EMOOSE 2007

Module: Model Driven Engineering

Instructor: Rubby CASALLAS

Teaching Assistant: Hugo ARBOLEDA.

---

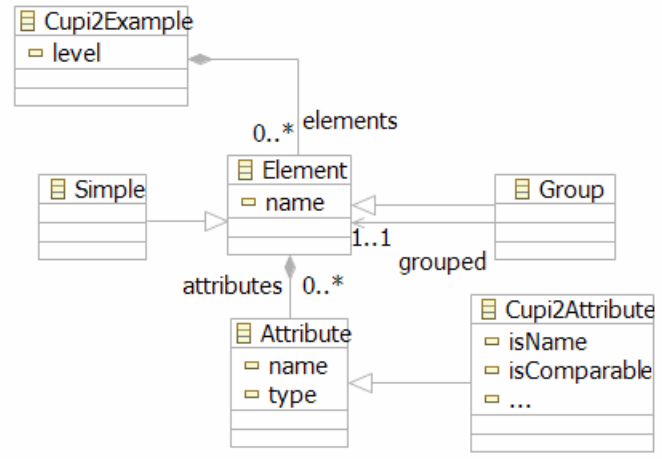
1 Purpose

The purpose of this tutorial is to get some familiarity with the EMF framework. Based on a simple example we are going to create a (meta)model and then to generate the Eclipse plug-ins to create and edit models conform to it.

This document is based on the Eclipse-EMF documentation[1].

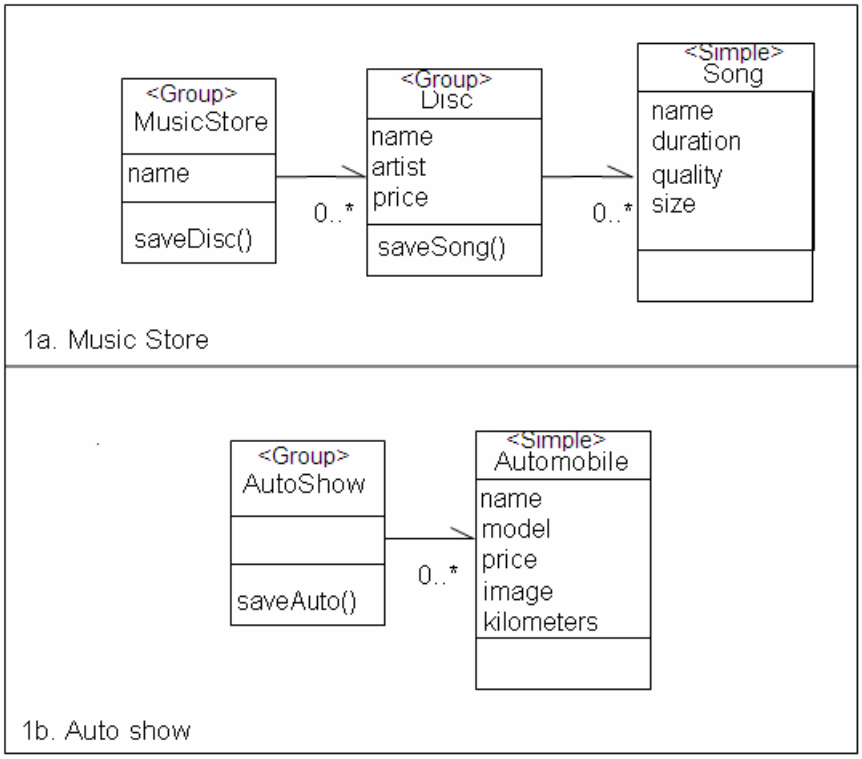
2 Example

For this tutorial, we have to create a Platform Independent Metamodel (PIMM) for the Cupi2 example.



By using this PIMM we'll be able for creating PIMs as the next.

[1] http://help.eclipse.org/help32/index.jsp?topic=/org.eclipse.emf.doc/references/javadoc/org/eclipse/emf/ecore/



3 Pre requisites

Please ask for the Eclipse configuration already prepared. This configuration contains:

- Eclipse Versión 3.3.1.1
- All the Model development tools (from the eclipse Europe updates).
  - EMF v2.3
  - GMF v1.0.1
  - UML2 v2.2.1
  - All the other stuffs.
- oAW 4.2

4 Creating an EMF model

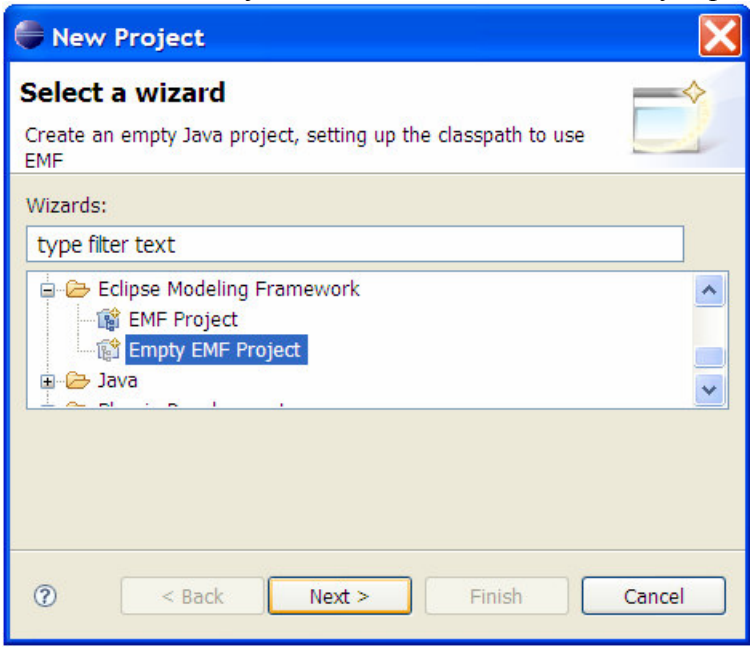
It is possible to create EMF models departing from:

1. *A manually created XMI document (for example an ecore model)*
2. *An XMI exported from a modeling tool as Rational Rose.*
3. *Java interfaces with annotations.*
4. *An XML scheme that describes the serialization of the model.*

This tutorial focuses on the first; thus, we are going to create an Ecore model and then, to generate the EMF model.

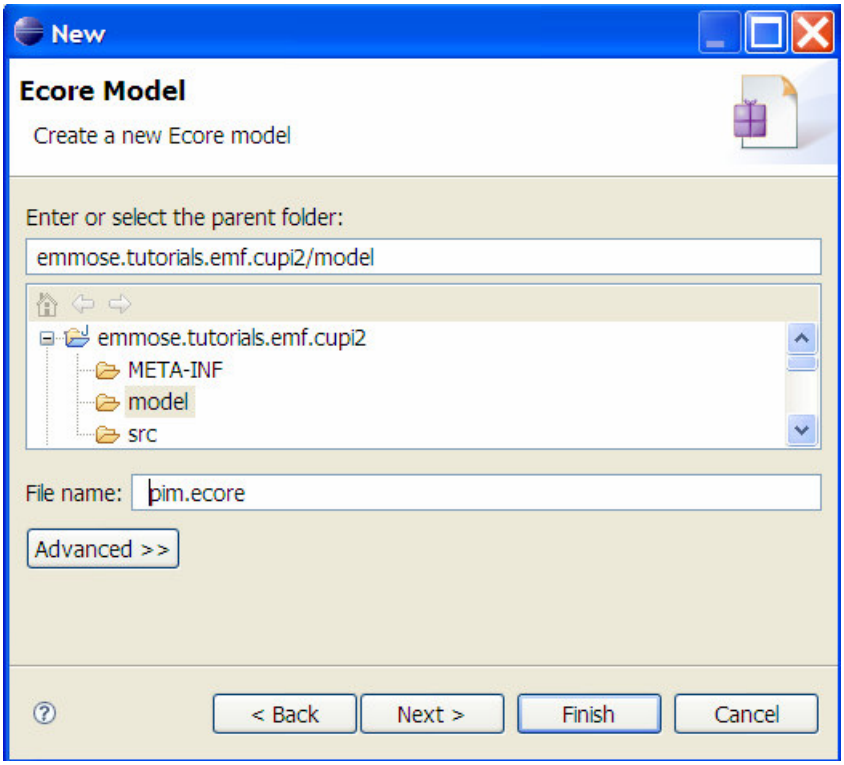
4.1 *Project creation*

The first step is to create an *EMF Project*. Call it *emmose.tutorials.emf.cupi2*.



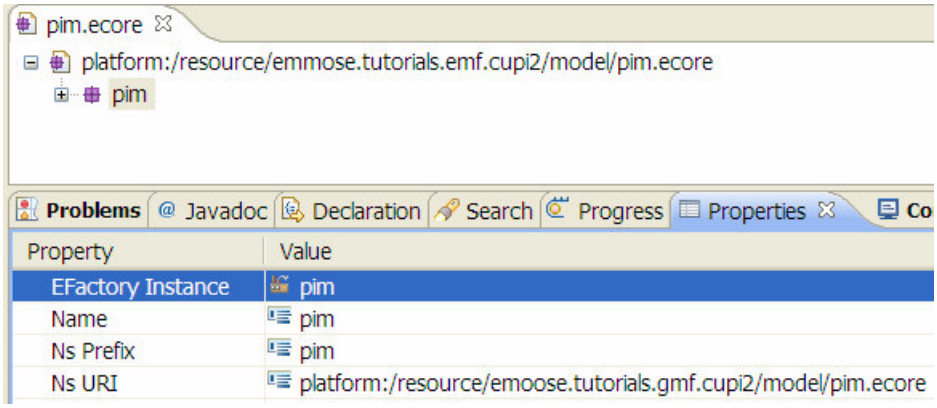
4.2 *Metamodel creation*

Now, inside the folder named *model* you have to create an Ecore model (File -> new -> other -> Example EMF Model Creation Wizard -> Ecore Model).



4.2.1 Root Package

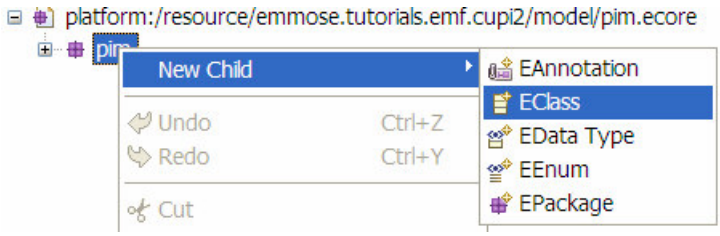
The general properties of the root package have to be defined from the beginning at the creation of the metamodel concepts and relationships.



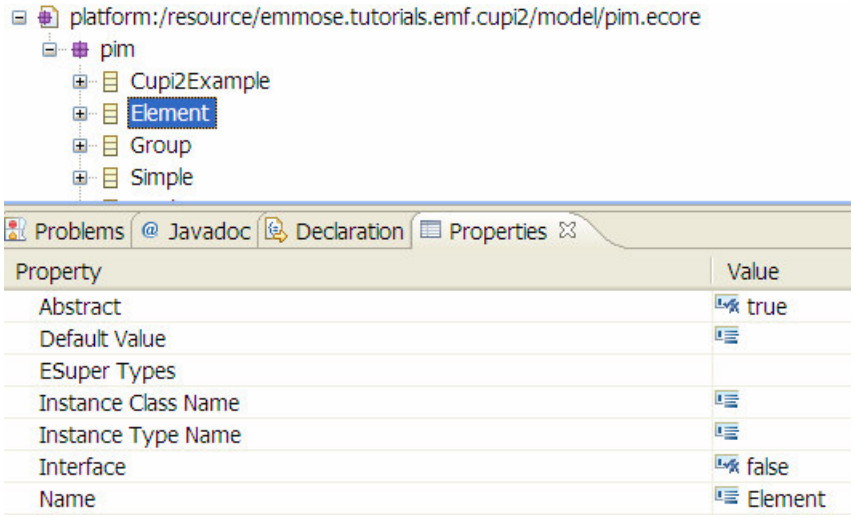
4.2.2 EClasses

Here we explain the creation of 4 *EClasses (Cupi2Example, Element, Group and Simple)*. As part of this tutorial the student should finish the creation of the metamodel (*EClasses Attribute* and *Cupi2Attribute*).

1. The three (4) *EClasses* must be created in the automatically created package *model.*
2. Please press right click on the package and select the option of creating New Child EClass. (Right click -> New Child -> EClass).

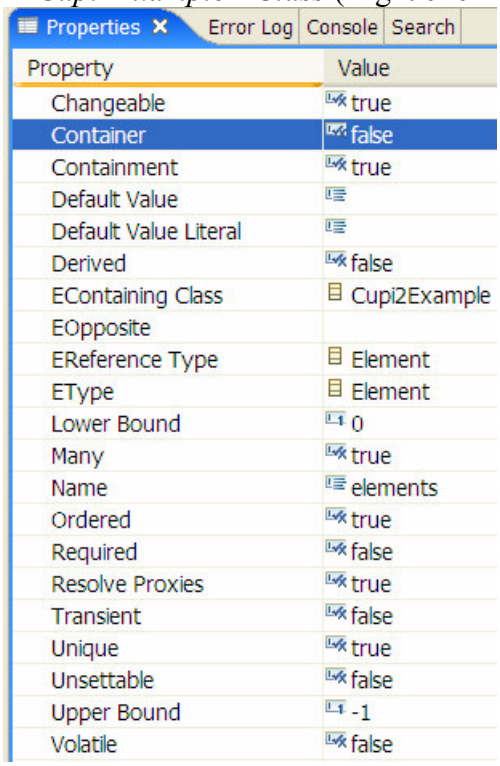


3. Name the *EClass Cupi2Example* and repeat the same process for the creation of the *Element, Group and Simple EClasses.*
4. Since the *EClass : Element* is abstract, then its property *Abstract* must be set to *true*.



4.2.3 EReferences, EAttributes and inheritance-relationships

1. The first relationship created is the one between *Cupi2Example* and *Element*. It is important to note that always should exist a main container *EClass*, in this case it is *Cupi2Example*. For the creation of this relationship an *EReference* is created for the *Cupi2Example EClass* (Right click -> New Child -> …). Then the properties are defined:



Name = elements
EType = Element
This means that the *EClass Cupi2Example* has a reference to a collection of *Element* called *elements.*

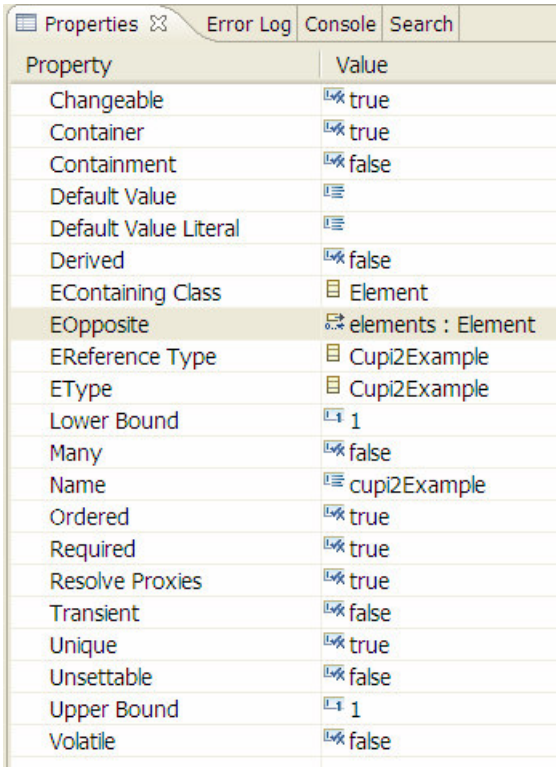
Lower Bound = 0
Upper Bound = 1
This means that the collection *elements* must have between cero (0) and undefined (-1) *Element.*

Container = False
Containment = True
This means that *Cupi2Example* is a container of *Elements.*

Note that the property *EOpposite* is empty. This will be setted after.

2. The second relationship created is the one between *Element* and *Cupi2Example*. This relationship is the "opposite" relationship to the first created relationship. For the creation

of this relationship an *EReference* is created for the *Element EClass* (Right click -> New Child -> …). Then the properties are defined:

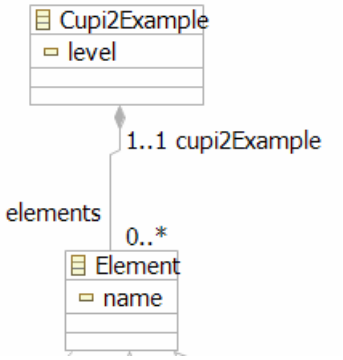


Name = cupi2Example
EType = Cupi2Example
This means that the *EClass Element* has a reference to a *Cupi2Example* element.

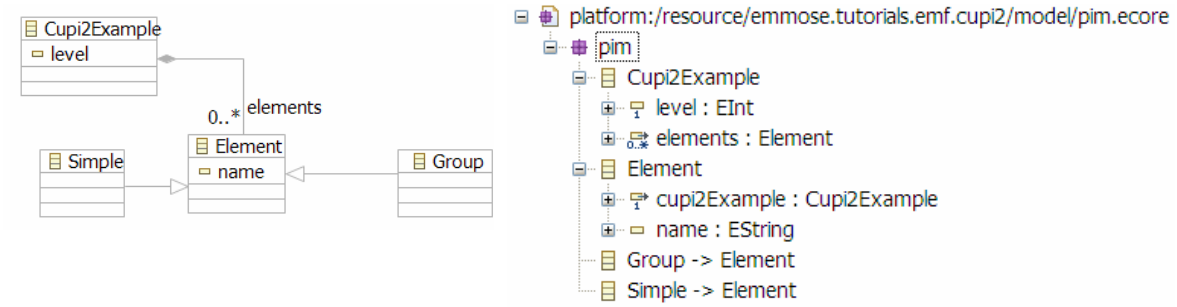
Lower Bound = 1
Upper Bound = 1
This means that must be created one and only one reference to a *Cupi2Example* element.

Container = True
Containment = False
EOpposite = elements : Element
This means that *Element* is contained in one *Cupi2Example.* Note that the property *EOpposite* makes reference to the already created *EReference elements* of the *Cupi2Example EClass* (step 2 of the current list). After this, the *EReference elements* automatically set the *EOpposite* property to *cupi2Example : Cupi2Example.*

3. The *EAttribute: level* is created for the *EClass: Elemente* (Right click -> New Child -> …). The *EType* is set to *EInt, Default Value Literals* is set to 1, and the *Lower* and *Upper Bound* are set to 1 (this means that the *EAttribute* is required). Figure at right presents the metamodel result.



4. The relationship of inheritance between *Group* and *Element,* and *Simple* and *Element* are created. Since *Group* and Simple are Sub-Types of *Element,* the property *Super Types* of *Group* and Simple are set to *Element.* The next figure shows the current state of the metamodel.



5. Finally please create the relationship from *Group* to *Element* for indicating that one *Group* can 'group' other elements. For the creation of this relationship an *EReference* is created for the *Element Group* (Right click -> New Child -> …). Then the properties are defined:

Property	Value
Changeable	true
Container	false
Containment	false
Default Value	
Default Value Literal	
Derived	false
EContaining Class	Group -> Element
EKeys	
EOpposite	
EReference Type	Element
EType	Element
Lower Bound	1
Many	false
Name	grouped
Ordered	true
Required	true
Resolve Proxies	true
Transient	false
Unique	true
Unsettable	false
Upper Bound	1
Volatile	false

Name = grouped
EType = Element

Lower Bound = 1
Upper Bound = 1

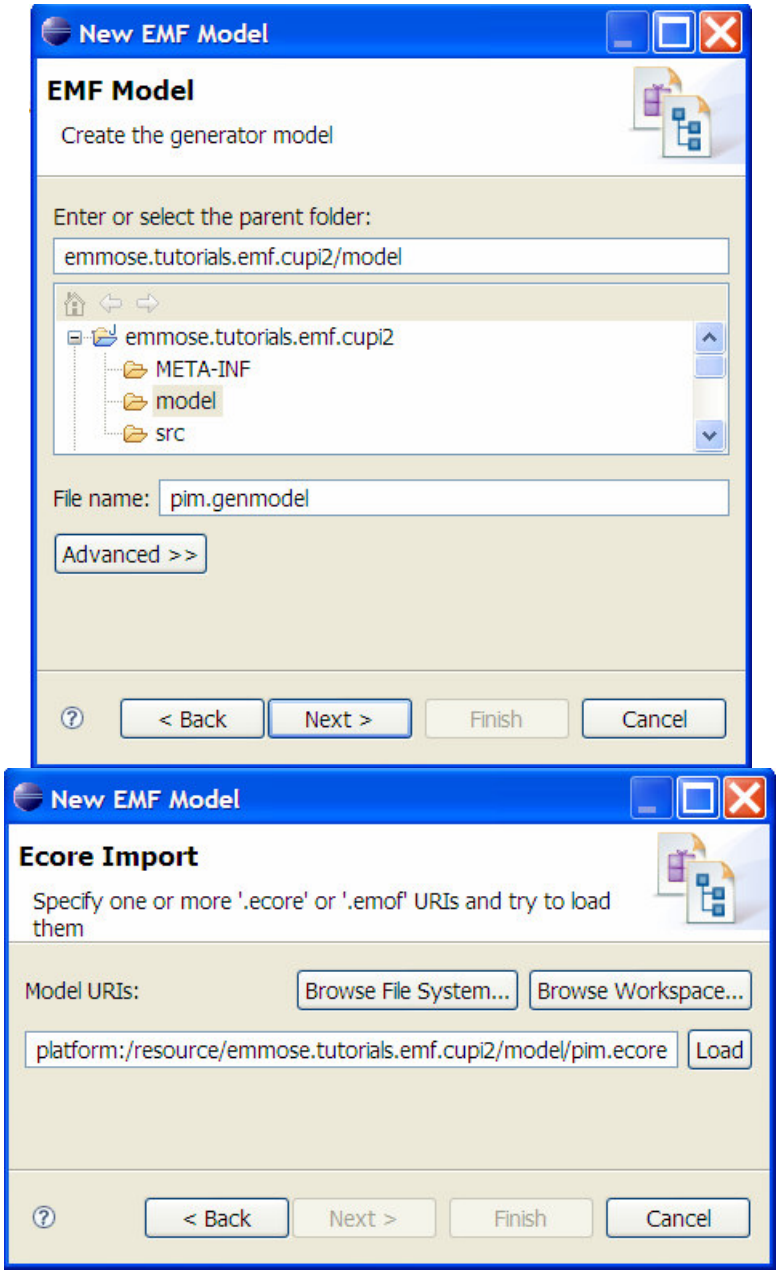
Container = False
Containment = False

6. The student must finish the creation of the metamodel, which is shown in the point two (2) of this document.

4.3 EMF Model Generation

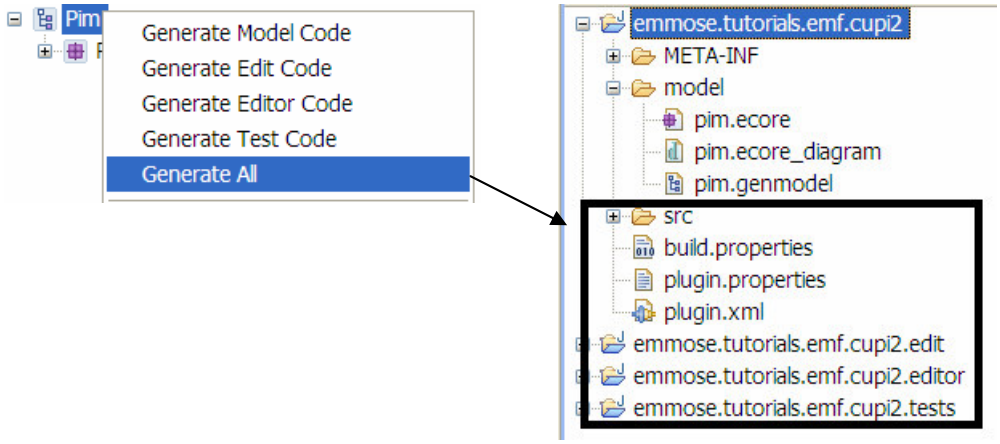
Once created the Ecore model it is possible to create the EMF model. The EMF model will be used for generating editors. These editors allow creating models that conform to the metamodel (Ecore Model).

Now, inside the folder named *model* you must create the EMF model (File -> new -> other -> Eclipse Modeling Framework -> EMF Model). Then, select as model importer the "*ecore model*" option, and finally select the recently created *model.ecore*.



5 Models Editor Creation

The model editors are automatically generated as Eclipse plug-ins by using an EMF facility. For this, open the recently created EMF model (*model.genmodel*) and press right click on it selecting the option *Generate All*.



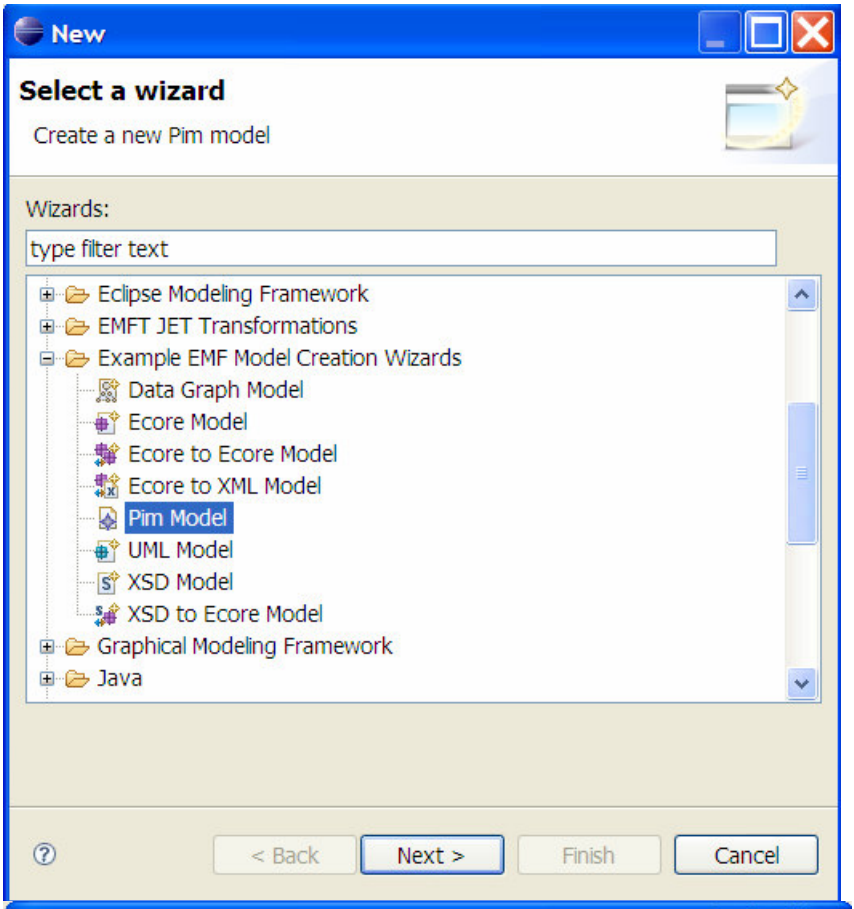
6 Executing a Model Editor

For creating models that conform to the created metamodels it is necessary to execute the created plug-in. For this press right click on the project and select Run As -> Eclipse Application; then, another Eclipse application is lunched.

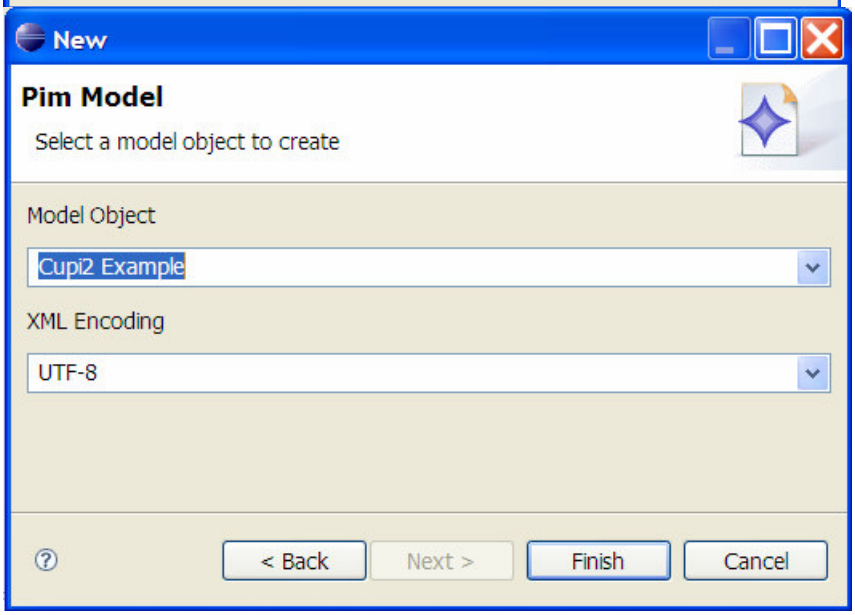
7 Creating a model that conform to the metamodel

For creating a new model that conforms to the metamodel by using the plug-ins we follow the next steps:

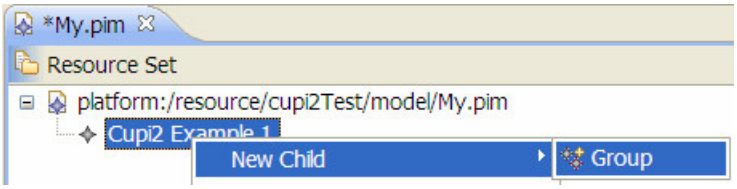
1. Create a new Project General Project (File ->New -> Project -> General -> Project).
2. Inside the new created project a model is created (right click on the project -> Other-> Example EMF Model Creation Wizards -> [*NameMetamodel*] Model)



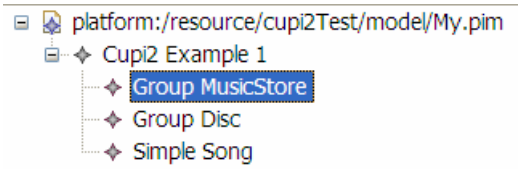
3. As *Model Object* you must select the defined main container *EClass*, in this case it is *Cupi2Example*.



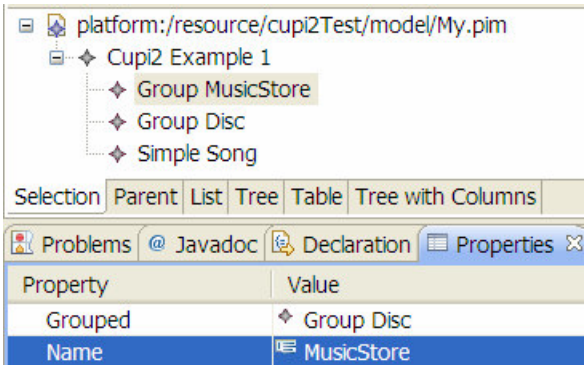
4. Finally the model is created and different model elements can be created.



5. Please create in the root *Cupi2Example1* two *Group* elements (named *MusicStore* and *Disc*) and one *Simple* element (named *Song).*



6. Finally, relate the elements: for the *MusicStore* set the *grouped* property to *Disc,* and for the *Disc* set the *grouped* property to *Son*.



7. After finishing the creation of the *Attribute* and *Cupi2Attribute EClasses*, complete this model adding *Attributes* to the *Disc* and *Song* elements.

8 Finishing the Cupi2 PIMM

The students will finish the creation of the metamodel and must create example models.