|  |
| --- |
| CEA LIST |
| Constraint Generation from the profile Developper Guide |
| [Tapez le sous-titre du document] |

|  |
| --- |
| TESSIER Patrick  12/04/2013 |



Sommaire

[Requirements 3](#_Toc353530368)

[Use Cases 3](#_Toc353530369)

[1. Generate constraints directly into the definition 3](#_Toc353530370)

[2. Generate constraints as EMF validation plugins 3](#_Toc353530371)

[2.1. Constraint in OCL 3](#_Toc353530372)

[2.2. Constraint in Java 3](#_Toc353530373)

[Solution proposition 4](#_Toc353530374)

[Test Use case 5](#_Toc353530375)

# Requirements

**ValidationReq001**: Papyrus shall validate constraints from profiles on models

**ValidationReq002**: User can precise on constraints if they may be validated live or batch mode, the associated error message and if it is activated by default

# Use Cases

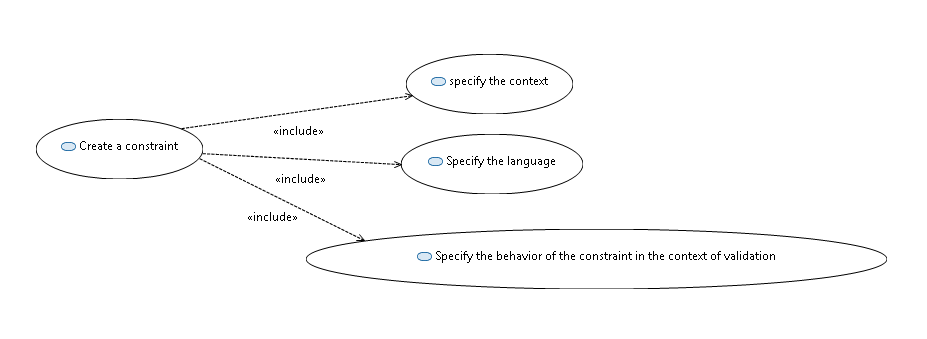
In order to validate constraints from a profile, first creation of constraint with specific annotations will be explained and then generation of validation will be explained.

Figure 1: Constraint creation

## Create a constraint

In the context of the construction of a profile, it is possible to add constraints.

A constraint is a condition or restriction expressed in a language for the purpose of declaring some of the semantics of an element.

#### How to

In the context of a profile diagram you can create a constraint from the palette.

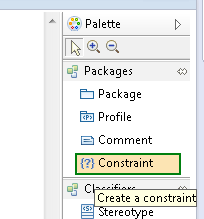


Figure 2: Constraint creation from the palette

From the model explorer, click right 🡪 new child🡪 Create a new Constraint

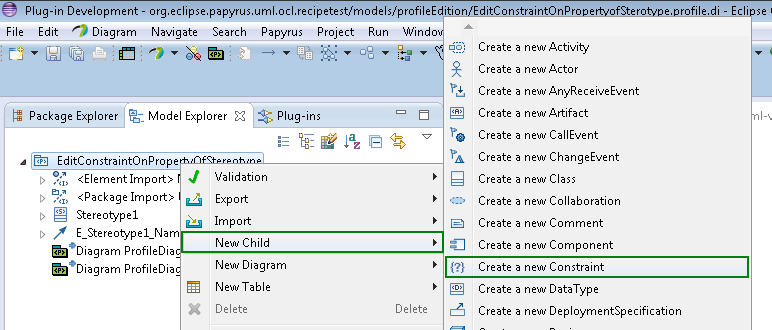


Figure 3: Constraint creation from the model explorer

## Specify the context

Before editing a constraint, it is important to specify the constraint; it can be a stereotype, an operation.

#### How to

From the diagram, a context link can be added between the constraints and the context of the constraint:

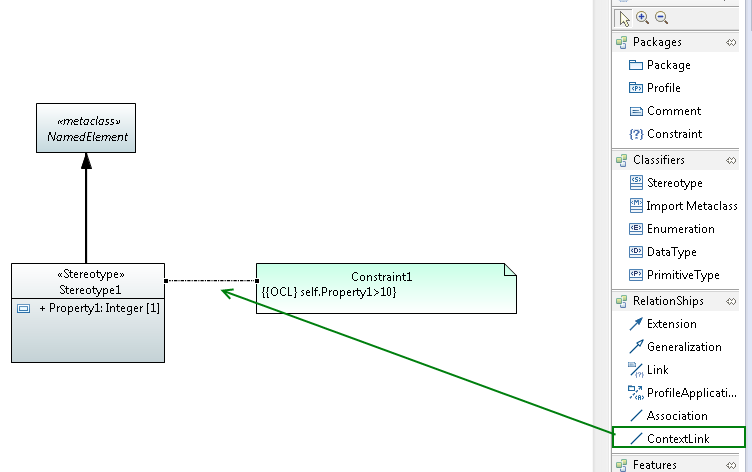


Figure 4: Context link creation in a the profile diagram

From the property view, the context can be specified:

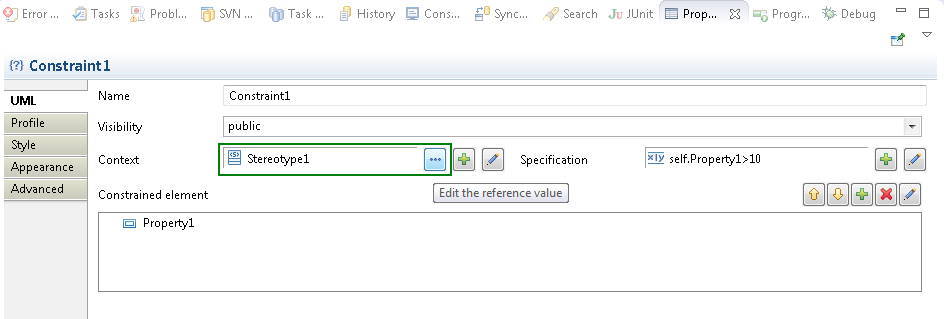


Figure 5: specify context from the property view

## Specify the language

It is possible to precise the language of the constraint into Papyrus. The constraint body may be written into OCL, JAVA, or natural language. In order to make constraints usable by the Papyrus, constraint must be written in OCL or JAVA.

#### How to

By default if your try to edit a constraint, de default language will be OCL.

At the creation, after set the context, you can choose the editor for the constraint:

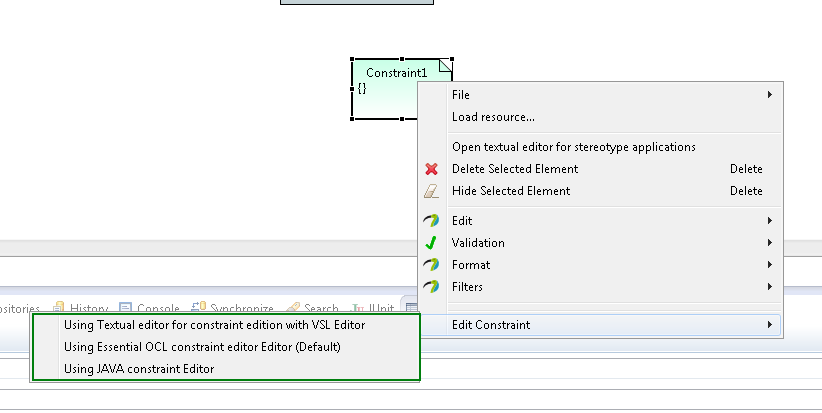


Figure 6: choose the editor for the constraint

You can also specify the language in the property view:

Select the field specification nad choose create or edit the asscoated opaque expression:

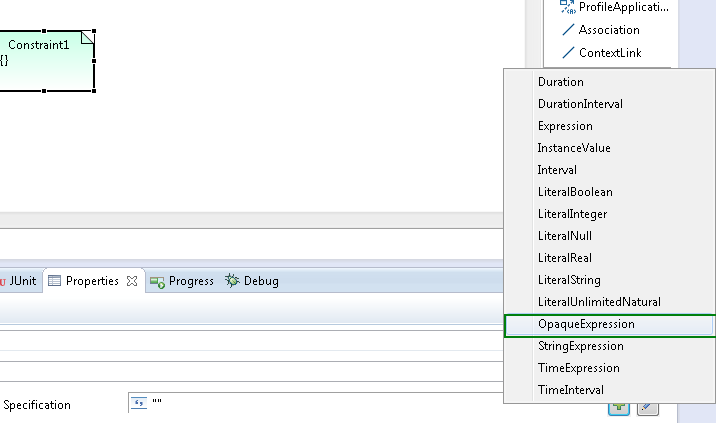


Figure 7: creation of a specification for the constraint

You can add the language, see following figure:

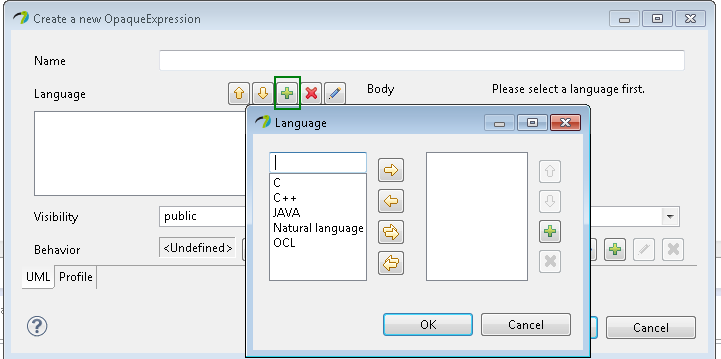


Figure 8: specify the language

Then if you want you write the body in the specified language:

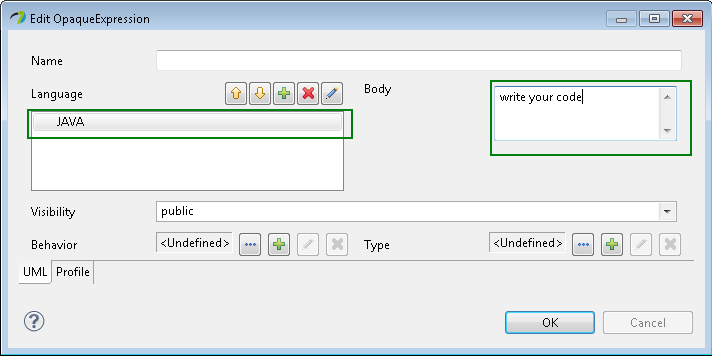


Figure 9: Write the body of the opaque expression

## Specify the behavior of the constraint in the context of validation

Into Papyrus, the behavior of the constraint during the validation can be precise.

It is possible to precise:

* The mode, if the validation of the constraint can be done, in t “batch” or “live” mode.
* The severity: The severity of the problem if the constraint is violated. This correlates to the severity of tasks in the Tasks view of the Eclipse environment. The default severity (if none specified) is ERROR. The CANCEL severity should be used with caution, as it causes the validation operation to be interrupted, possibly resulting in the loss of valuable diagnostic information from other constraints.
* The message: the message that will be displayed if the constraints is violated
* The description: a description of the constraint
* If it is enable by default: Indicates if this constraint should be enabled by default.

For advanced user you can also precise:

* The Id : the constraint id
* The status code: The plug-in unique status code, useful for logging.
* the target of validation, the element to be validated

#### How to

***Apply the DSML validation profile***

1. Select the profile root and apply the profile
2. Select the profile tab in the property view
3. Click on the registered profile button
4. Selection the DSML validation profile

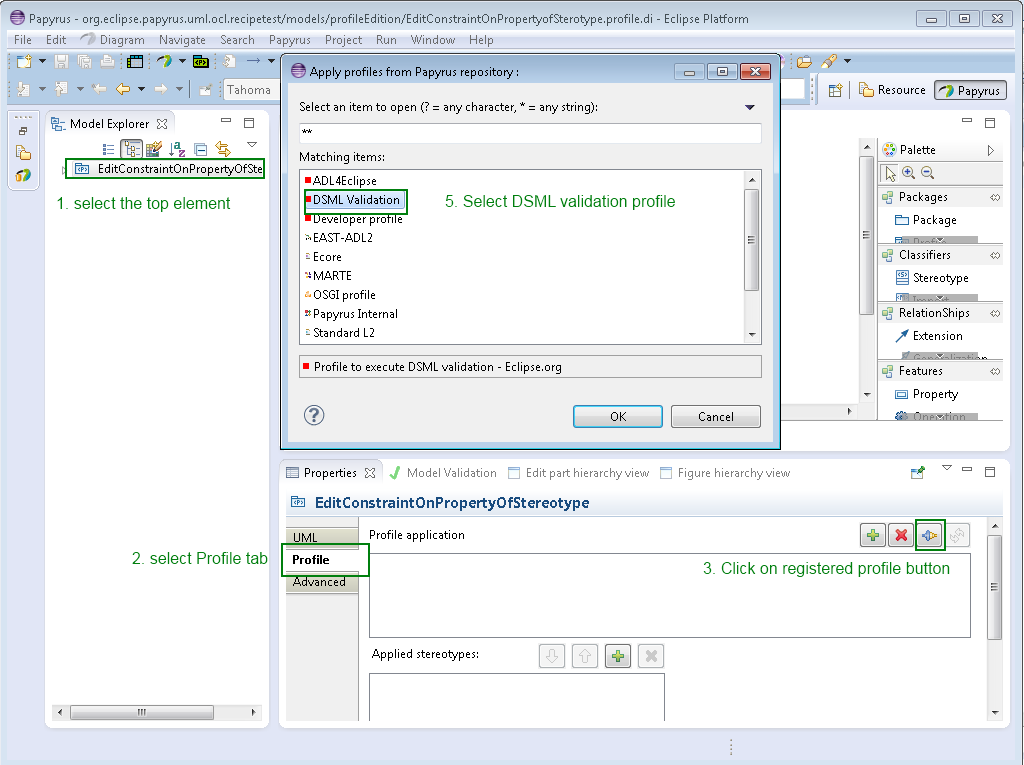


Figure 10: apply DSML profile

**Apply Stereotype on constraint**

1. Select the constraint
2. Select profile tab of the property view
3. Click on apply profile button
4. Select validationRule stereotype

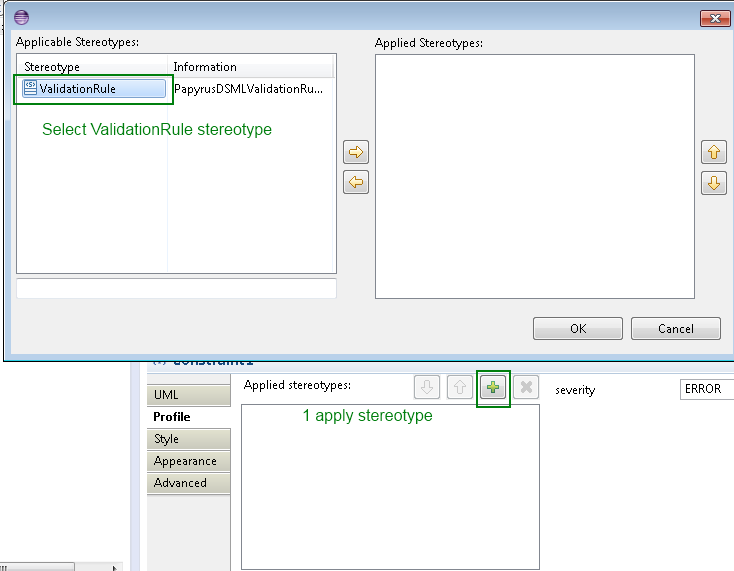


Figure : apply validationRule sterotype

Then you edit property of stereotype to add information about the behavior of the validation

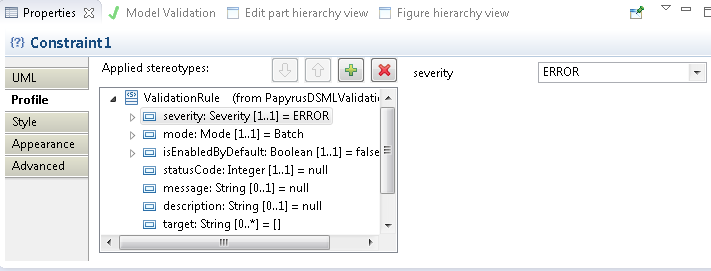


Figure 12: edition of the property of stereotype

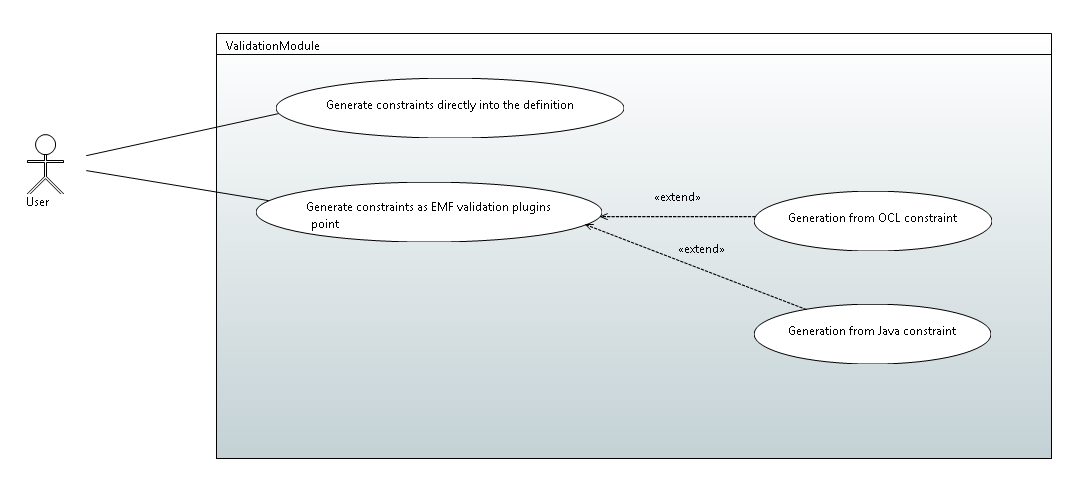


Figure : Constraint generation

## Generate constraints directly into the definition

Constraint written in OCL in the profile can be generated into the definition of the profile and taken in charge during the validation of the model.

The problem with this use case is to know if it is possible to respect the requirement ValidationReq002.

#### How to

When you save the profile,Papyrus asks to you if want to define the profile.

Click on yes and then do not forget to check save OCL constraint in the definition

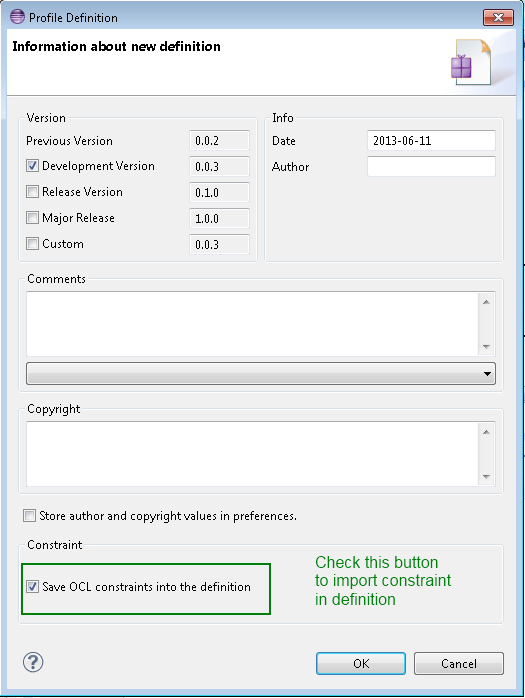


Figure 14: save OCL constraints in the definition

## Generate constraints as EMF validation plugins

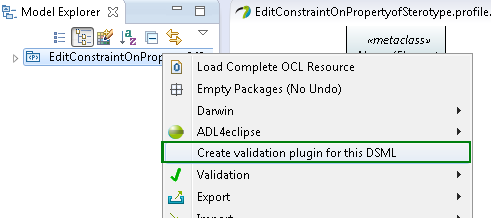
The user can generate plugins that wrap constraint and can be used in the EMF plugin validation.

The constraint can be generated in Java code, or directly from OCL.

#### How to

Select the root profile

Click right on the model explorer🡪 create validation plugin for this DSML



Then a wizard to create the validation is launched. You have to give a name for you validation plugin.

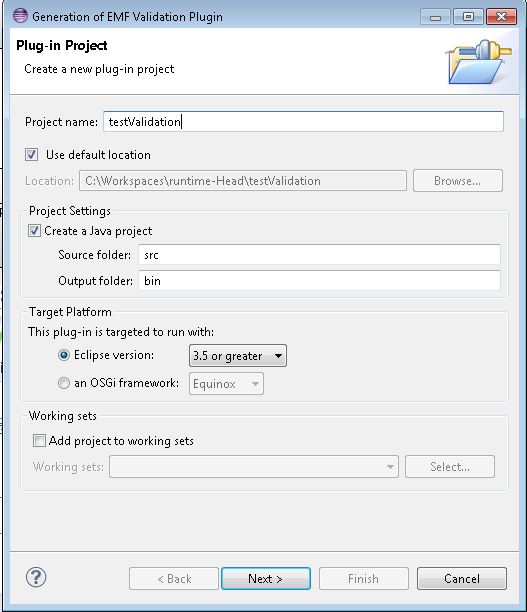


Figure 15: wizard of constraint validation creation

The you need to install the created plugin.

# Solution proposal

# Test Use case