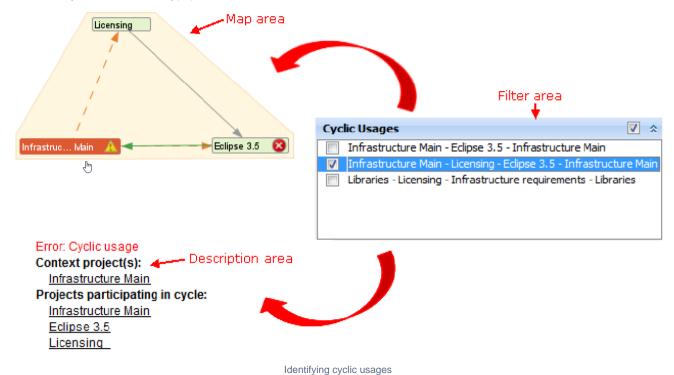
Solving Project Usage Problems

Using the Project Usage Map, you can identify usage problems. Then you can easily select and open projects that need to be fixed.

Resolving cyclic usages

Problem

When a project decomposition is used, the project is split into smaller projects. You can benefit from this decomposed project - you need to load only small part of the project instead of all of it. This reduces complexity and even improves performance. In addition to this, you would not expect and most likely not care to load the remaining parts of the main project each time you open a single part of it. This can happen if you have cyclic usages, i.e., project parts are using the main project. Typically, such usages increase complexity and reduce performance. Also, if the project takes part in a cycle, it cannot be reused as an independent part of another project. Other projects taking part in the cycle will be automatically used as well. Cycles are often a symptom of unintentional usages, created unknowingly by the user.



Solution

The Project Usage Map automatically identifies cycles and highlights them in the Repository View. You can then open a Project Usage Map for the suspected projects participating in a cycle to analyze it more closely and if necessary - break usages that cause cycles.

To resolve the cyclic usages

Sule to check and resolve the model level usages between the project and module.

- 1. Open the main project of the cycle.
- 2. From the Options menu, select Modules. The Modules dialog opens.
- 3. Select a module which should be removed, and click the Lock button.
- 4. Click the RemoveModule button.
- 5. Click **OK** when you are done.

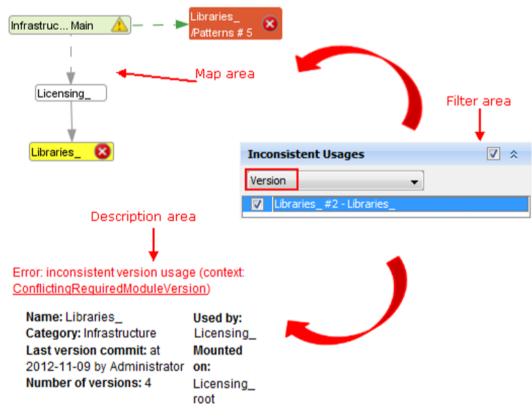
Modules	×				
Specify module usage options Modules allow the separation of projects into parts to allow the reuse of project parts in other projects. Select a used module to change its usage options (accessibility, load mode, index usage, and shared package mounts). To use or					
Project Eclipse 3.5 UML_Standard_Profile.mdzip Infrastructure Main (by Administrator)	Infrastructure Main usage in Project Eclipse 3.5 Module Version Current: Trunk, 5 [Latest] Required: Trunk, 5 [Latest] Module Accessibility Module Load Mode Image: Always load Image: Always load <td< th=""></td<>				
Unlock Use Module Remove Module Im	port Reload Unload Open as Project Options OK Cancel Help				

Removing module to break cyclic usage

Resolving version inconsistencies

Problem

You are using different versions of the same project in your main project.



Identifying version inconsistency

Solution

The Project Usage Map highlights these inconsistencies. You can then open projects with inconsistent usages and fix them by unifying the used project version.

To resolve a version inconsistency

Sure to check and resolve the model level usages between the project and module.

- 1. Open a project using the projects of inconsistent version.
- 2. From the Options menu, select Modules. The Modules dialog opens.
- 3. Select a module which version you want to change, and click the Lock button.
- 4. In the ModuleVersion area, click the ... button. The EditBranches dialog opens.
- 5. Select the wanted version.
- 6. Click **OK** when you are done.

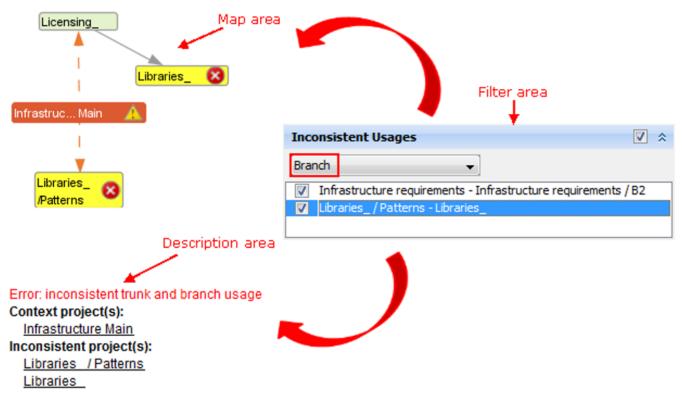
Modules						
Specify module usage options Modules allow the separation of projects into parts to allow the reuse of project parts in other projects. Select a used module to change its usage options (accessibility, load mode, index usage, and shared package mounts). To use or unuse a module from this						
Project Infrastructure Main UML_Standard_Profile.mdzip Server source Libraries_(by Administrator) CDO 2.0 Libraries_2 5				tructure Main		
Edit Branches				x	le Load Mode	
Libraries_				_	Nways load	
Version	Author	Date	Comment		autoload with prompt	
		11/23/12 3:31 PM 11/23/12 3:31 PM 10/26/12 9:54 AM		<u>_ </u>	1anual load	
5	Administrator Administrator Administrator Administrator	11/23/12 3:37 PM 11/23/12 3:37 PM 11/23/12 3:37 PM 11/23/12 3:37 PM		-	Mounted On infra	
Rename	Set As L	atest	Compare	51	Open as Project Options	
		OK Close	Help		Cancel Help	

Changing module version

Resolving branch inconsistencies

Problem

You are using a version from the trunk and branch of the same project in your main project



Identifying branch inconsistency

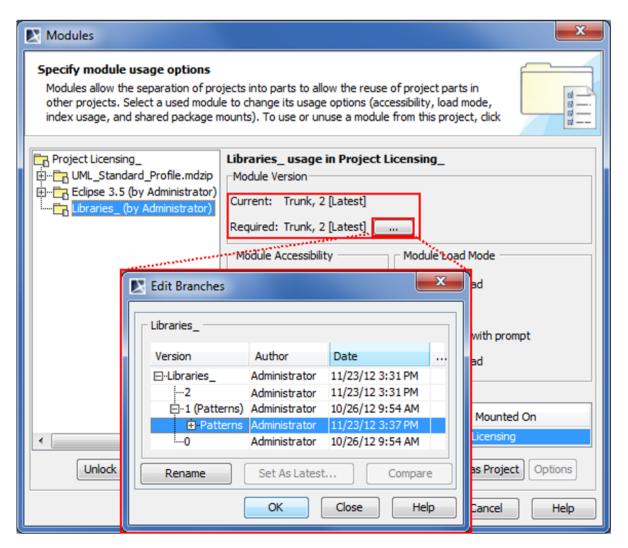
Solution

The Project Usage Map highlights these inconsistencies. You can then open projects with inconsistent usages and fix them by unifying the used project branch.

To resolve a branch inconsistency

Sure to check and resolve the model level usages between the project and module.

- 1. Open a project using the projects of inconsistent version.
- 2. From the Options menu, select Modules. The Modules dialog opens.
- 3. Select a module which version you want to change, and click the Lock button.
- 4. In the ModuleVersion area, click the ... button. The EditBranches dialog opens.
- 5. Select the wanted branch or trunk.
- 6. Click **OK** when you are done.

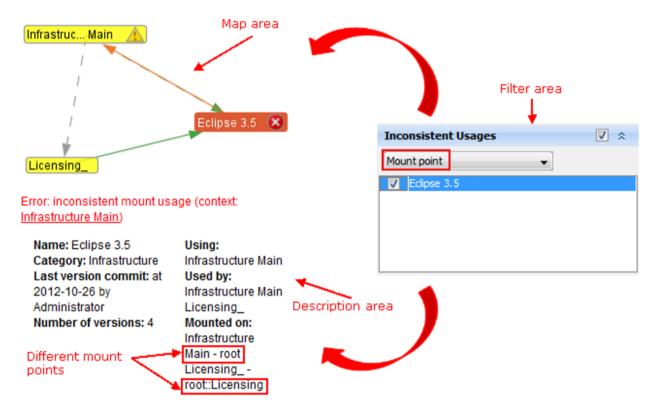


Changing module branch

Resolving mount point inconsistencies

Problem

You are mounting (mount - the other project usage in a particular package of the main project) the used project in different packages in your main project.



Identifying mount point inconsistency

Solution

The Project Usage Map highlights these inconsistencies. You can then open projects with inconsistent usages and fix them by unifying the used project mounted package information.

To resolve a mount point inconsistency

Sure to check and resolve the model level usages between the project and module.

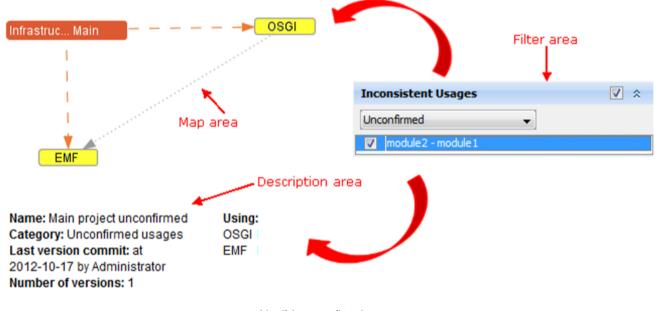
- 1. Open a project using the projects of inconsistent version.
- 2. From the Options menu, select Modules. The Modules dialog opens.
- 3. Select a module which version you want to change, and click the Lock button.
- 4. In the **Module Packages** table, select a wanted module.
- 5. In the Mount On cell, click the ... button. The Select Package dialog opens.
- 6. Select the wanted package and click **OK**.
- 7. Click **OK** when you are done.

Modules						
Specify module usage options Modules allow the separation of projects into parts to allow the reuse of project parts in other projects. Select a used module to change its usage options (accessibility, load mode, index usage, and shared package mounts). To use or unuse a module from this project, click the Use						
Project Licensing_ UML_Standard_Profile.mdzip Eclipse 3.5 (by Administrator) Libraries_ (by Administrator)	Library usage in Project Server architecture Module Version Current: Trunk, 4 [Latest] Required: Trunk, 4 [Latest] Module Accessibility Module Load Mode Autoload Mode Autoload Autoload Autoload Autoload with prompt Module Packages:					
	Shared Package Preferre	ed Path Mounted On Licensing				
		OK Cancel Help				

Changing mount point

Resolving unconfirmed usages

This type of usages is created automatically.



Identifying unconfirmed usages

Solution

Sure to check and resolve the model level usages between the project and module.

There are two ways to solve unconfirmed module usage situation - either confirm it or reject it.

If the module usage A => B is good and necessary according to the end-user policy, it can be confirmed. I.e., the user-defined usage is to be created in place of the current unconfirmed automated module usage.

To do that, use the Confirm and use the module into <module_name> solver of the validation result. This solver opens the standard Use Module wizard and pre-selects the required module. When wizard is completed, the necessary user-defined module usage is created.

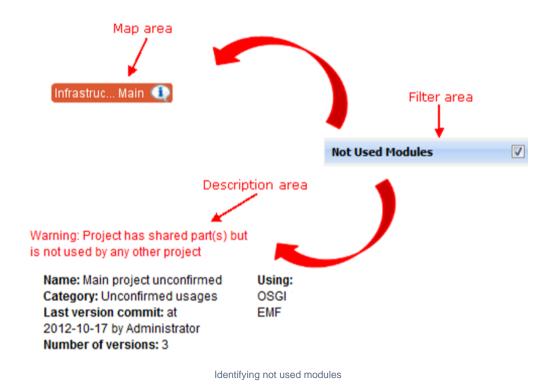
If the usage A => B is not good according to the end-user policy (for example – leads to module usage cycles or is incorrect because of semantically there should be no dependency between these modules), then it needs to be rejected and removed. To remove the usage, the model-level references, which are causing this automated module usage need to be changed – either removed or redirected to different elements.

Somore information about resolving the unconfirmed usages, see Resolving unconfirmed module usages.

Resolving not used modules problem

Problem

When the number of projects in the repository grows it is common for some of the projects to become outdated or not used anymore. You would prefer removing them BUT you are not sure if they are not used by some other, still-active project.



Solution

Sure to check and resolve the model level usages between the project and module.

The Project Usage Map highlights unused modules. Based on this information, you can move all of the unused modules into the deprecated category or remove them entirely from the repository.

Related Pages:

Unknown macro: 'list-children'