

Resolving unconfirmed usages of packages

For the overview of automated project usages and situations when they can become unconfirmed, see [Controlling dependency creation between used projects](#).

Unconfirmed project usages are caught by special [automatic validation rules](#) and displayed as a warning or error validation result. The severity depends on the exact model composition situation, but the solution methods are the same in both cases.



Model Usages

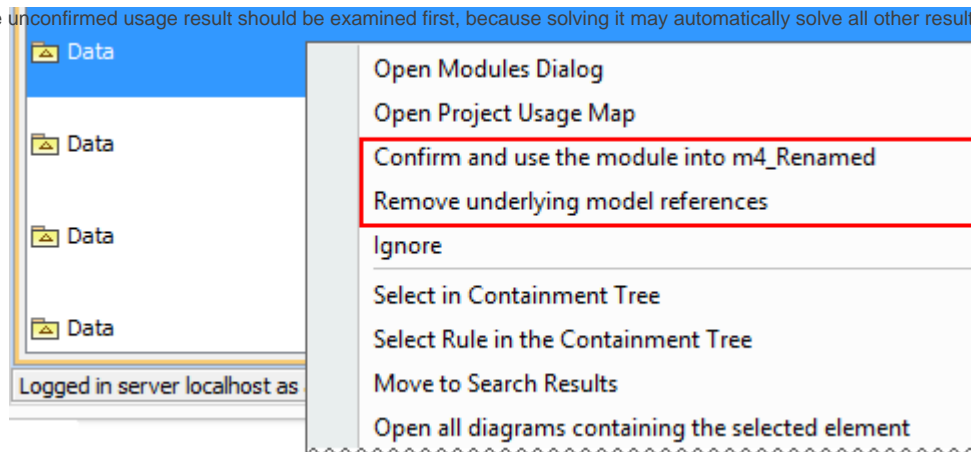
Since problematic project usages are not model elements, they cannot be visible in the [Containment tree](#). So when the model contains unconfirmed usages, the root package *Model* is highlighted as incorrect (regardless of which usage is actually problematic).



Model Usages

When there is an error level unconfirmed usage of some used project, then all the elements of the used project, that are referenced from other model places are shown as [Recovered Elements](#). These recovered elements are in turn flagged as errors. Hence single unconfirmed usage can cause a lot of errors. There are two ways to resolve an unconfirmed used project usage. The usage can be either confirmed or rejected.

In this situation, the unconfirmed usage result should be examined first, because solving it may automatically solve all other results.



Two ways for resolving unconfirmed used project usages.

Confirming usages

If the used project usage A B is good and necessary according to the user policy, it can be confirmed, i.e., the user-defined usage is to be created instead of the current unconfirmed automated used project usage.

You can confirm an automated usage

- By using the specific command
- Manually

To confirm an automated usage by using the specific command

1. Right-click the automated usage in the validation results.
2. Select **Confirm and use the used project into <used project_name>** from the shortcut menu (see the preceding figure). The [Used project wizard](#) opens. The required used project is already selected in the first step of the wizard.
3. Click **Finish**. The necessary user-defined used project usage will be created.

To confirm an automated usage manually



Example

In this case, you can better control the usage creation process. So if you need, for example, to use not only a required used project but some other one that brings the required used project as a part of it, be advised to confirm the usage manually.



Important!

However, this is the case when you must be sure that the user-defined usage will be created in the right place.

brings In the main menu, click **Options > Used Projects**. The [Used Projects dialog](#) opens.

2. Select a used project that is the usage target resource.
3. Click the **Used project** button. The [Used Project wizard](#) opens.
4. Follow the steps of the usual procedure (select a required used project and finish or continue to the second wizard step to provide more configuration options for used project usage and then finish).

Rejecting usages

If the usage A B is not good according to the user policy, it must be rejected and removed.

You can reject an automated usage

- By using the specific command (for removing underlying model level references)
- Manually (by removing or redirecting underlying model level references to different elements)

To reject an automated usage by using the specific command



Model Usages

Use this way to reject a model-level usage that, for example, was created inadvertently.

1. Right-click the automated usage in the validation results.
2. Select **Remove underlying model references** from the shortcut menu (see the preceding figure). All underlying model-level references (the ones that cause this automated usage) will be removed.



This command is very similar to the **Clear Recovered Element Usages** command.

To remove an automated usage manually

In more complex cases, you may want to address each usage individually. Use dependency analysis means to examine what dependencies are there and review them on a case-by-case basis (see [Analyzing usages and dependencies](#), [Analyzing package dependencies](#), [Displaying related elements](#), [Dependency matrix](#), and so forth).

Useful commands for preparing to resolve unconfirmed used project usages

When you need to explore your project composition in detail before attempting to resolve unconfirmed usages, use the following commands for navigating:

- Open [Used Projects dialog](#)
- Open [Resource Usage Map](#) for collaboration projects

Related pages

- [Analyzing Usages and Dependencies](#)
- [Analyzing Package Dependencies](#)
- [Using other Projects in a Project](#)
- [Validation](#)