

Moving elements to other projects

On this page

- [Moving elements to/from used projects](#)
- [Fixing element references in other projects](#)

You can move selected elements from the main server project to a [directly used project](#). The relations are also moved with the element(s).



Moving cannot be undone

This action cannot be undone as a new version is automatically created and committed to the Teamwork Cloud server after the selected elements are moved to a new location.

Moving elements to/from used projects

To move elements from the main server project to a used one

1. In the Model Browser, select one or more elements.
2. Right-click the selected elements. From the shortcut menu, choose **Refactor > Move Element(s) to Other Project**.
3. The **Move Elements** dialog opens. Select a package or a classifier in the used project where you want to move the elements. You will see the classifier as a possible destination **if** only classifier elements are selected to move. Otherwise, only packages will be shown.



Drag the selection

You can drag and drop the selected elements to the used project package or classifier instead of using the shortcut menu. This action can be performed if the following conditions are met:

4. Do one of the following:
 - Click **Move to Selected** if the selected elements contain at least one non-classifier element, it is restricted to drop to other targets than packages.
 - click **Select More** to specify your selection. The **Select Elements** dialog elements opens, in which you can:
 - **add** or remove some elements from a selection or
 - if the selected elements to move contain only classifiers, moving is allowed to either packages or classifiers. Change the destination (click  at the end of the row). The **Select destination** dialog will list destination targets by the type of the element selected to move (i.e. if it is a classifier element, classifiers will be listed in the Tree and List tabs). Otherwise only packages will be listed.
 - Click **OK** when you are done.
5. A notification appears, asking if you want to [check dependencies among projects](#). Select **Yes/No** to continue moving elements.
6. After you move the elements, new main project and used projects versions are created and automatically committed to the server. Additionally, the user project is switched to the latest version.

To move elements from a used project to the main one

1. Open the used project and create a new package.
2. In the Model Browser, select the elements you want to move and drag them to the newly created package.
3. [Export the package](#) to a new temporary project.



If the elements you want to export have dependencies in the project you want to export them from, a dependency warning will be prompted in the **Package Dependencies** panel.

4. [Remove the used temporary project](#), which is displayed under the package *Project Usages* in the Model Browser. Commit the changes.
5. Open the main project and [update the used project](#) containing the elements to its latest version (or remove it keeping references), in which the used project no longer contains the elements that were moved.
6. [Use the new temporary project](#) in the main project.
7. [Import the content of the used temporary project](#) to the project. Once you click the  button, the **Import** dialog opens.
8. In the **Import** dialog, you can select to import the elements either with the new element IDs or with the old ones.
 - a. Selecting **Import with New IDs** creates new IDs for the imported elements in the main project. This way, element IDs are not duplicated, ensuring that the main project remains intact. However, if the elements of the used project are used in other projects, choosing **Import with New IDs** may break those references.



It is highly recommended to import elements with new IDs as it is the safest option regarding project consistency.

- b. Selecting **Import with Old IDs** imports the elements with their existing IDs, which can be useful if the elements are used in multiple other projects. However, this may cause duplicate element IDs, resulting in the main project being corrupted. To proceed with the import using the old IDs, click **Yes** in the **Question** dialog, acknowledging that the procedure might corrupt the data of the main project.
- c. **Question** dialog opens, click **Yes** to save and commit the changes.



It is recommended to delete the temporary project from the server repository once its data is imported to the main project, especially if the data is imported with the old IDs, to make sure there are no duplicate elements with the same element IDs.

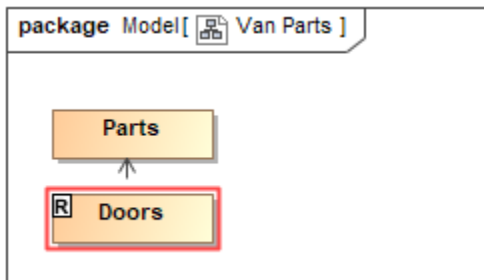
Fixing element references in other projects

If the elements that were moved from a used project to a new one were used in other projects prior to the move, validation issues may occur in those projects once the elements are moved from their original project to a new one. The procedure for resolving such issues depends on whether the elements were imported with the old element IDs or the new ones during the element move procedure from a used project to a new one.

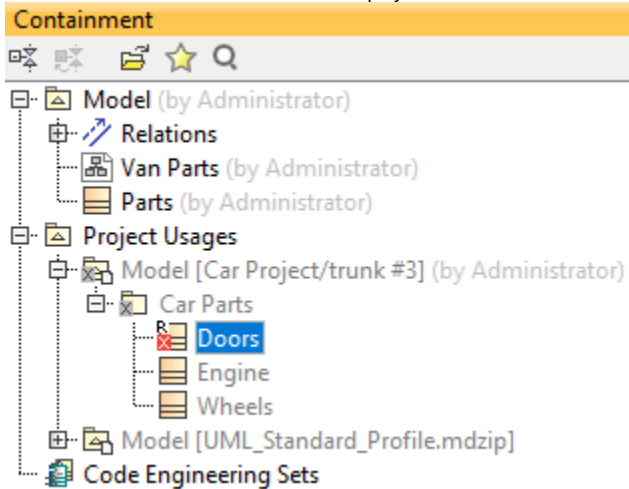
To fix references in other projects after the import

1. Validation issues will appear in the main project as the elements that were used from the used project no longer exist in the newest version of the used project. These elements will be displayed:

- In a diagram - the element symbol is displayed with a red border and a letter *R* on its corner.



- In the Model Browser - the element is displayed with a red X mark and a letter *R*.



- In the **Active Validation Results** panel on the bottom right of the modeling tool.

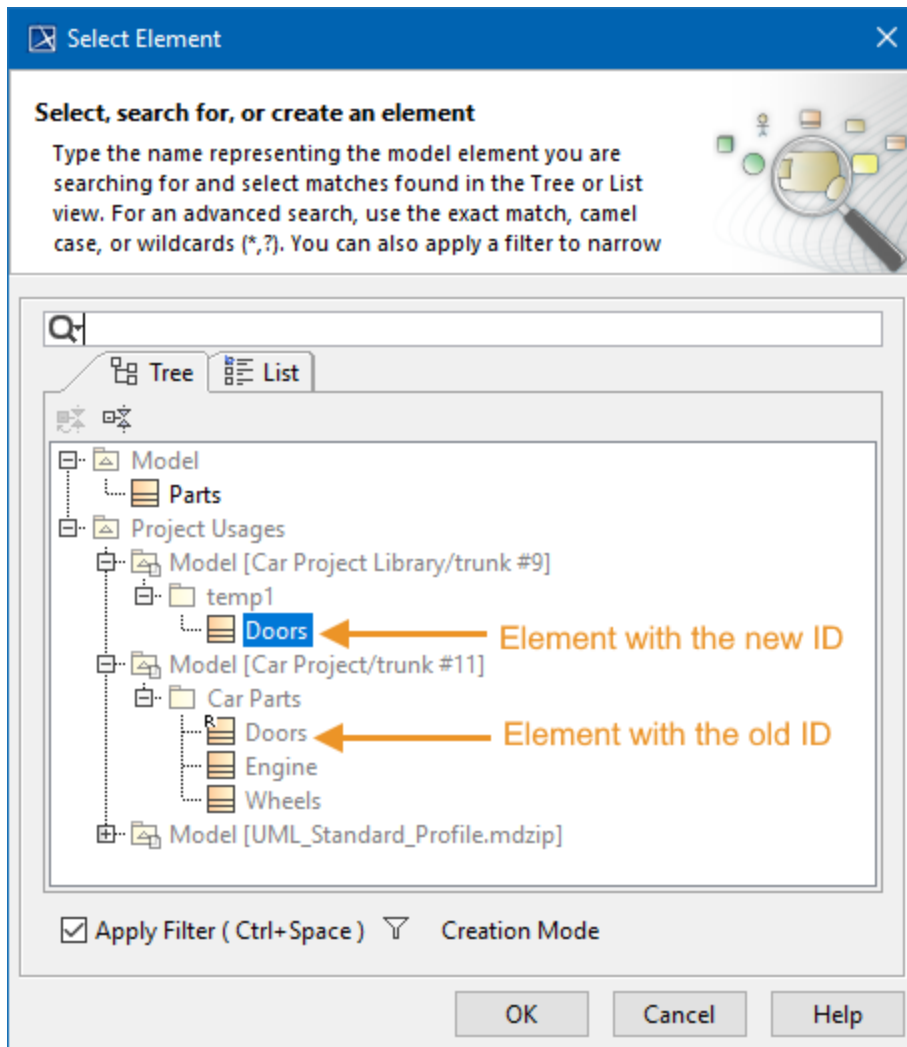


2. In the other project using the elements that are no longer contained in the used project, [use the new project](#) to which the elements were moved.



If the elements were imported with the **old IDs**, Step 4 resolves the validation issues, as the newly used project contains the moved elements with the same old IDs that were initially used in this project.

3. If the elements were imported with the **new IDs**, validation issues will still remain after Step 4 as the element in the newly used project has a new ID, while the opened project used the element with the old ID. To resolve the issue:
 - a. Double click on the element or its symbol, select **Validation > Change Usages To. Select Element** dialog opens.
 - b. In the model tree, you can see both the previously used element with the old ID, displayed with a letter *R* on it and the newly used element with the new ID. Double-click on the newly used element to replace the used element with the old ID with the one with the new ID.



Example

Say you have three projects: Car Project, Library and Van. Car Project has element Doors. Project Van uses the Car Project and its element Doors in a diagram.

Related pages

- You identify the Package Dependencies
 - Start using a project in your project
- importing the data of the project *temp* to the Library (either with the old IDs or with the new ones).

You would then need to remove the used project *temp* from the Car Project and commit the changes. This removes the element Doors from the Car Project completely.

Now you would need to open the project Van, which uses the element Doors through the used project Car Project. The Car Project version would need to be updated to the latest one. This latest version no longer contains the element Doors, as it was moved to the Library. This will cause a validation issue in the project Van, as the element Doors is no longer to be found in the used project Car Project.

To resolve the validation issue, you would need to use the project Library, as it is the one containing the element Doors. If the data of the project *temp* was imported to the Library with the old IDs, the validation issue would be resolved. If the data was imported using the new IDs, the validation issue would remain, as the Van project would still have the instance of element Doors with the old ID. You would need to resolve the issue by selecting the element with the new ID (located in the newly used Library project) as described in Step 3.