



# ÉCOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE ET D'ANALYSE DES SYSTÈMES

RAPPORT DE PROJET IDM DE 3 ÈME ANNÉE

---

## **Projet : Manipulation de modèles par programmation**

Soutenu par :

Taddist Afaf

Lejdaini Fatimazahrae

Sous la direction de :

M. ElHamlaoui Mahmoud

# Sommaire

<b>1</b>	<b>Modélisation</b>	<b>1</b>
1.1	Métamodèle Ecore de changement . . . . .	2
1.2	Grammaire du modèle de changement . . . . .	3
1.3	Quelques fonctions utilisées . . . . .	4
<b>2</b>	<b>Réalisation</b>	<b>6</b>
2.1	Exemple de test d'éditeur textuel . . . . .	6

# Table des figures

1.1	Ecore de changement . . . . .	2
1.2	Grammaire utilisée . . . . .	3
1.3	Suite de la grammaire utilisée . . . . .	3
1.4	Fonction name . . . . .	4
1.5	Fonction existEclass . . . . .	4
1.6	Fonction existEattribute . . . . .	5
2.1	SQL ecore cible . . . . .	6
2.2	SQL modèle cible . . . . .	7
2.3	Exemple de changement . . . . .	7
2.4	Après application de changement . . . . .	8



# Chapitre 1

## Modélisation

### 1.1 Métamodèle Ecore de changement

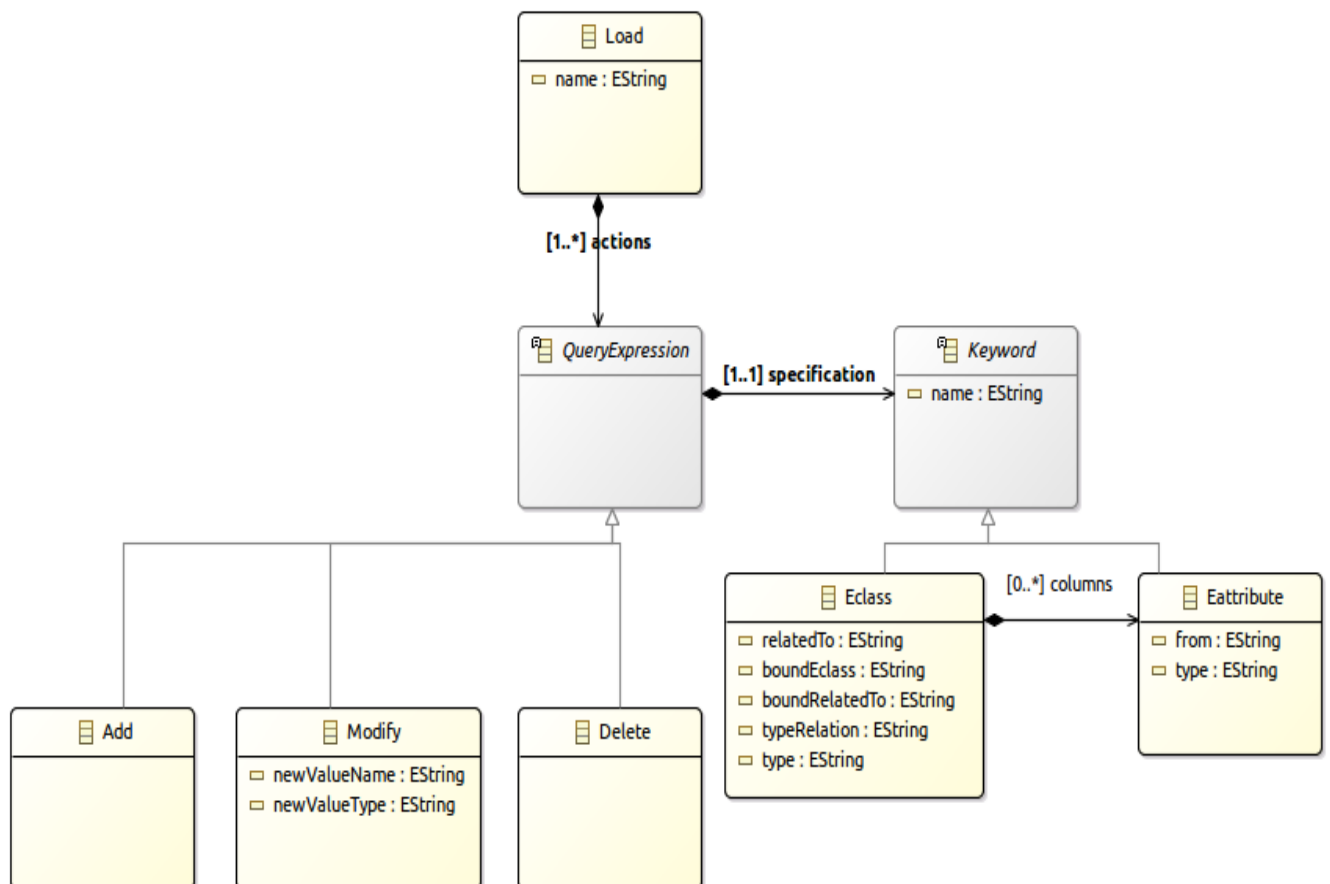


FIGURE 1.1 – Ecore de changement

## 1.2 Grammaire du modèle de changement

```
Load returns Load:
  'Load'
  name=EString
  'Do' '{' actions+=QueryExpression ( ";" actions+=QueryExpression)* '}' ;

QueryExpression returns QueryExpression:
  Add | Delete | Modify;

Keyword returns Keyword:
  Eclass | Eattribute;

EString returns ecore::EString:
  STRING | ID;

Add returns Add:
  'Add'
  specification=Keyword ;

Delete returns Delete:
  'Delete'
  specification=Keyword ;

Modify returns Modify:
  'Modify'
  specification=Keyword
  ('newValueName' newValueName=EString)?
  ('newValueType' newValueType=EString)? ;

Eclass returns Eclass:
  {Eclass}
  'Eclass'
  name=EString
  ( ':' (type=EString) )?
  (
    '['
      ('relatedTo' relatedTo=EString)?
      ('boundEclass' boundEclass=EString)?
      ('boundRelatedTo' boundRelatedTo=EString)?
      ('typeRelation' typeRelation=EString)?
    ']'
  )?
  ( '(' columns+=Eattribute ( "," columns+=Eattribute)* ')' )? ;

Eattribute returns Eattribute:
```

FIGURE 1.2 – Grammaire utilisée

```
Eattribute returns Eattribute:
  {Eattribute}
  'Eattribute'
  name=EString
  ('from' from=EString)?
  ('type' type=EString)?
  ;
```

FIGURE 1.3 – Suite de la grammaire utilisée

## 1.3 Quelques fonctions utilisées

Cette fonction permet de retourner le nom d'un objet passé comme argument.

```
public static String name(EObject object) {
    EClass metaClass = object.eClass();
    String nameMeta = metaClass.getName();
    return nameMeta;
}
```

FIGURE 1.4 – Fonction name

Cette fonction permet de tester si une classe existe dans le méta modèle cible.

```
public String ExistEclass(Hashtable h) {
    String result="";
    for(Iterator<EObject> ai= MResource.getAllContents(); ai.hasNext(); ) {
        EObject object = (EObject) ai.next();
        EClass eClass = object.eClass();

        if (eClass.getName().equals(h.get("type"))) {
            if (object.eContainer() != null) {
                Container = object.eContainer().eClass().getName();
            }
            EAttribute Name = (EAttribute) eClass.getEStructuralFeature("name");
            if (h.get("name").equals(object.eGet(Name)))
            {
                result="Exists";
                break;
            }
            else {
                result="NotExists";
            }
        }
    }
    return result;
}
```

FIGURE 1.5 – Fonction existEclass

Elle permet de tester si un attribut existe dans une classe donnée de le méta modèle cible.

```
public String ExistEattribute(Hashtable h) {
    String result="";
    EClass eclass= object.eClass();
    EList<EAttribute> eAttributes = eclass.getEAttributes();
    for (EAttribute eAttribute : eAttributes) {
        if ((h.get("name")).equals(eAttribute.getName())) {
            result="Exists";

            eAttChosen=eAttribute;
            break;
        }
        else {
            result="NotExists";
        }
    }
    return result;
}
```

FIGURE 1.6 – Fonction existEattribute



# Chapitre 2

## Réalisation

### 2.1 Exemple de test d'éditeur textuel

On essaye d'utiliser un modèle de SQL pour tester notre éditeur textuel .

```
<?xml version="1.0" encoding="UTF-8"?>
<ecore:EPackage xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance"
  xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" name="sql" nsURI="http://sql.fr" nsPrefix="sql">
  <eClassifiers xsi:type="ecore:EClass" name="Database">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name" eType="ecore:EDatatype http://
www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="cont" upperBound="-1" eType="#//Table"
      containment="true"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="TypeDB" eType="ecore:EDatatype http://
www.eclipse.org/emf/2002/Ecore#//EString"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Table">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name" eType="ecore:EDatatype http://
www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EReference" name="contC" upperBound="-1"
      eType="#//Column" containment="true"/>
  </eClassifiers>
  <eClassifiers xsi:type="ecore:EClass" name="Column">
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="name" eType="ecore:EDatatype http://
www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="type" eType="ecore:EDatatype http://
www.eclipse.org/emf/2002/Ecore#//EString"/>
    <eStructuralFeatures xsi:type="ecore:EAttribute" name="PrimaryKey" eType="ecore:EDatatype http://
www.eclipse.org/emf/2002/Ecore#//EBoolean"/>
  </eClassifiers>
</ecore:EPackage>
```

FIGURE 2.1 – SQL ecore cible

```

<?xml version="1.0" encoding="ASCII"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:sql="http://sql.fr">
  <sql:Database xmi:id="_8w-aQdCVeeloICn_Gc5pQ" name="DB1" TypeDB="postgresql">
    <cont name="T1">
      <contC name="C1" PrimaryKey="true"/>
    </cont>
    <cont name="T2">
      <contC name="C2" PrimaryKey="true"/>
      <contC name="C3" PrimaryKey="true"/>
    </cont>
    <cont name="T3"/>
  </sql:Database>
</xmi:XMI>

```

FIGURE 2.2 – SQL modèle cible

```

Load SQL Do
{
  //Add Databases
  Add Eclass EnsiasIngenieur :Database (Eattribute mysql type TypeDB) ;
  Add Eclass EnsiasDoctorat :Database;

  //Add Table + column
  Add Eclass Cours : Table [relatedTo EnsiasIngenieur] ;
  Add Eclass name : Column [relatedTo Cours] (Eattribute true type PrimaryKey);
  Add Eclass coefficient : Column [relatedTo Cours];

  //Add Table + column
  Add Eclass Filiere : Table [relatedTo EnsiasIngenieur] ;
  Add Eclass id : Column [relatedTo Filiere];
  Add Eclass name : Column [relatedTo Filiere] (Eattribute true type PrimaryKey)
  Add Eclass nbrEleve : Column [relatedTo Filiere];

  //Modify Name Database
  Modify Eclass EnsiasDoctorat :Database newValueName EnsiasDoctorant ;

  //Modify Value of Attribute
  Modify Eattribute PrimaryKey from id newValueName true;

  //Delete Database
  Delete Eclass EnsiasDoctorant ;

  //Delete Attribute
  Delete Eattribute PrimaryKey from name

}

```

FIGURE 2.3 – Exemple de changement

---

```

<?xml version="1.0" encoding="ASCII"?>
<xmi:XMI xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:sql="http://sql.fr">
  <sql:Database xmi:id="_8w-aQdCVEeeloICn_Gc5pQ" name="DB1" TypeDB="postgreSQL">
    <cont name="T1">
      <contC name="C1" PrimaryKey="true"/>
    </cont>
    <cont name="T2">
      <contC name="C2" PrimaryKey="true"/>
      <contC name="C3" PrimaryKey="true"/>
    </cont>
    <cont name="T3"/>
  </sql:Database>
  <sql:Database name="EnsiiasIngenieur" TypeDB="mysql">
    <cont name="Cours">
      <contC name="name"/>
      <contC name="coeficcient"/>
    </cont>
    <cont name="Filiere">
      <contC name="id" PrimaryKey="true"/>
      <contC name="name"/>
      <contC name="nbrEleve"/>
    </cont>
  </sql:Database>
</xmi:XMI>

```

FIGURE 2.4 – Après application de changement