value parameters, include **the headings Goal, Min, and Max in the imported file.**
1. If you are in a parameter project, select the parameter project or and then click **Excel Round Trip**  > **Import**.

OR

If you are in an assembly or workitem, click **Details** > **Parameters** tab and then click **Excel Round-trip**  > **Import from Excel**.

The **Import** panel appears.

2. Click the **XLS template** hyperlink on the **Import** panel.

Active Workspace downloads the Excel template for parameters.

3. Add the new parameters to the Excel file exported in the previous step and then save the file.
4. Drag the Excel file from Windows Explorer into the **Choose File** dialog box in the **Import** panel, click **Open**.
5. Select the **Auto-create and Release Parameter Definitions** checkbox if you want to create parameter definitions that do not exist and release the parameter definitions. The check box is selected by default.
6. Click **Import**

The system imports the Excel file and the parameter definitions in it are added to the parameter dictionary.

If the system cannot recognize or read the Excel file (for example, because it is in an incorrect format), it shows an error message, otherwise it shows a success message.

## Import parameters with Microsoft Excel round trip

When you perform a round-trip import of parameters using Microsoft Excel, consider the following:


- For regular parameters, ensure that the **import template for parameters is correctly configured** for your data.

- For table value parameters, include **the headings Goal, Min, and Max in the imported file.**
1. Export the parameters from a parameter project into an Excel file.
  2. Update the existing parameter definitions in the exported Excel file and then save the file.

To add a new parameter, right-click the source row and then click **Insert...** to add the new row. The exported Excel file may contain multiple rows for **Source**. Each row represents the parent of the parameters under it.

**Note:**

For the **Parameter Definition** column cells, ensure that you enter the parameter definition name *only*; do *not* include the parameter ID. Otherwise, you receive an import error.

3. Select the parameter project and then click **Excel Round Trip**  > **Import**.

Active Workspace displays the **Import** panel.

4. Drag the Excel file from Windows Explorer into the **Choose File** dialog box in the **Import** panel and click **Open**.
5. Select the **Auto-create and Release Parameter Definitions** checkbox to create parameter definitions that do not exist, and release the parameter definitions.
6. Click **Import**.

The system imports the Excel file.

- For existing parameters, it updates the modified properties. It cannot add them to the current parameter project, assuming these parameters were exported from a different parameter project.
- For new parameters, the system creates and adds them to the source defined in the Excel file.

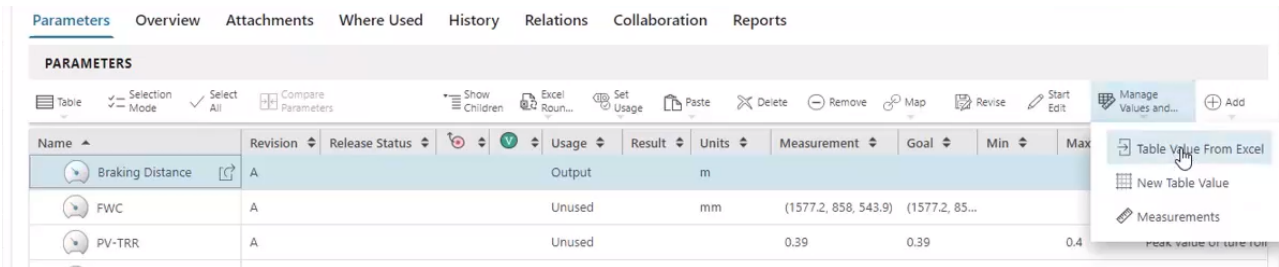
## Import table parameter values from Microsoft Excel

You can import multidimensional table value data from Microsoft Excel into a parameter. Before performing this procedure, include **the headings Goal, Min, and Max in the imported file.**

1. Select the parameter project and then select the parameter for which you want to import the multidimensional parameter data.
2. Select **Manage Values and Measurements > Table Value From Excel**.

The **Open** dialog box appears.

The following graphic shows the user preparing to import Microsoft Excel table data values into the **Braking Distance** parameter.



3. Select the Microsoft Excel file that contains the table data, and then click **Open**.

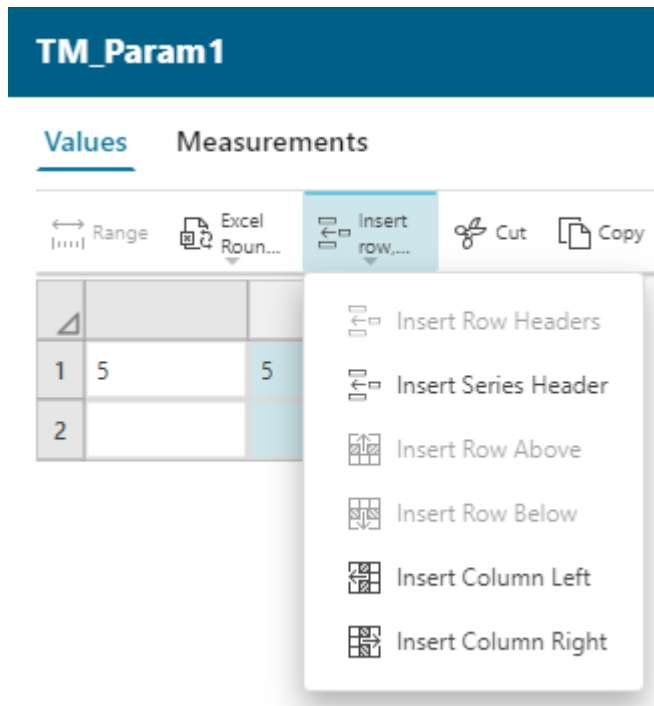
A dialog named after the parameter opens. In the following graphic, the user has imported the value table for the **Braking Distance** parameter. The **Braking Distance** dialog opens and shows the **Values** tab and imported value table.

Speed (mph)	Braking Distan	Reaction Dista	Stopping Dista
10	5	20	25
15	11	31	42
20	19	42	61
25	30	53	83
30	43	64	107
35	59	75	134
40	76	86	162
45	97	97	194
50	119	108	227
55	144	119	263
60	172	130	302
65	202	141	343
70	234	152	386
75	268	163	431
80	305	174	479
85	345	185	530
90	386	196	582

4. Perform any of the following with the value table:

- To modify table values, click **Start Edit**.
- By default, the table displays the Goal values. To see the Min and Max values as well, click **Range**.
- To modify the row or column structure, select a row or column and then select an action such as **Insert Row Above** or **Insert Column Left** from the menu.

In the follow graphic, the user is preparing to insert a row below the **90 mph** row.



- To delete a row or column, select the row or column, and then click **Delete**.
- To clear row or column values, select the row or column, and then click **Clear Contents**.
- To edit table values, click **Start Edit**.
- To copy and paste multiple values, multiselect the row and column values, and then click **Copy**. Select one cell as a starting point to paste the values, and then click **Paste**.

#### 5. Click **Import**.


The system imports the Excel file.

- For existing parameters, it updates the modified properties. It cannot add them to the current parameter project, assuming these parameters were exported from a different parameter project.
- For new parameters, the system creates and adds them to the source defined in the Excel file.

## Export parameter definitions from the parameter dictionary

As the parameter designer, you can export the parameter definitions from the selected parameter dictionary into a Microsoft Excel file.

Before performing this procedure, ensure the **export template for parameter definitions is correctly configured** for your data.

1. Select the parameter dictionary, and then choose **Excel Round Trip**  **Export to Excel**.

Active Workspace displays the **Export to Excel** panel.

2. In the **Settings** section, select **Include ID as Hyperlink** check box to make the IDs hyperlinks.
3. Select **Template**, choose the required template from the drop-down list. By default, the **ParameterDefinition\_template** template is selected.
4. Select the **Run in Background** check box if you want to continue with your work while the export takes place.
5. Click **Export**.

The system creates an Excel file containing the parameter definitions under the selected parameter dictionary in a folder on your local workstation. The parameters are *packed*, that is, all properties for a given parameter are in a single row.

## Configure an export template for parameters

Teamcenter includes a default template for exporting parameters into a Microsoft Excel file. This template contains default parameter properties and parameter types.

You can add custom parameter properties to the template.

**Note:**

Any addition of custom parameter properties apply to import functionality only. These changes do not apply to export functionality.

The following image shows a custom property named *Description1* added to the template.



The following image shows an example of both a custom property named *Description-Detailed* and a custom type named *MeasurableAttributePnt* added to the template.





Parameter											
Release Status	Description	Description-Detailed	Parameter Definition	Data Type	Units						
(Read Only)			(Read Only)	(Read Only)	(Read Only)						
as header or constant information. But if the header cell has a format (%Real Property Name), property during sheet generation. cedence. s to be exported for each matching rule. adsheet. information will be taken and applied to the generated excel sheet.											
							Level	Relation	Type	TransferMode	
{%last_release_status}	{%object_desc}	{%object_desc_detailed}		{%att1AttrType}	{%att0Uom}	<rule>			{%S-Att0MeasurableAttributeInt}	{%-Parameter_in_product_Excel_TM}	
						<rule>			{%S-Att0MeasureValueInt}	{%-Parameter_in_product_Excel_TM}	
{%last_release_status}	{%object_desc}	{%object_desc_detailed}		{%att1AttrType}	{%att0Uom}	<rule>			{%S-Att0MeasurableAttributeDbI}	{%-Parameter_in_product_Excel_TM}	
						<rule>			{%S-Att0MeasureValueDbI}	{%-Parameter_in_product_Excel_TM}	
{%last_release_status}	{%object_desc}	{%object_desc_detailed}		{%att1AttrType}	{%att0Uom}	<rule>			{%S-Att0MeasurableAttributeBool}	{%-Parameter_in_product_Excel_TM}	

### Restrictions and limitations

In the Excel file, you must leave one empty column between the properties table on the left and the rules table on the right.

AE	AF	AG	AH
<b>Value File</b>			
File Name			
			<b>Level</b>
		<rule>	
		<rule>	

### Procedure

1. Click **Explorer**  and navigate to the folder where you want to create the template.
2. On the primary toolbar, choose **More Commands**  > **New**  > **Add** .

The **Add** panel appears.

3. Click the **Search** tab and use the in-context search box to locate **Parameter\_template**.
4. Select the Excel template and click **Add**.

The new template object appears in the structure.

5. Open the template object, click **Attachments** and download the template file.
6. (Optional) To add a custom parameter property, open the downloaded file and do the following:
  - Right-click the table to the left and click **Insert** to add a column in the desired location.

- Enter the parameter property in row 5, for example, AttOMeasurableAttribute#object\_desc1.
- Enter the name of the property in row 6, for example, Description1.
- Enter the data to be exported in curly braces and preceded by a percent symbol, for example, {%object\_desc1}.

Refer to the BMIDE for information on property names.

7. (Optional) To add a custom parameter type, open the Microsoft Excel file and do the following:
  - Right-click the table to the right and click **Insert** to add a row in the desired location.
  - Enter **<rule>** in the yellow-highlighted cell.
  - Enter a name for the custom type in the **Type** column.
  - Enter information in the **TransferMode** column.
8. Enter values for all properties in the left-hand table.
9. Save the Microsoft Excel file to keep all changes.

You can rename the Excel file, if desired.

10. In **Attachments**, click **Add to** .

The **Add** panel appears.

11. Click **Choose File**, navigate to the Microsoft Excel file location, and click **Open**.
12. In the **Type** field, filter and select **MS ExcelX**.
13. Click **Add**.

The modified template file appears in **Attachments**.

Note:

You can delete the original template file from **Attachments**, retaining only the modified copy.

## Configuring Excel templates

### Configure import template for parameter definitions

As a parameter designer, you can create your own Excel template for importing parameter definitions, as follows.

1. Create a Excel template file with the name **My\_ParameterDefinition\_import\_template.xlsm**, based on the default **ParameterDefinition\_import\_template** template file in Teamcenter.

**Name** and **Data Type** are mandatory properties.

You can add columns to or remove columns from the Excel template as necessary. If you add a new column, you must provide a valid internal name in row 5. For example, the internal name for the **Name** column is **Att0AttributeDefRevision#object\_name**.

	Att0AttributeDefRevision#object_name
5	
6	<b>Name</b>

2. Create an MS Excel dataset in Teamcenter with the name **My\_ParameterDefinition\_import\_template** and upload the Excel template file to this dataset.
3. Update the **PLE\_Parameter\_Definition\_Import\_Excel\_Template** preference with the value **My\_ParameterDefinition\_import\_template**.

After completing this procedure, when you click the **XLS Template** hyperlink on the **Import** panel for parameter definitions, the system creates an Excel file based on the **My\_ParameterDefinition\_import\_template** template.

### Configure import template for parameters

As a parameter designer, you can create your own Excel template for importing parameters, as follows.

1. Create a Excel template file with the name **My\_Parameter\_import\_template.xlsm**, based on the default **Parameter\_import\_template** template file in Teamcenter.

**Name** and **Parameter Definition** are mandatory properties.

You can add columns to or remove columns from the Excel template as necessary. If you add a new column, you must provide a valid internal name in row 5. For example, the internal name for the **Name** column is **Att0MeasurableAttribute#object\_name**.

	Att0MeasurableAttribute #object_name
5	
6	<b>Name</b>

2. Create an MS Excel dataset in Teamcenter with the name **My\_Parameter\_import\_template** and upload the Excel template file to this dataset.
3. Update the **PLE\_Parameter\_Import\_Excel\_Template** preference with the value **My\_Parameter\_import\_template**.

After completing this procedure, when you click the **XLS Template** hyperlink on the **Import** panel for parameters, the system creates an Excel file based on the **My\_Parameter\_import\_template** template.

### Configure export template for parameter definitions

As administrator, you can create your own Excel template for importing parameter definitions, as follows.

1. Create a new Excel template file with the name **My\_ParameterDefinition\_template.xlsm**, based on the default **ParameterDefinition\_template** template file in Teamcenter.

**Name** and **Data Type** are mandatory properties.

You can add columns to or remove columns from the Excel template as necessary. If you add a new column, you must provide a valid internal name in row 5. For example, the internal name for the **Name** column is **Att0AttributeDefRevision#object\_name**.

	Att0AttributeDefRevision#objec t_name
5	
6	<b>Name</b>

2. Create a new Excel export template using the **add\_req\_templates** utility, for example:

```
add_req_templates -u=Tc-admin-user -p=password -g=group
-i=My_ParameterDefinition_template.xlsm -t=ExcelTemplate
```

3. Add the **apply\_packing** and **parameter\_templates** template rules to the new Excel template using the **set\_excel\_template\_rules** utility, for example;

```
set_excel_template_rules -u=Tc-admin-user -p=password -g=group
-templates=My_ParameterDefinition_template -rules=apply_packing,parameter_templates
```

After completing this procedure, you can find **My\_ParameterDefinition\_template** in the dropdown list of templates in the **Export to Excel** panel for the parameter dictionary.

### Configure export template for parameters

As administrator, you can create your own Excel template for exporting parameters, as follows.

1. Create a new Excel template file with the name **My\_Parameter\_template.xlsm**, based on the default **Parameter\_template** template file in Teamcenter.

**Name** and **Parameter Definition** are mandatory properties.

You can add columns to or remove columns from the Excel template as necessary. If you add a new column, you must provide a valid internal name in row 5. For example, the internal name for the **Name** column is **Att0MeasurableAttribute#object\_name**.

	Att0MeasurableAttribute #object_name
5	
6	<b>Name</b>

2. Create a new Excel export template using the **add\_req\_templates** utility, for example:

```
add_req_templates -u=Tc-admin-user -p=password -g=group -i=My_Parameter_template.xlsm
-t=ExcelTemplate
```

3. Add the **apply\_packing** and **include\_attachments** template rules to the new Excel template using the **set\_excel\_template\_rules** utility, for example;

```
set_excel_template_rules -u=Tc-admin-user -p=password -g=group
-templates=My_Parameter_template -rules=apply_packing,parameter_templates,
include_attachments
```

After completing this procedure, you can find **My\_Parameter\_template** in the dropdown list of templates in the **Export to Excel** panel for the parameter project.

# 7. Managing characteristics

## About characteristics

### Characteristics definition

The Model-Based Characteristics (MBC) standard defines the nomenclature, definitions, symbols, data structures, and practices for identifying, communicating, and exchanging model-based characteristics. It provides a common approach for tagging and uniquely identifying product characteristics essential for the product realization process. This standard supports product characteristic tagging for verification, acceptance, change control, non-conformance reporting, and protecting confidential information. It includes optional augmentations for criticality classifications, product requirement associations, and verification plan requirements. Characteristics are defined using the industry standard DMSC (Digital Metrology Standards Consortium) methodology.

Characteristics information can include traits, qualities, or attributes belonging to an element of a feature. This information can include size, location, form, or property. These characteristics can define a specification limit and a dimension with tolerance.

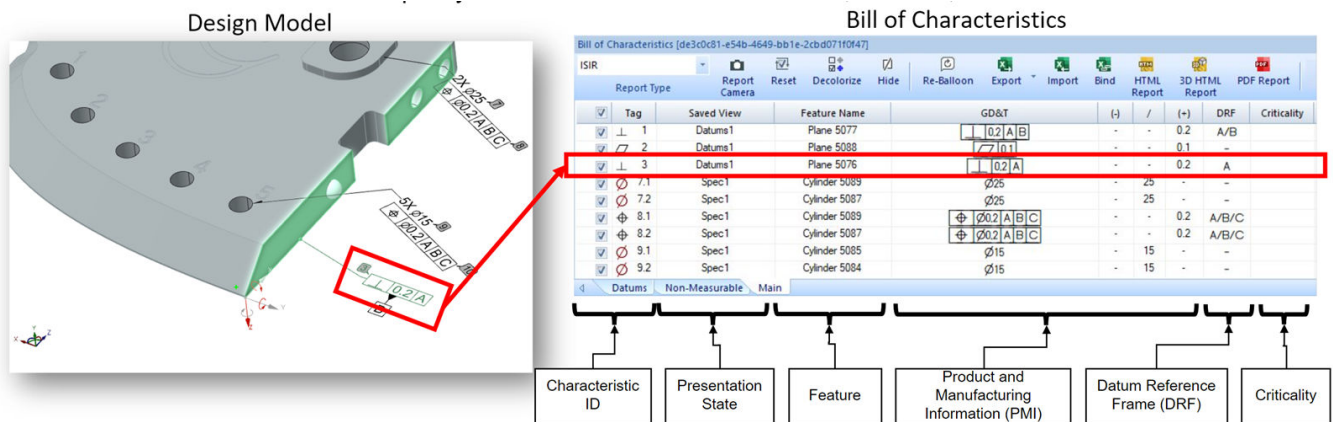
Teamcenter supports all types of characteristics defined in the DMSC. The following are examples of some supported characteristics:

<b>Flatness</b>	Defines the uniformity of the surface using a tolerance value and datum reference frame.
<b>Cutoff</b>	Defines the surface finish information.
<b>Linear Dimension</b>	Defines the straight-line distance between two points on a design using nominal, upper, and lower values.
<b>Note</b>	Captures relevant text information relating to a characteristic.

### Characteristics and CAD designs

Characteristics data is part of Product and Manufacturing Information (PMI) and is applied to the 3D CAD designs in NX through NX Inspector. This tool allows for the creation, import, and editing of the characteristics data. It allows you to view and manage the authored characteristics in the context of the displayed part in NX.

The following example shows a 3D model with characteristics data applied to its features.



The characteristics are defined in the **Bill of Characteristics**. When the design is saved to Teamcenter, these characteristics are also saved along with the part data.

## The process for saving characteristics to Teamcenter

When the part is saved to Teamcenter, the characteristics data is included with the part and is displayed under **Characteristics** when viewing the part's properties. The **Characteristics** area displays all the characteristics applied to the design. You can view the details for individual characteristics, but you cannot edit them. Any changes to the characteristic data must be done in the CAD design and published using NX Inspector. When the part is saved, the updated characteristics are displayed in Teamcenter.

The following steps detail the journey of characteristics from the CAD design to Teamcenter:

1. The CAD designer defines the characteristics as PMI in the engineering requirements for a CAD part feature.
2. Using **NX Inspector**, the CAD designer derives the details of a given PMI into separate characteristics.

These derived characteristics are saved with the part data.

3. When the CAD designer saves and publishes the design data to Teamcenter, the associated characteristics data is published along with the design.
4. During the save and publish process, the characteristics details are saved to a separate dataset for the part. This dataset is called **the characteristics file**.
5. A Teamcenter process reads this file and interprets each characteristics line item into a unique characteristic related to the PMI, feature, and part.
6. Once the data is processed, Teamcenter displays the characteristics data in the **Characteristics** area for the part.

- Other processes, such as Teamcenter Quality and Manufacturing, access and use the defined characteristics in their plans.

## The characteristics file

The **Characteristics File** is stored as an attachment for a part that contains characteristics data published to Teamcenter by **NX Inspector**.

This file contains the following data:

**CHX\_PART\_F** This file stores the part data authored in the NX CAD application.

**ILE**

**IPXML** This file is generated by **NX Inspector** from the authored NX part data.

**JSON** This file is authored along with the previously mentioned files and is included with the characteristics data when it is published to Teamcenter.

To view the view characteristics file for a part with characteristics data, select **Attachments**.

The screenshot shows the Teamcenter interface for part 034483/A;1-Part1. The 'Attachments' tab is selected, displaying a list of files. The file 'Chx0ModelBasedDS' is highlighted, with details: Characteristics File, Type: Model Based Characteristics, Owner: ed (ed), and Date Modified: 21-Oct.

## Characteristics prerequisites

The following prerequisites must be met for Teamcenter to display characteristics information.

- Your administrator has installed the **Parameter Management** and **Characteristics Indexer** using Deployment Center. This installs the necessary components for characteristics support.
- Your administrator has configured the Dispatcher service to recognize and add characteristics information published for a CAD design.
- CAD engineers have authored and published the characteristics data for the part using **NX Inspector**.

## View characteristics data for a part

You can view characteristics data for a part to see all the characteristics related to the part. The data is displayed in a table with information, such as the name, type, unit of measure, and additional details.

### Prerequisites

The following prerequisites must be met for characteristics data to display in Teamcenter:

- The characteristics data is **defined for the CAD design as Product and Manufacturing Information (PMI)**.
- The Dispatcher service is configured by your administrator to add characteristics data to Teamcenter.

### Procedure

1. Search for a part that contains characteristics data and open it to view its properties.
2. Click **Characteristics**.

A list of all characteristics for the part is displayed.

Tag	Name	Type	Information	Unit
PC10001	PC10001	Angular Dimension	Angular Dimension 90 +.02 -.003	°
PC10002	PC10002	Radial Dimension	Radial Dimension 9.5 +.02 -.01	mm
PC10003	PC10003	Arc Length Dimension	Arc Length Dimension 279.3 +.01 -.01	mm
PC10004	PC10004	Angular Dimension	Angular Dimension 124.2 +.01 -.01	°
PC10005	PC10005	Linear Dimension	Linear Dimension 76200 +0.3 -.02	µm
PC10006	PC10006	Linear Dimension	Linear Dimension 50.7 +1.5 -1.25	mm
PC10007	PC10007	Radial Dimension	Radial Dimension 9.5 +1.5 -1.25	mm
PC10008	PC10008	Linear Dimension	Linear Dimension 12.7 +.5 -.25	mm
PC10009	PC10009	Linear Dimension	Linear Dimension 0.0762 +.0015 -.0025	m
PC10010	PC10010	Linear Dimension	Linear Dimension 76.2 +.002 -.003	mm
PC10011	PC10011	Linear Dimension	Linear Dimension 60.3 +.25 -.125	mm
PC10012	PC10012	Linear Dimension	Linear Dimension 3 +.5 -.25	in


3. Select a characteristic from the list, and then click **Open** to view additional details about the characteristic.

The screenshot shows the Teamcenter interface for a characteristic named PC10002/A-PC10002. The interface is divided into a navigation sidebar on the left and a main content area. The main content area has two tabs: "Overview" (selected) and "Where Used". Below the tabs, there is a header bar with the following information: "Owning User: ed (ed)", "Date Modified: 05-Nov", "Type: Radial Dimension", and a search box containing "PC10002/A-PC10002".

The "Properties" section is expanded, showing the following details:

Name:	PC10002
Tag:	PC10002
Type:	Radial Dimension
Information:	Radial Dimension 9.5 +.02 -.01
Nominal:	9.5
Upper:	0.02
Lower:	-0.01
Before Appended Text:	
After Appended Text:	
Above Appended Text:	
Below Appended Text:	
Reference:	False
Inspection:	False
Criticality:	
Unit:	mm
UUID:	<jz003302>-e5b6e1a7-c1c2-45ea-b146-1296ec0a3ebc.a
Parent UUID:	<jz003302>-RMAIL_PML_Part_Inspector_50_v4.prt R0002c3ae00000024
Owning User:	ed (ed)
Creation Date:	05-Nov
Date Modified:	05-Nov
Last Modifying User:	ed (ed)

4. Do any of the following as necessary:

- Click **Overview** to view the detailed properties, such as additional information about the characteristic, criticality, owning user, unit of measure, and more.
- Click **Where Used** to view the **Source Items** and **Consuming Items** for the characteristic.
  - **Source Items** displays the parts associated with the characteristic. You can select a part from the list and then click **Open**  to view the properties for the selected part.
  - **Consuming Items** displays the objects that are using the characteristic data, such as a Teamcenter quality inspection definition.

**Note:**

The characteristics data displayed in Teamcenter is read-only and cannot be changed. If the characteristics require changes, they must be updated in the PMI for the associated CAD design and then saved to Teamcenter.

# 8. Exploring data with customized queries using recipes

## About recipes and samples provided

Recipes are custom queries created and used by technical team members such as feature owners, design and release engineers, and system engineers. Recipes let you create queries using a form-based interface to automate data collection for specific engineering activities across domains. You can recall saved recipes for reuse in feature areas to quickly and dynamically recall specific data views.

Active Workspace provides the following sample recipes that you can search for and execute:

- MBSE\_Capital\_Recipe

Recipe that fetches project model elements from the selected seed object, which is the basis for your query.

- MBSE\_Fetch\_RFL\_From\_Tracelinks


Recipe that returns requirements to functions and to logical objects that are tracelinked from the selected seed object.

## Create and execute a recipe

You can build a custom query using recipes to view data and recall the recipe for use later.

1. Navigate to the folder where you want to create the recipe.
2. Click **More Commands** **...** > **New** **✱** > **Add**.
3. In the **Type** field, enter the search term **recipe**, and then click **Recipe**.
4. Complete the **Add** dialog box and then click **Add**.

Active Workspace creates the recipe.

5. If the recipe is not open by default, navigate to the recipe and click **Open** .

The recipe appears in **Overview**.

Note:

To edit the **Overview**, click **Edit > Start Edit**.

6. In the **Seeds** panel, click **Add** ⊕.
7. Complete the **Add** panel to select one or more seed objects, and then click **Add**.
8. Complete the **Query** panel:
  - a. In the **Revision** section, click **Add** ⊕.
  - b. Complete the **Add** dialog box, and then click **Add**.
  - c. Select a **Closure Rule** from the drop-down list menu.
9. Click **Show Results**.

The query results appear in a table .

## Create a recipe in context

You can build a custom query using recipes to view data and recall the recipe for use later. You can create the recipe based on a selected object, such as a system model block or requirement.

1. Select an object in the **Content** panel.
2. Click **New** ✨ > **Create Recipe**.
3. Complete the **Create Recipe** dialog box, select the **Create and Open** check box, and then click **Create**.

Active Workspace creates the recipe, using the selected object in the **Seeds** panel.

4. (Optional) In the **Seeds** panel, click **Add** ⊕ to add more seed objects.
5. Complete the **Query** panel:
  - a. In the **Query** section, click **Add** ⊕.
  - b. Complete any editable fields in the **Add** dialog box, and then click **Add**.

The fields available depend on whether you are creating the recipe from a saved working context or you have just selected a context.



- c. Select a **Closure Rule** from the drop-down menu.
6. Click **Show Results**.

The query results appear in a table.

## Load, edit, save, or copy a recipe


### Load and edit recipes

You can open an existing recipe and then modify the recipe values.

1. Perform a search or locate a recipe in the folder in which it was created.
2. Select the recipe and then click **Open** .
3. Perform any of the following:
  - To edit, the **Overview** panel **Name** or **Description**, click **Edit**  > **Start Edit**, modify the content, and then save the edits.
  - Edit the **Seeds** and **Query** panels, and then save the recipe.

### Save or copy a recipe


You can save a recipe or create a duplicate of an existing recipe with a different name.

1. Create or load a recipe.
2. Click **New**  > **RecipeSaveAs**.

Save the recipe under an existing name or a new name.

## Execute a recipe in context

You can run a saved recipe on a selected object in a structure.

1. Locate and select one or more objects in the Results panel or in the **Relations** tab diagram.
2. Click **Manage**  > **Execute Recipe**.
3. (Optional) Enter a filter term.
4. Select a recipe, and then click **Execute**.

The recipe runs on the selected object and the recipe page appears, showing the query results in a table.

The screenshot shows the 'SEEDS' configuration panel on the left and the 'RESULTS' table below it. The 'SEEDS' panel includes a table with columns for Name, Category, Revision, Include, and Owner. The 'Include' column has checkboxes for each row. Below the table is an 'In BOM' checkbox and a 'Show Results' button. The 'RESULTS' table has columns for Object String, Revision, Type, Is Occurrence..., Release Sta..., Owner, Name, ID, and Occurrence....

Name	Category	Revision	Include	Owner
VEHICLE	Top BOM Node	A	<input checked="" type="checkbox"/>	
MIRROR	Seed	A	<input checked="" type="checkbox"/>	
SEATING SYSTEM	Seed	A	<input checked="" type="checkbox"/>	
Braking	Seed	A	<input checked="" type="checkbox"/>	

Object String	Revision	Type	Is Occurrence...	Release Sta...	Owner	Name	ID	Occurrence...
ELECTRONIC MIRRORS	A	System Block Revision	True			ELECTRONIC MIRRORS	027091	
MIRROR	A	System Block Revision	True			MIRROR	027086	
TRIM	A	System Block Revision	True			TRIM	027085	
SEATING SYSTEM	A	System Block Revision	True			SEATING SYSTEM	027094	

5. (Optional) Do any of the following and click **Show Results**:

- Toggle the **Include** check box to include or exclude object seeds.
- Select the **In BOM** or **In Context** check box to include only objects within the BOM or select context, which excludes downstream related objects such as requirements, parameters, or trace links.
- Click **Table** to switch to **Tree** view.

In the **Tree** view, if the closure rule traversal paths are different for the same object, you see the object twice—each with a different parent. For example, if we reach to object *D* from *A* directly and from *C* through *B*, then the **Tree** view shows the following:

*A*

—*D*

*B*

—*C*

—*D*

- **Update the variant.**

## Change the configuration

You can change the revision rule, unit, date effectivity, and variant rule, which includes Product Configurator variants. If you do not apply the Product Configurator context, then no variant rule is applied.

1. **Execute a recipe.**
2. Click **Edit**  > **Start Edit** .
3. In the **INCLUSION RULE** panel, click **Configuration** .

A message warns you that a new configuration may update the seed.

4. Click **Add**.
5. In the **Add** dialog box, select a **Variant** and click **Add**.
6. Click **Show Results**.

**Note:**

If variant conditions are not defined on one of the **SEEDS** object child children, then the seed object is displayed in the recipe regardless of the variant that is applied.

7. (Optional) Click **Save Recipe** to save **INCLUSION RULE** panel changes.