MDE Tool

1. Tester le plugin

To conduct plugin testing within Visual Paradigm, proceed as follows:

* Download and install Visual Paradigm.
* Copy the "Code Generation" folder from the "MDEPlugin" directory and paste it into the "plugins" folder of Visual Paradigm. For instance, the path looks like: "C:/Users/<user\_name>/AppData/Roaming/VisualParadigm/plugins".
* Reload Visual Paradigm.

For further details on how to use the tool, please consult the "Guide.docx" document in the documentation.

1. MDE Tool guide.

Code Generation using the MDE Tool involves three steps: architecture specification, PIM parameterization, and code generation.

1. Architecture specification:

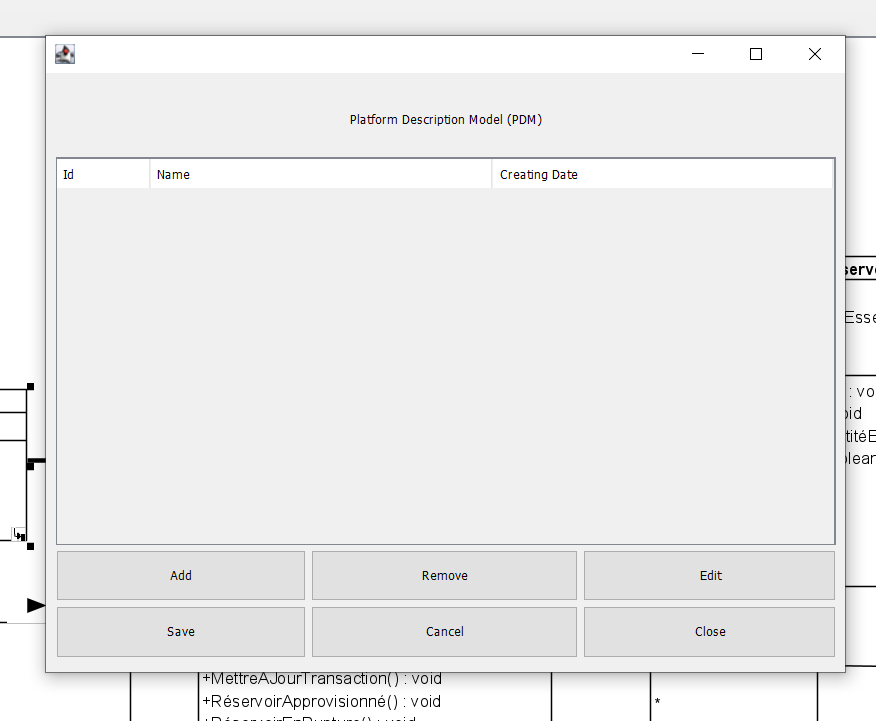
To specify the architecture, follow these steps:

* Open Visual Paradigm, then click on "Platform Description Model" in the "MDE" tab.

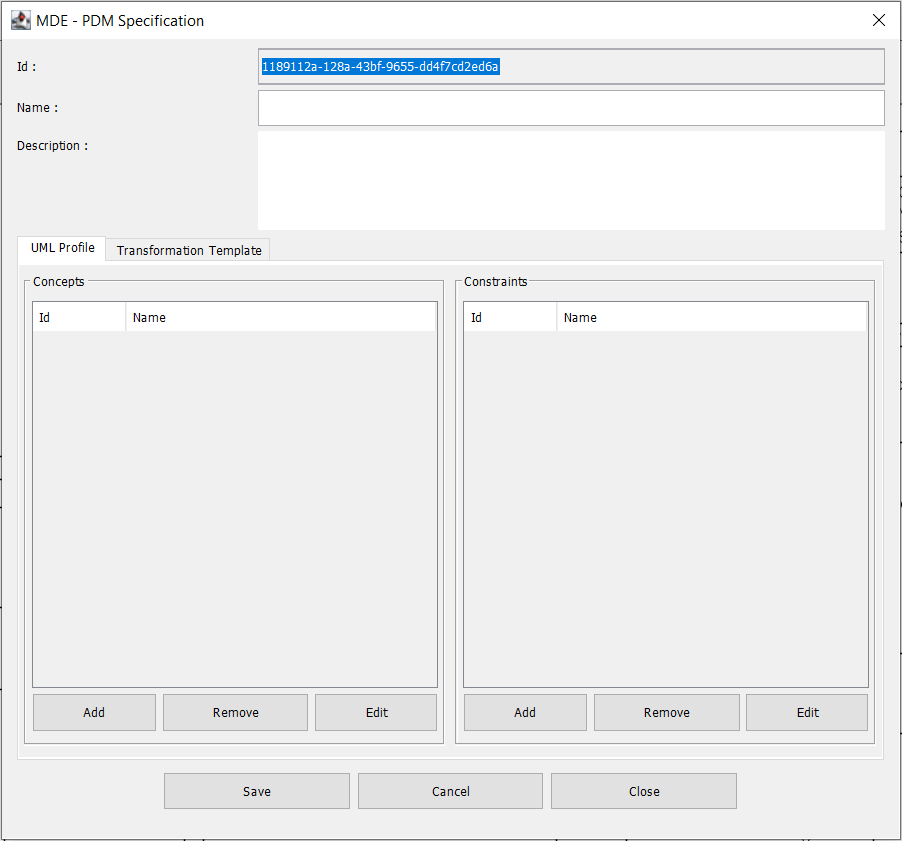
Une image contenant texte

Description générée automatiquement

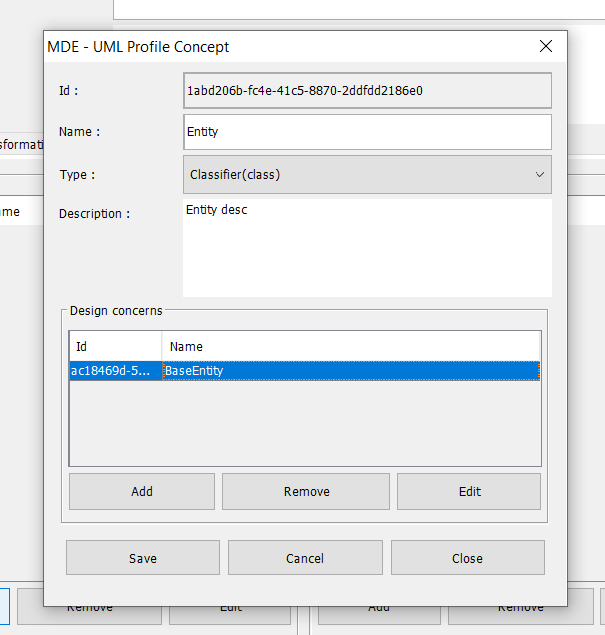
* Click on "Add" in the main window to add a PDM.



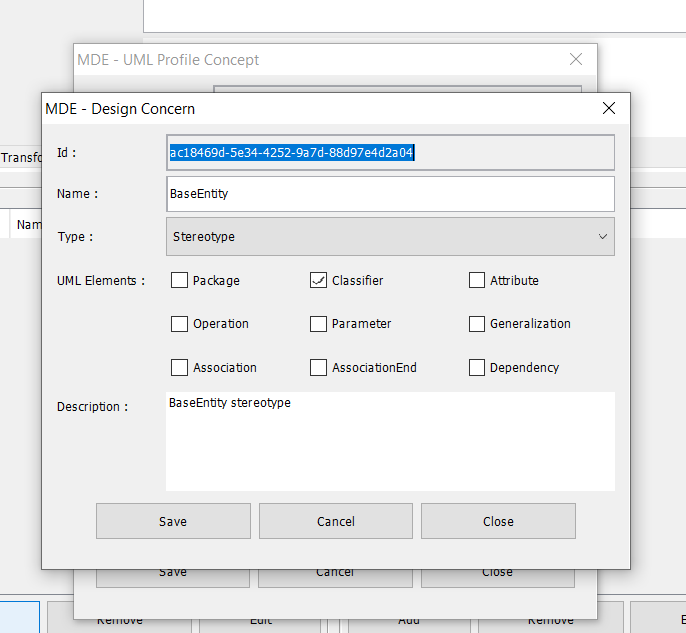
* A PDM is defined by a UML profile and a set of transformations. Click on "Add" in each section to add a UML profile concept or constraint, or a transformation.



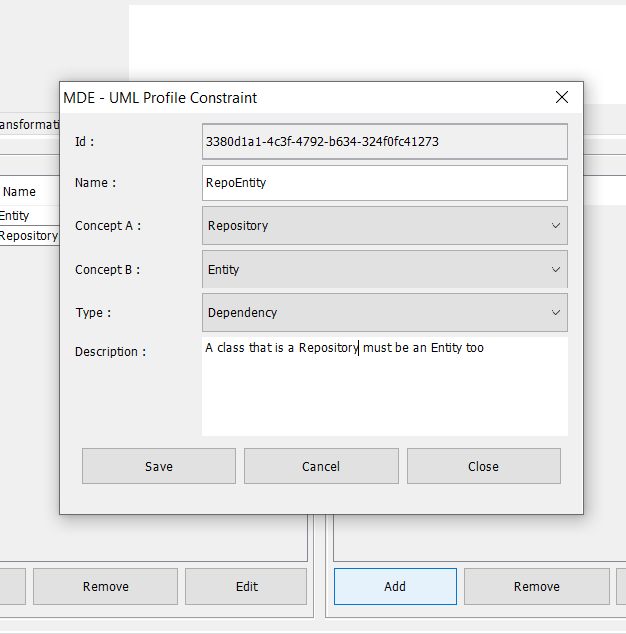
* To add a concept, enter: the name, type, description, and design concerns. To add a design concern, click on "Add" in the "Design concerns" section.



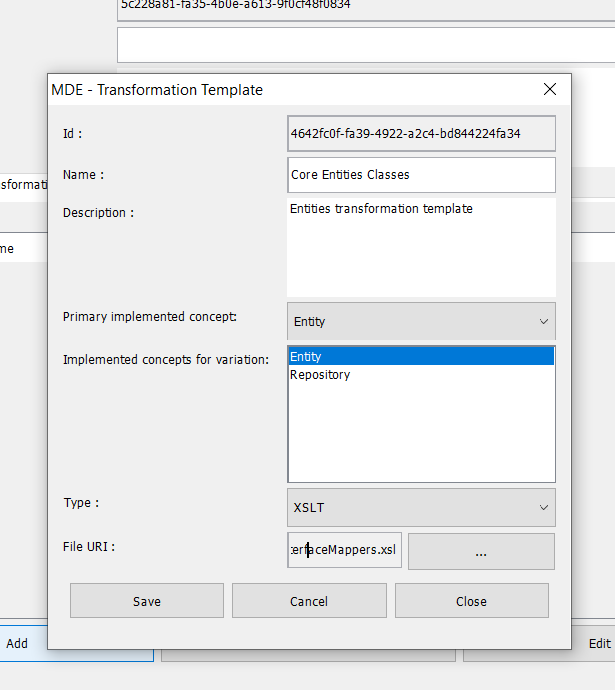
* To add a "design concern," enter the name, type, concerned UML elements, and description.



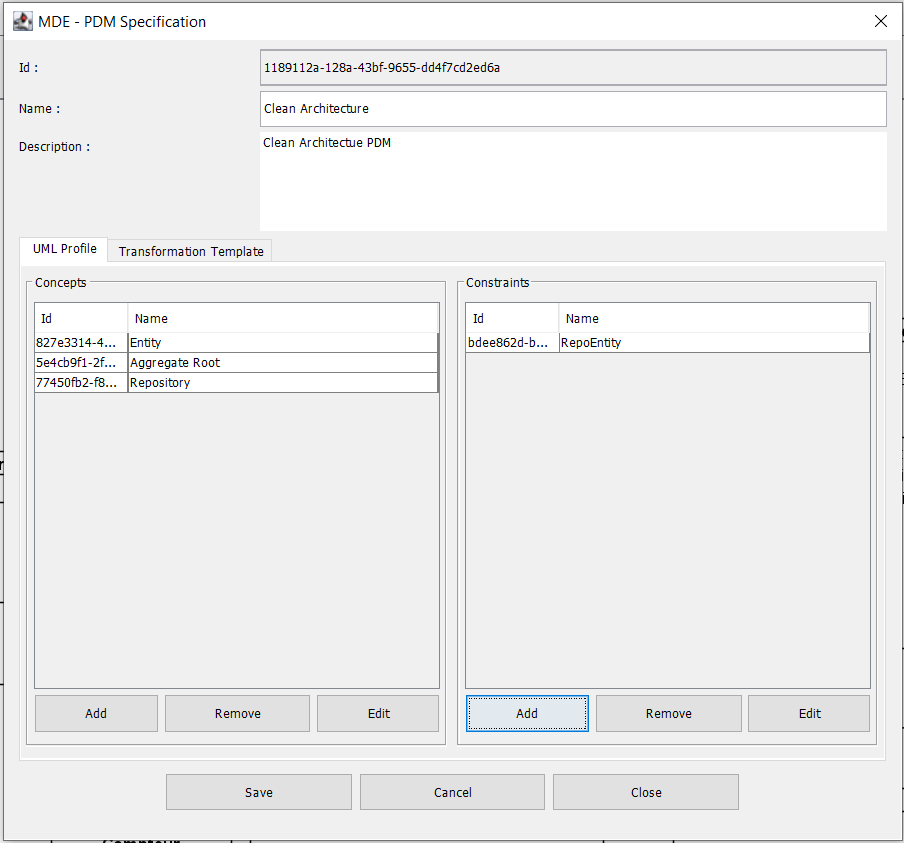
* After defining the concepts, we need to establish constraints between them. To add a constraint, enter the name, the first concept, the second concept, the type, and the description.



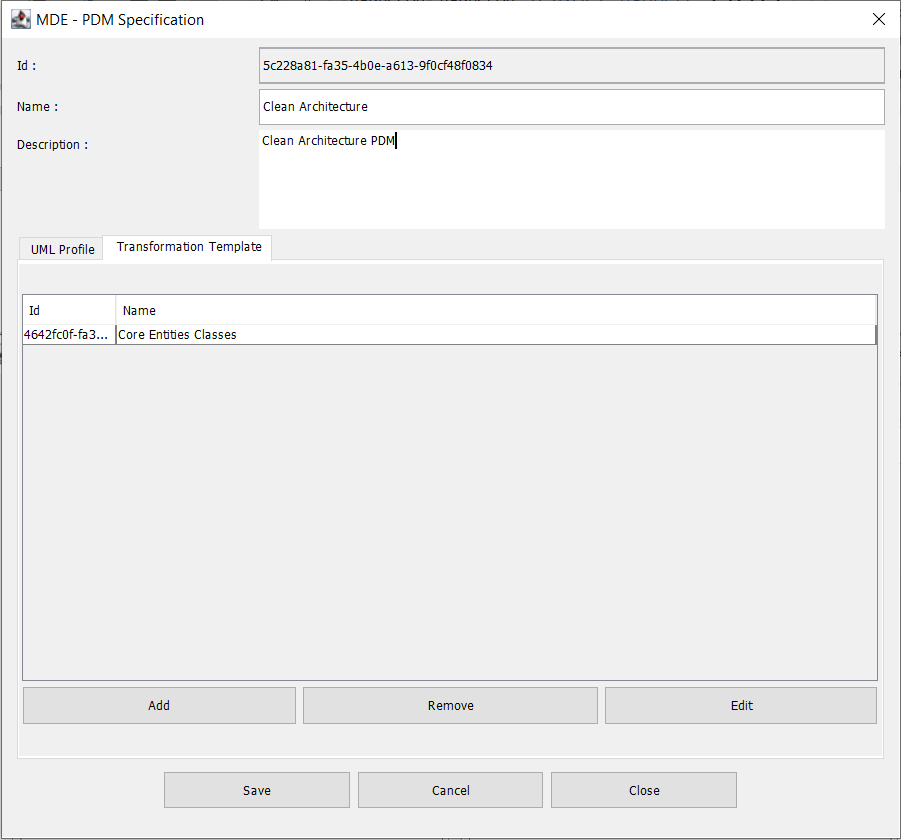
* The second part of architecture specification involves defining transformations. To add a transformation, enter the name, description, main implemented concept, implemented concepts for variation, type, and the path to the transformation.



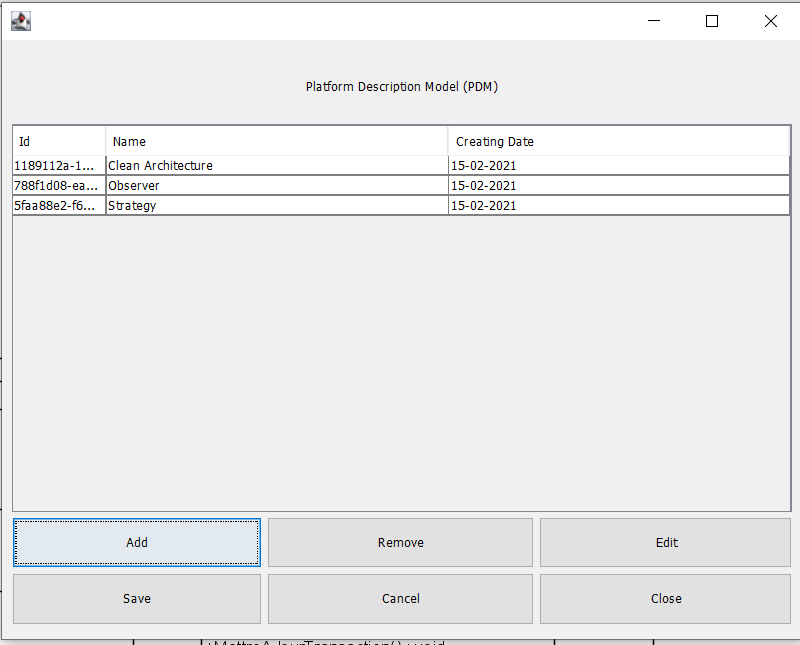
* Here is the UML profile after its definition. The user can add, modify, or delete a concept or constraint.



* Here are the transformation models after their definition. The user can add, modify, or delete a transformation model.
* After architecture specification, click on "Save".



* After adding the "clean architecture" PDM, I added others for design patterns (observer and strategy). The user can add, modify, or delete a PDM.
* Finally, click on "Save" to save the PDMs.

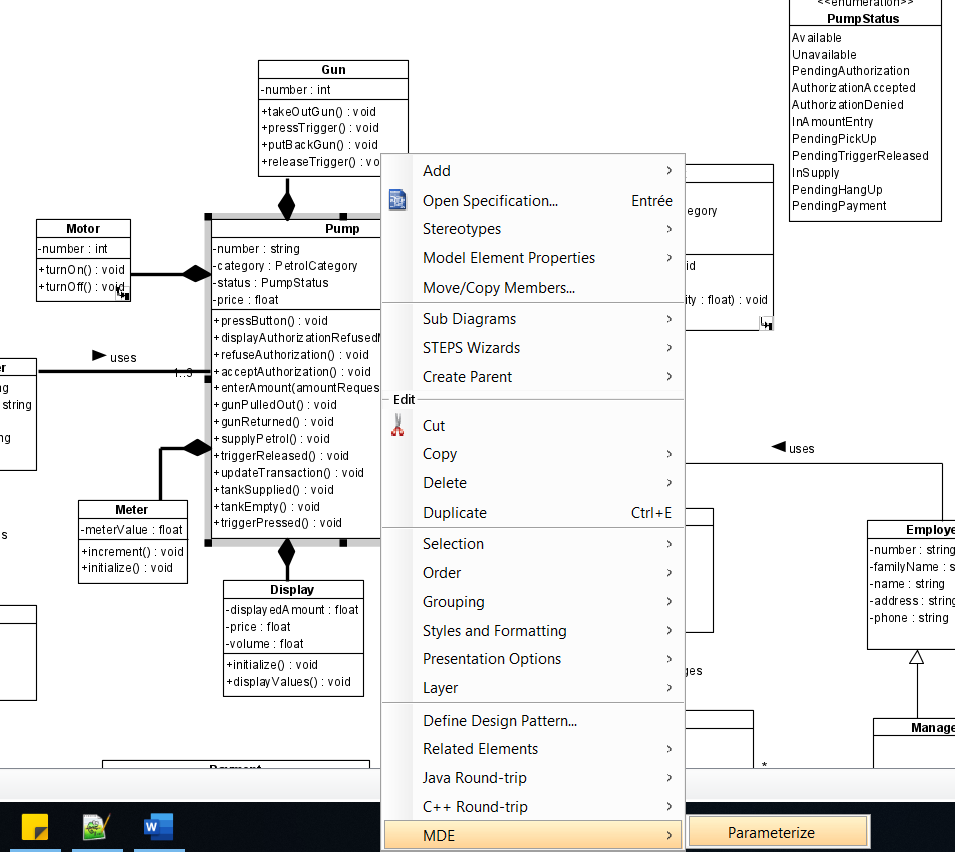


* Les The PDMs are saved in XML format in the following path:

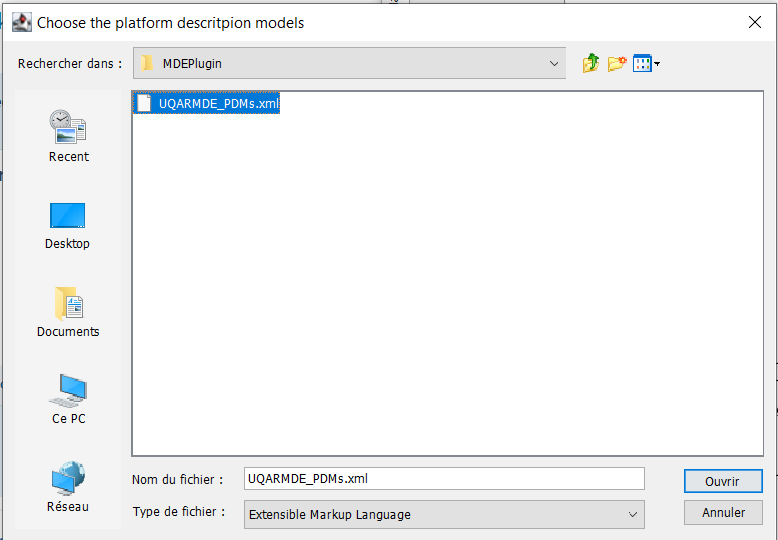
C:\Users\<User\_Name>\Documents\MDETool\PDMs\<PDM\_Name.xml>

1. PIM parameterization:

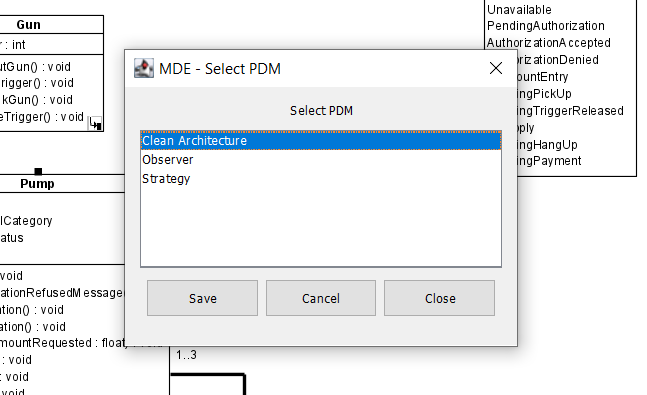
* Select the UML elements to be parameterized, then right-click and choose "MDE," and then click on "Parameterize." In the following example, I have selected the "Pump" class.



* Choose the previously defined PDMs.



* Select one or more PDMs to parameterize the PIM and click on "Save." In this example, I have chosen only the "Clean Architecture" PDM.

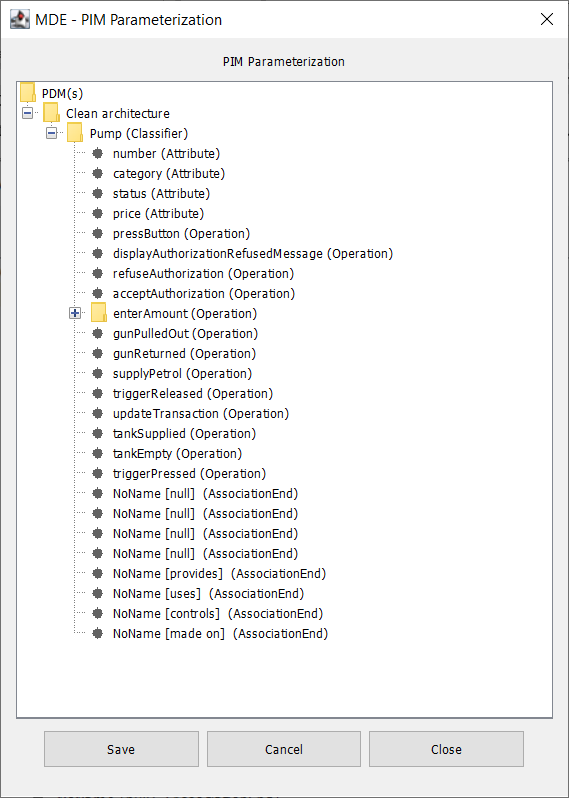


* The opened window consists of a hierarchy of elements. The root nodes of the hierarchy are the chosen PDMs. The branches from the root nodes represent the UML elements selected in Visual Paradigm. Furthermore, certain selected UML elements will have sub-branches, leading to:

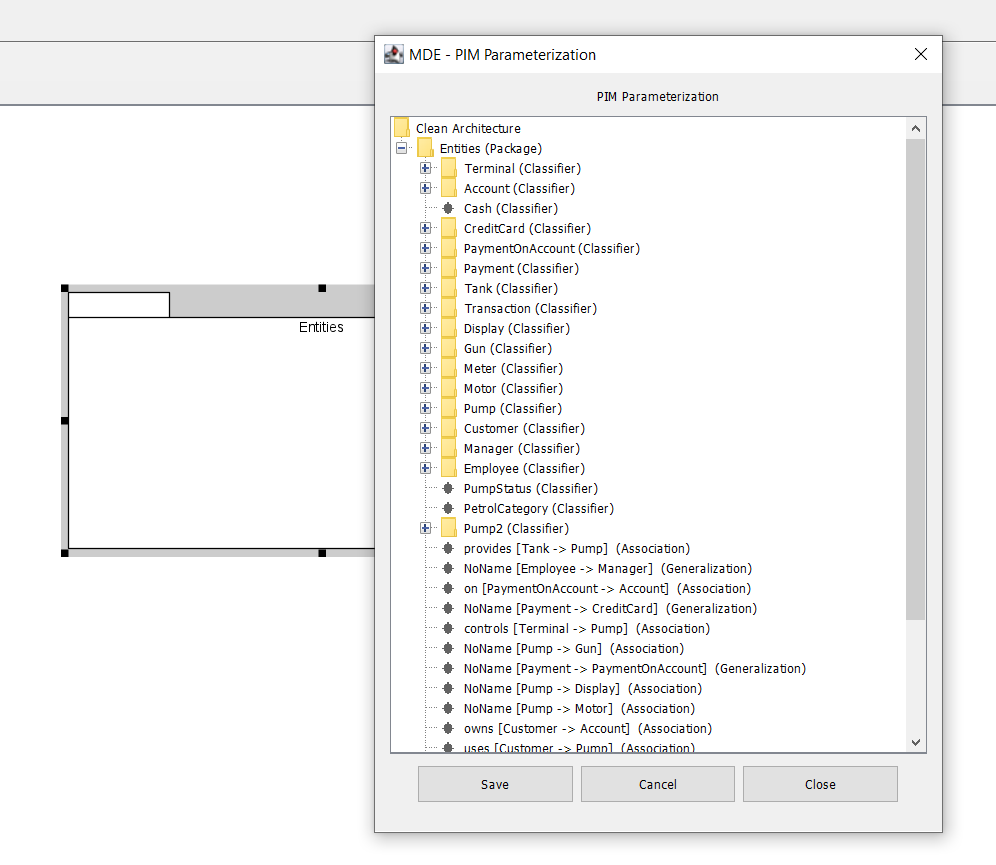
- If the selected UML element is a package, its branches will consist of classes and associations.

- If the selected UML element is a class, its branches will include attributes, operations, and association ends.

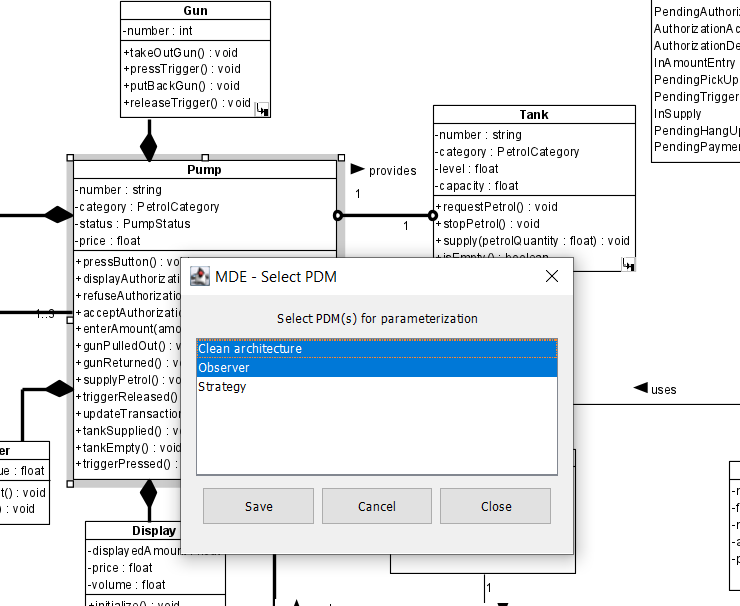
- If the selected UML element is an operation, its branches will include parameters.



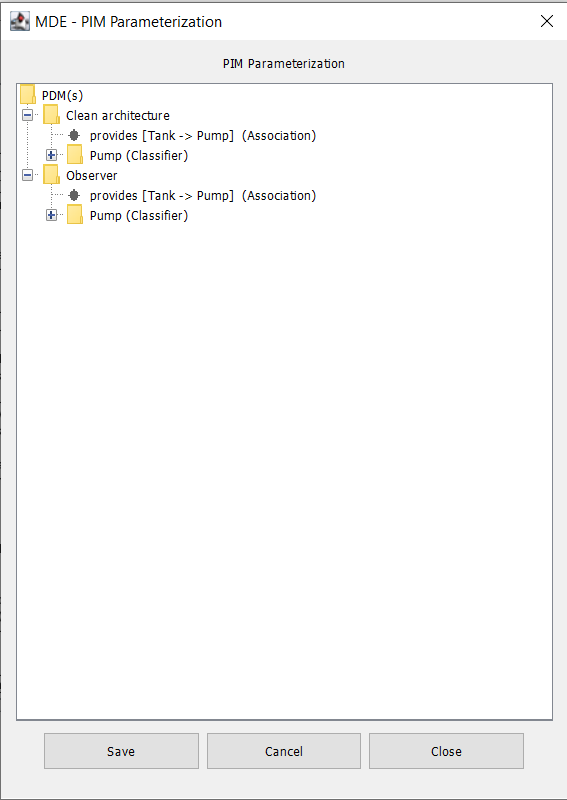
* Here is the hierarchy of elements when I selected a package for parameterization.



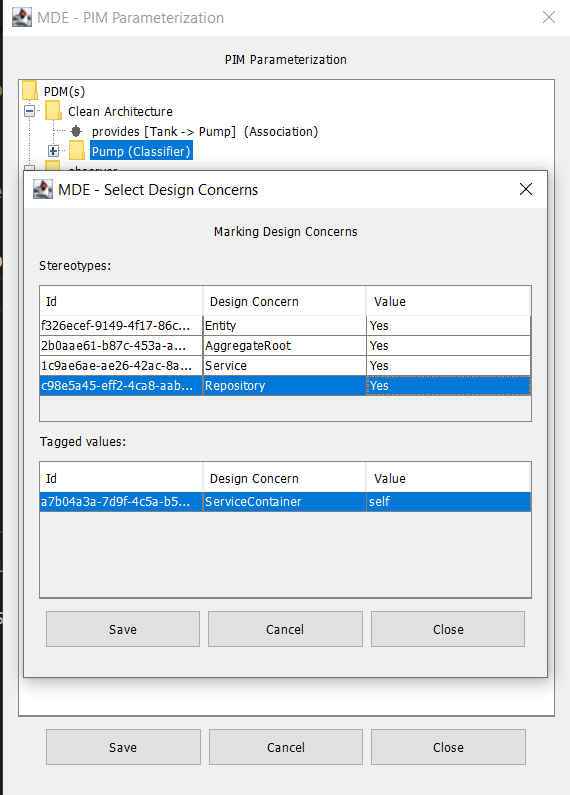
* The user can choose multiple PDMs and select multiple UML elements simultaneously. For instance, I selected the PDMs "Clean architecture" and "Observer," and I chose the class "Pump" and the association "provide" (between "Pump" and "Tank").



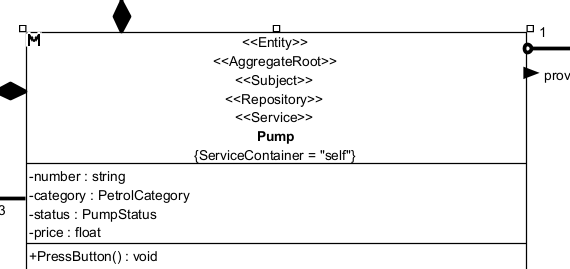
* The tool will list each chosen PDM, and for each PDM, it will list the selected UML elements. Therefore, the user can parameterize a UML element with multiple PDMs simultaneously.



* Click on a UML element to parameterize it.
* In the following example, I clicked on the "Pump" class within the "Clean architecture" PDM. Subsequently, a window opens containing the applied design concerns for UML elements of type "Classifier."
* For stereotypes, select "yes" or "no" as the value. And for tagged values, enter the desired value.
* Click on "Save."



* The user can do the same thing for the "Pump" class, but with the design concerns from the "Observer" PDM.
* After saving the parameterization, the user can visualize the result in the PIM.

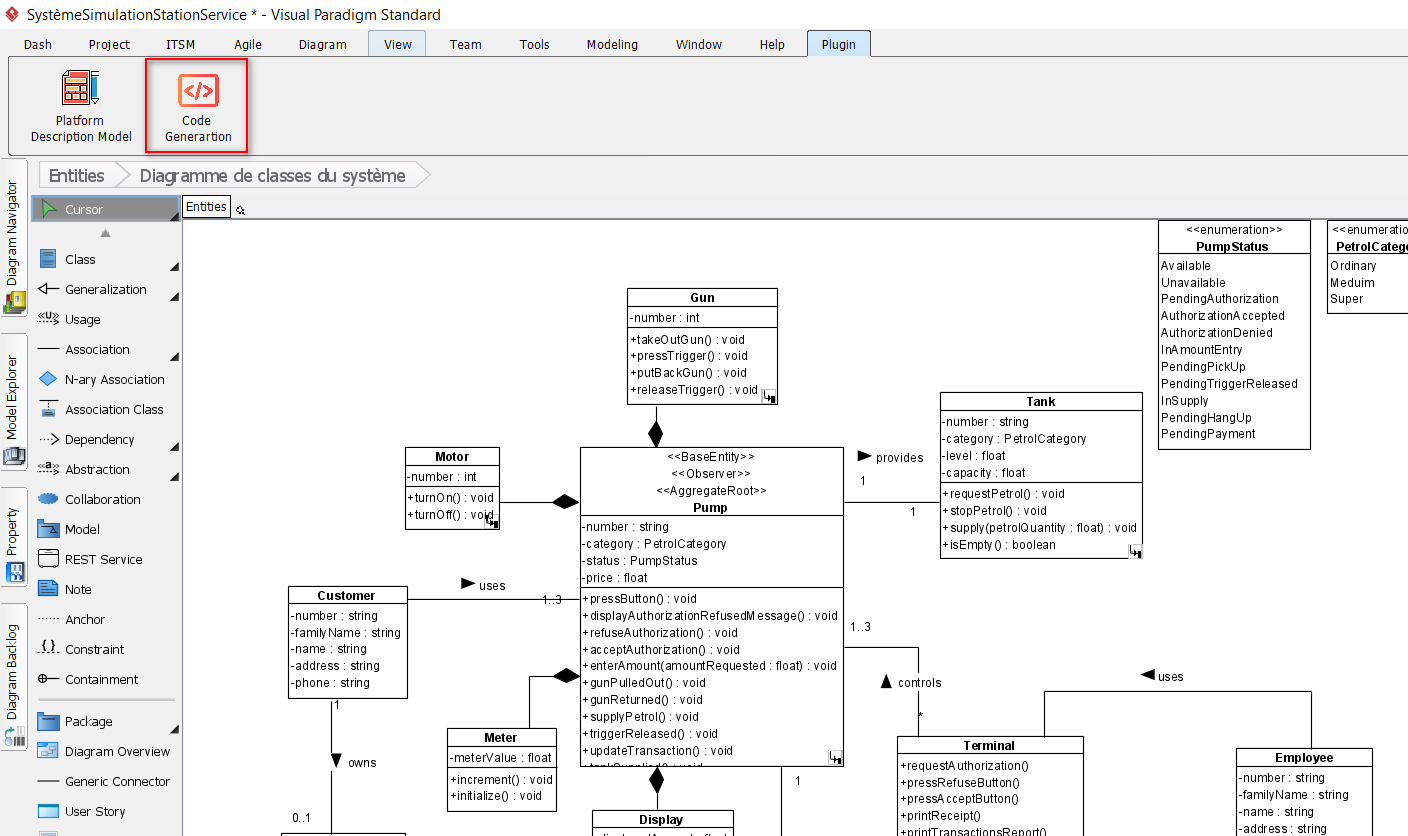


* The XML file containing the result of the parameterization is saved in the path:

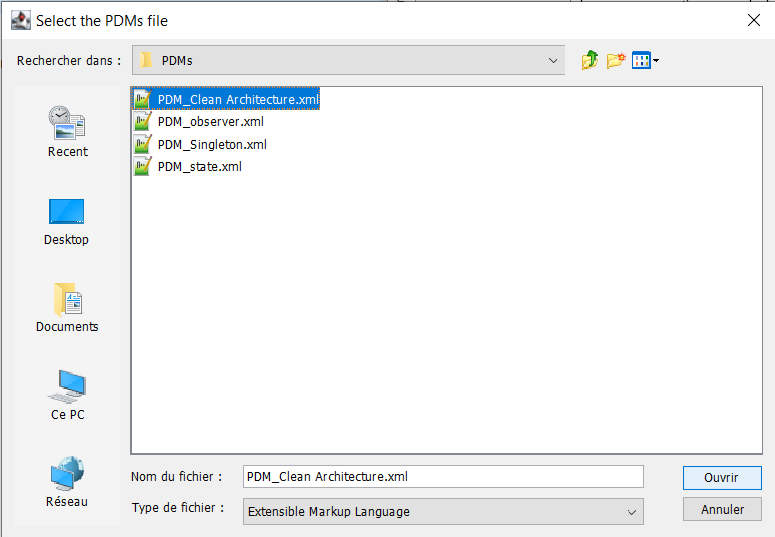
C:\Users\<User\_Name>\Documents\MDETool\<VP\_Project\_Name>\<VP\_Project\_Name>\_PIMParametrization.xml

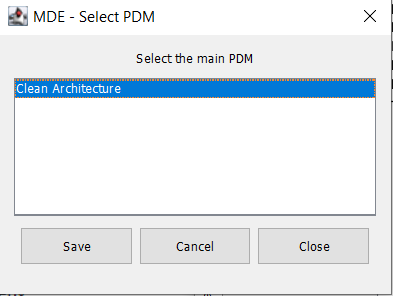
1. Code generation:

* To generate the code, click on "Code Generation" in the "MDE" tab.

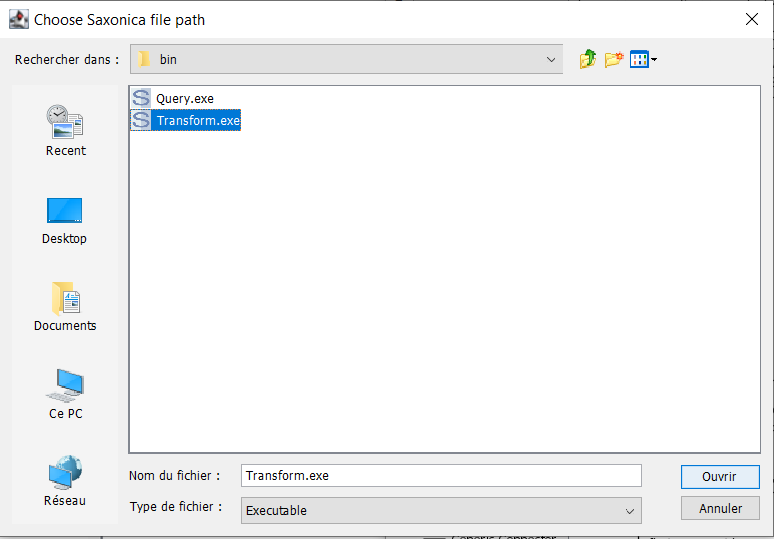


* Choose the primary PDM containing the transformations, from the previously specified PDMs within the directory labeled "…/Document/MDETool/PDMs."

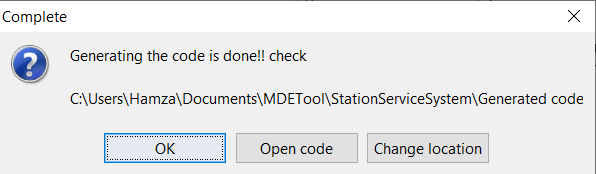




* Select the path to the transformation tool "Saxonica." For example:

"C:\Program Files\Saxonica\SaxonHE9.9N\bin\Transform.exe".

* Wait for the code generation to complete, then click on "Open code" to navigate to the code location.



* The code is generated in the folder: « …/Documents/MDETool/<project\_name>/Generated code ».
* Alongside the code, you will find transformations and other files created by Visual Paradigm, including the input model file "project.xml."

