



TEAMCENTER

Enterprise Model Management

Teamcenter 2412

Unpublished work. © 2025 Siemens

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

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

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

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

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

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

About Siemens Digital Industries Software

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

Support Center: support.sw.siemens.com

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

Contents

Overview of Enterprise Model Management 1-1

An example of the flow of data between Git or a file system and Teamcenter 2-1

Overview of using file-based integration 3-1

Working with file system data

Import file system data into Teamcenter _____ 4-1

Update file system data in Teamcenter _____ 4-2

Download file system data from Teamcenter _____ 4-4

Working with Git data

Import Git data into Teamcenter _____ 5-1

Update Git data in Teamcenter _____ 5-5

Download Git data from Teamcenter _____ 5-9

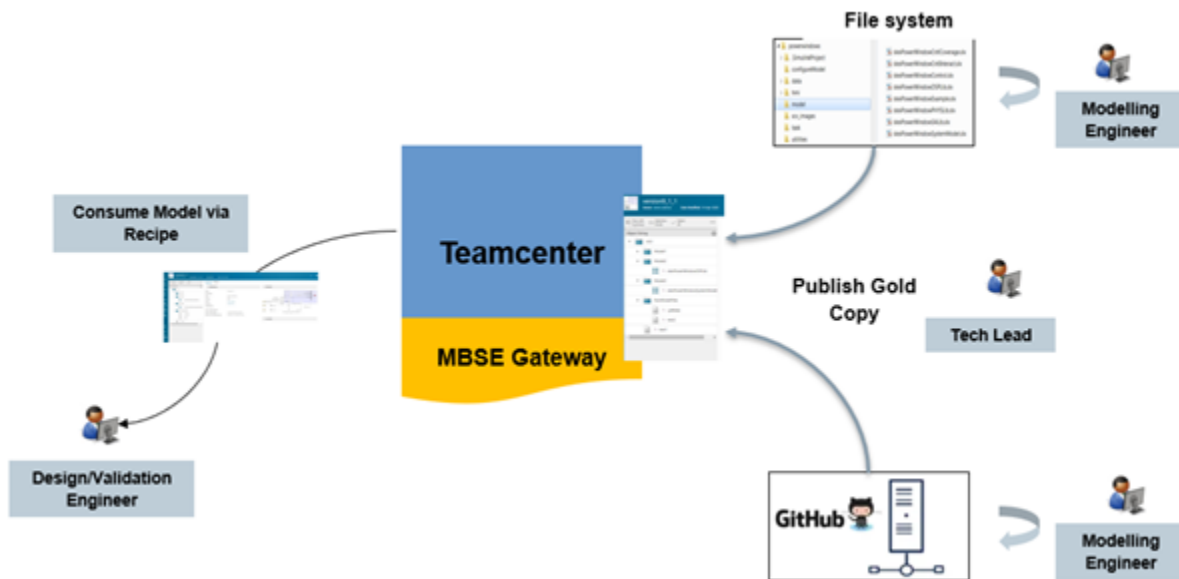
Release a model collection 6-1

Compare the baselines or BVR structures of two model collection objects 7-1



1. Overview of Enterprise Model Management

The MBSE Integration Gateway integration allows modeling engineers and team leads to save the released models from Git and their file systems to Teamcenter. The daily modeling work is performed and managed in Git or in the file system. When the model is ready for *release* and for sharing with the enterprise, the data is imported into Teamcenter. The data that is imported is the released data (tag) or the gold copy of the data.



The models in Git or in the file system that are saved to Teamcenter have access to standard Teamcenter services and can be related to the system-of-interest, such as system, function, and requirements elements, for cross-product traceability. The models in Git can also be used in MBSE processes such as verification and system modeling. The folder structure of the models that are imported is replicated in Teamcenter.

Using this, you can:

- Import Git data or data on the file system into Teamcenter.
Gold copies of the models are imported from Git or the file system.
- Update existing models to create a new baseline.
- Download models from Teamcenter.

Models from one or many systems can be downloaded for co-simulation.



Note that:

- All the models saved in Teamcenter are placed in the **Model Collection** object.
- References between models and model libraries depends on the connector configuration. For example, in the Simulink connector, references are preserved, while in the common connector, the references are not preserved.

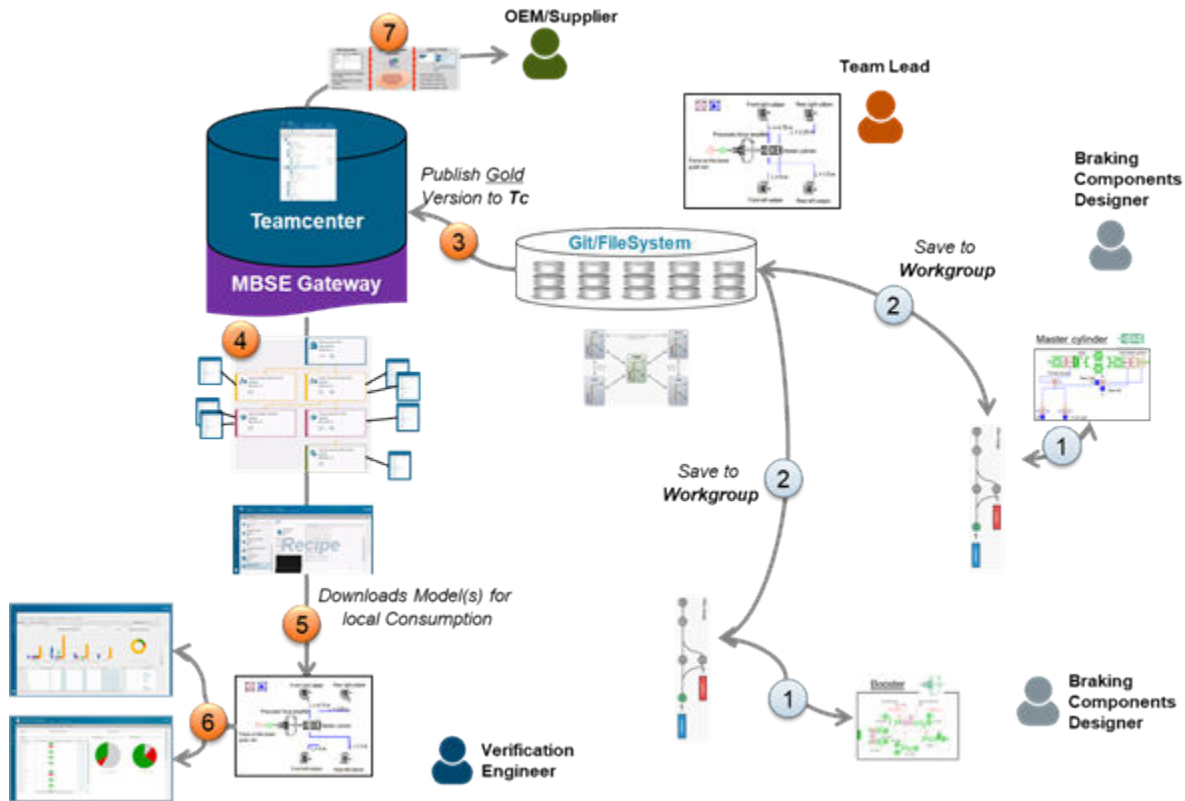
The following SCM applications are supported:



Where do I go from here?

 Installer	See Checklist for deploying Teamcenter MBSE Integration Gateway.
 Business User	
Understand the flow of data	See an example of the flow of data between Git or a file system and Teamcenter.
Work with file system data	Follow the steps outlined in the sections on how to import and update file system data in Teamcenter and download this data from Teamcenter.
Work with Git data	You can import and update Git Data in Teamcenter and download this data from Teamcenter.

2. An example of the flow of data between Git or a file system and Teamcenter



This topic illustrates the flow of data between Git or a file system and Teamcenter.

1. Modeling engineers author models in Git or a file system from their local repositories.
2. Engineers regularly push their work to the main repository in case of Git.
3. When the model is ready for release and the team is ready to share the model with the enterprise, the team lead or the modeling engineer publishes or imports a *gold* version of the model to Teamcenter.
4. Models can be related to a system-of-interest such as system, function, and requirements elements for cross-product traceability.
5. A verification engineer downloads the model for local consumption (for example, performing a system simulation).

3. Overview of using file-based integration

Most behavior modeling tools are file-based and client-only tools. Such tools are used from desktops. In the Embedded Software domain, many modeling tools are used to define and simulate a system's behavior.

Engineers from a project team create a simulation project that is a collection of artifacts defining a system or a subsystem. Data in the form of models, model-dependent data, model-derived data, and other files are organized in a specific manner for each system. Engineers add, update, or remove the contents of such projects when designing or testing the system features. Such simulation projects are imported to Teamcenter, and a *model collection* object is created to represent this project.

Administrators use the import feature to import the project data and associate it to a model collection. Engineers can continue working in their tools. Whenever a project reaches a milestone, the project can be updated in Teamcenter. From Teamcenter, the data can be downloaded and used in downstream processes such as verification and system simulation.

4. Working with file system data

Import file system data into Teamcenter

1. In Active Workspace, choose **More Commands...** > **Import/Export**  > **Import Model Collection**.

A MIME-type file is downloaded in your browser. Depending on how your browser is configured, this file may automatically launch the **Model Management** client.

Import Model Collection Close

Folder Path: *

Choose Folder

Model Collection: *

Revision: *

Property File:

Choose File

Import

2. In the **Import Model Collection** panel, provide the information for import as follows:

Name	Action to be performed
Folder Path	Paste the location of the model folder in the file system.
Model Collection	Specify the name of the model collection if you do not want to use the default name.
Revision	Specify the revision.
Property File	Specify the path to the properties file. This file contains the definition for the additional properties that you want to import.

3. Click **Import**.

The **Model Management** client displays the status of the import. You can navigate to the imported content by clicking the link in the client or you can close the client and refresh your browser.

The files are imported to your **Home** folder and organized under a **Model Collection** type object.

The screenshot displays the Teamcenter Model Management client interface. The top navigation bar shows the path: powerwindows > powerwindows > model > slexPowerWindowControl. Below the navigation bar, there is a toolbar with icons for 'Tree with Summary', 'Add', 'Find', 'Filter', 'Context', 'Markup', and 'Edit'. The main area is divided into two panes. The left pane shows a tree view of model collections with columns for 'Element Name', 'ID', 'Revision', and 'R'. The right pane shows the 'PROPERTIES' for the selected element, including fields for ID, Revision, Revision Name, Description, Occurrence Name, Reference Designator, Sequence, Quantity, Unit Of Measure, Release Status, Date Released, Release Effectivity, Element Effectivity ID, and Element Effectivities.

Element Name	ID	Revision	R
powerwindows	027166	A	powe
powerwindows	027192	A	powe
model	027212	A	mode
hmi	027195	A	hmi
SimulinkProject	027193	A	Simu
configureModel	027204	A	config
scv_images	027179	A	scv_in
utilities	027220	A	utilite
data	027191	A	data
task	027207	A	task
work	027184	A	work

PROPERTIES	
ID:	027161
Revision:	A
Revision Name:	slexPowerWindowControl
Description:	
Occurrence Name:	
Reference Designator:	
Sequence:	50
Quantity:	
Unit Of Measure:	each
Release Status:	
Date Released:	
Release Effectivity:	
Element Effectivity ID:	
Element Effectivities:	

Update file system data in Teamcenter

1. In Active Workspace, open a model collection that you want to update, and choose **More Commands...** > **Manage** > **Update Model Collection**.

A MIME-type file is downloaded in your browser. Depending on how your browser is configured, this file may automatically launch the **Model Management** client.

Update Model Collection
Close

Folder Path: *

C:\bhm\staging\powerwindows

Revision: *

B

Update

- In the **Update Model Collection** panel, provide information for the import as follows:

Name	Action to be performed
Folder Path	The location of the model folder is automatically selected.
Revision	Specify a name for the revision.

- Click **Update**.

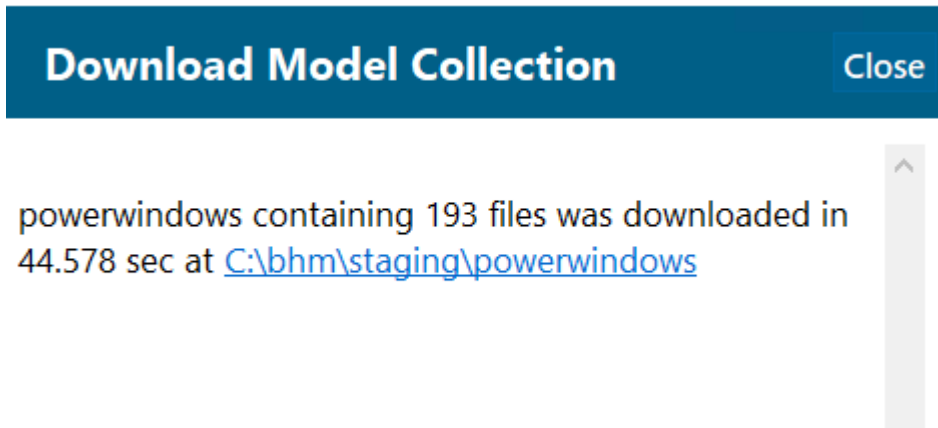
The **Model Management** client displays the status of the update. You can navigate to the updated content by clicking the link in the client or you can close the client and refresh your browser.

After updating, you can compare the metadata of the two different baselines by performing a **model collection compare**.

Download file system data from Teamcenter

1. In Active Workspace, open a model collection that you want to download and choose **More Commands...** > **Share**  > **Download Model Collection**.

A MIME-type file is downloaded in your browser. Depending on how your browser is configured, this file may automatically launch the download using the **Model Management** client.



The model collection is downloaded in the staging directory.

5. Working with Git data

Import Git data into Teamcenter

1. In Active Workspace, choose **More Commands...** > **Import/Export**  > **Import Model Collection**.

A MIME-type file is downloaded in your browser. Depending on how your browser is configured, this file may automatically launch the **Model Management** client.

Import Model Collection
Close

Source Link : *

Model Collection: *

Revision: *

LOGIN

Login: *

User: *

Password: *

Save Credentials

Import

2. In the **Import Model Collection** panel, provide the information for import as follows:

Name	Action to be performed
Source Link	Drop the Git tag or paste the URL of the Git

Name	Action to be performed
------	------------------------

	tag. The Git tag can be an annotated tag or a lightweight tag.
--	--

Model Collection	Specify the name of the model collection.
-------------------------	---

Revision	Specify the name of the revision.
-----------------	-----------------------------------

Provide the login details for accessing the Git repository as follows:

Login	Select from the following login options:
--------------	--

- **User/Password:** To use the username and password of your Git account.

Login: *

User/Password

User: *

Required

Password: *

|

Type the username and password of your Git account.

- **Personal Access Token:** To use the personal access token generated using the Git tool.

Name	Action to be performed
------	------------------------

Login: *

Personal Access Token

Personal Access Token: *

Required

Type the Personal access token.

- **SSH:** To use SSH authentication

Login: *

SSH

SSH Key Path: *

Choose File

Required

Passphrase:

Choose the SSH Key Path file and provide the passphrase.

Save Credentials Select the check box to save the login credentials to enable autologin.

3. Click **Import**.

The **Model Management** client displays the status of the import. You can navigate to the imported content by clicking the link in the client or you can close the client and refresh your browser.

The files are imported to your **Home** folder and organized under a **Model Collection** type object.

The screenshot displays the Teamcenter Model Management interface. The top navigation bar includes the 'modelmanagement' logo and various filters like 'Revision: Global (Latest Working)', 'Date: Today', 'Units: None', 'Variant: No Variant Rule', and 'Expansion: No Rule'. Below the navigation bar is a toolbar with icons for 'Tree with Summary', 'Add', 'Find', 'Filter', 'Context', 'Markup', and 'Edit'. The main area is a table listing model elements with columns for 'Element Name', 'ID', 'Revision', and 'Revision Name'. The table contains several rows, including 'modelmanagement', 'powerwindow', 'model', 'task', 'slexPowerWindowExample', 'utilities', 'testUtilities', 'results', 'hmi', 'configureModel', and 'data'. To the right of the table is a '3D Overview' panel with tabs for 'Overview', 'Parameters', and 'Changes'. The 'Overview' tab is active, showing a 'PROPERTIES' section with fields like 'ID', 'Revision', 'Revision Name', 'Description', 'Occurrence Name', 'Reference Designator', 'Sequence', 'Quantity', 'Unit Of Measure', 'Release Status', 'Date Released', 'Release Effectivity', 'Element Effectivity ID', 'Element Effectivities', 'Owner', and 'Group ID'.

Element Name	ID	Revision	Revision Name
modelmanagement	027239	A	modelmanageme
modelmanagement	027240	A	modelmanageme
powerwindow	027249	A	powerwindow
model	027246	A	model
task	027248	A	task
slexPowerWindowExample	027236	A	slexPowerWindow
utilities	027243	A	utilities
testUtilities	027247	A	testUtilities
results	027244	A	results
hmi	027245	A	hmi
configureModel	027241	A	configureModel
data	027250	A	data

After importing, we recommend that you classify the imported model collection and **then release it using the Model Collection Release Workflow** so that the imported model becomes a gold copy.

Update Git data in Teamcenter

1. In Active Workspace, open a model collection that you want to update, and choose **More Commands...** > **Manage** > **Update Model Collection**.

A MIME-type file is downloaded in your browser. Depending on how your browser is configured, this file may automatically launch the **Model Management** client.

Update Model Collection
Close

Source Link : *

Revision: *

LOGIN

Login: *

User: *

Password: *

Save Credentials

Update

2. In the **Update Model Collection** panel, provide information for the import as follows:

Name	Action to be performed
Source Link	Drop the Git tag or paste the URL of the Git tag. The Git tag can be

Name	Action to be performed
------	------------------------

Revision	<p data-bbox="469 245 748 312">an annotated tag or a lightweight tag.</p> <p data-bbox="469 338 777 403">Specify the name of the revision.</p>
-----------------	--

If you have not saved the login credentials, provide the login details for accessing the Git repository as follows:

Login Select from the following login options:

- **User/Password:** To use the username and password of your Git account.

Login: *

User: *

Password: *

Type the username and password of your Git account.

- **Personal Access Token:** To use the personal access token generated using the Git tool.

Name	Action to be performed
------	------------------------

Login: *

Personal Access Token

Personal Access Token: *

Required

Type the Personal access token.

- **SSH:** To use SSH authentication

Login: *

SSH

SSH Key Path: *

Choose File

Required

Passphrase:

Choose the SSH Key Path file and provide the passphrase.

Save Credentials Select the check box to save the login credentials to enable autologin.

3. Click **Update**.

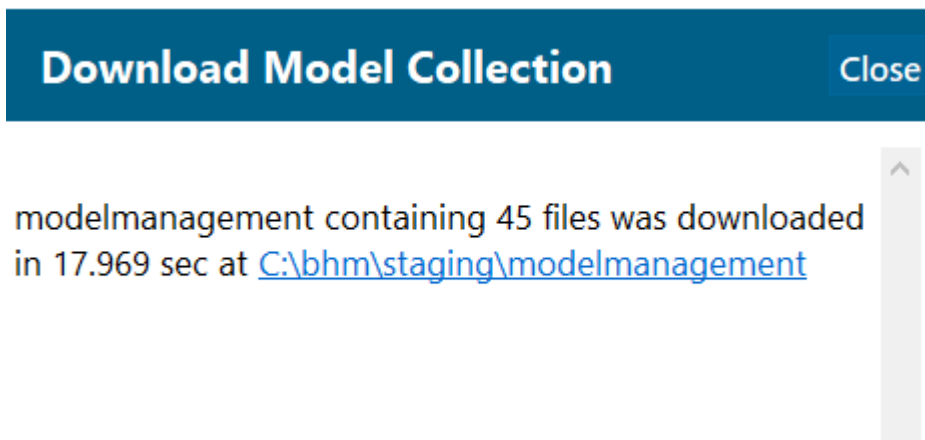
The **Model Management** client displays the status of the update. You can navigate to the updated content by clicking the link in the client or you can close the client and refresh your browser.

After updating, you can compare the metadata of the two different baselines by performing a **model collection compare**.

Download Git data from Teamcenter

1. In Active Workspace, open a model collection that you want to download and choose **More Commands...** > **Share** > **Download Model Collection**.


A MIME-type file is downloaded in your browser. Depending on how your browser is configured, this file may automatically launch the download using the **Model Management** client.



The model collection is downloaded in the staging directory.

6. Release a model collection

After you import a model collection, we recommend that you release it so that the model collection becomes a gold copy.


1. In Active Workspace, open a model collection that you want to update, and choose **More Commands...** > **Manage**  > **Submit to Workflow**.
2. In the **Submit to Workflow** dialog box, select **Model Collection Release Workflow** as the workflow template and click **Submit**.

7. Compare the baselines or BVR structures of two model collection objects

You can compare the metadata of two **Model Collection** objects using the following as follows, depending on whether you are using the BVR mode or the WSO mode:

- **Compare the BVR structures of two model collection objects**
- **Compare the baselines of two model collection objects**

Compare the BVR structures of two model collection objects

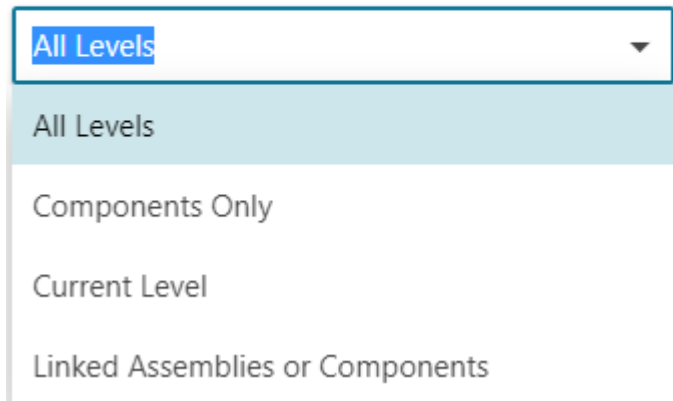
1. Select two model collection structures or two revisions of a model collection structure from either of the following locations:
 - Search results
 - The **Content** tab
 - The **History** tab
 - A saved working context
 - Favorites
2. Click the **Open**  button and then select **Open**.

The selected model collection structures are displayed in separate views.

3. Choose **More Commands...** > **View**  > **Compare Structures**.

The **Compare** panel appears.

4. Specify the comparison level by choosing **Current Level**, **All Levels**, **Components Only**, or **Linked Assemblies or Components** from the list.



5. Specify the information you want to view in the comparison results by selecting from the following check boxes:
- **Matched**
 - **Different**
 - **Unique in Left View**
 - **Unique in Right View**

Compare

✕ Close

All Levels

▼ OPTIONS

▼ DISPLAY

- Matched
- Different
- Unique in Left View
- Unique in Right View

▼ EQUIVALENCE

- Dynamic Equivalence

6. To enable dynamic equivalence when running an accountability check, select the **Dynamic Equivalence** check box.

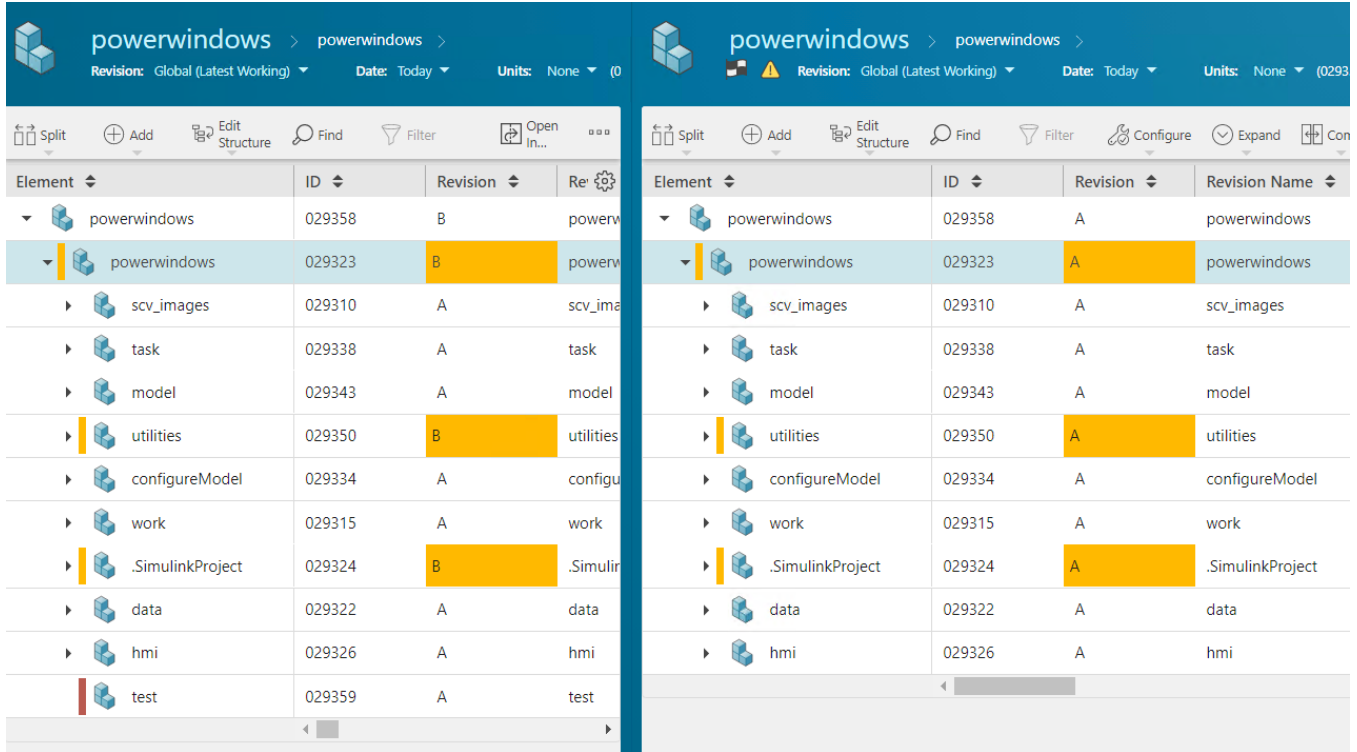
In some comparisons, matching occurrences in two structures have slightly different properties and are therefore not reported as *equivalent* when you run an accountability check. To report such occurrences, run a dynamic equivalence check.

Your administrator must specify the list of internal BOM line properties to be used as dynamic equivalence criteria, using the **MEAccountabilityCheckDynamicIDICProperties** preference. If no values are specified, **item ID** is used as the default attribute for comparison.

7. (Optional) Select the **Run in background** check box. This is recommended when an **All levels** or a **Components only** comparison is performed on large structures.

For the compare operation to run in the background, your system administrator must configure Dispatcher services.

8. Click **Compare**.



Compare the baselines of two model collection objects



1. Choose **General** from the **Advanced Search** panel and from the **Type** list, choose **Model Collection**. In the **Name** box, enter the name of the **Model Collection** object.



Click **Search**.










2. From the search results, select two **Model Collection** objects, and choose **More Commands...** > **View** > **Compare Model Collection**.

The **Branch Content Comparison** page shows the results of the comparison of the two **Model Collection** objects. The colored bars against the names indicate the comparison results, and they must be interpreted as follows:

- Yellow: The contents of the **Model Collection** objects are different.
- Blue: The contents of the **Model Collection** object on the right are different.
- Red: The contents of the **Model Collection** object on the left are different.

 **K1**
 Owner: 
 Date Modified: 25-Nov-2019 15:52

 **K2**
 Owner: 
 Date Modified: 25-Nov-2019 15:54

Name	Result	Left Object	Right Object	Left State
 testModelCollection	Different	testModelCollection	testModelCollection	
 Sample	Left Only	1 - Sample		Published
 Sample	Right Only		2 - Sample	
 SingleModel	Right Only		1 - SingleModel	
 SingleModel	Left Only	1 - SingleModel		Published
 test1.txt	Right Only		1 - test1.txt	
 test1.txt	Left Only	1 - test1.txt		Published
 test2.txt	Right Only		2 - test2.txt	
 test2.txt	Left Only	1 - test2.txt		Published