

Modélisation et génération de code

Équipe Pédagogique Réseau Informatique@ UVC
2018

Table des matières



I - Objectifs	3
II - Modélisation orientée objet à l'aide de Mérisse et UML	4
1. Rappels sur Mérisse	4
2. Définition de quelques éléments de UML	5
3. Héritage à l'aide de Mérisse et UML	8
4. Exercice : Exercice 1	9
5. Exercice : Exercice 2	9
III - Génération de code	10
1. Code Java	10
2. Code C++	12
IV - Exercice : Exercice 1	14
V - Solutions des exercices	15



Objectifs

A la fin de cette leçon, vous serez capable :

- Apprendre à modéliser un objet à l'aide de Merise et Uml
- Générer un code en JAVA et C++

Modélisation orientée objet à l'aide de Mérise et UML



Objectifs

Apprendre à modéliser un objet à l'aide de Merise et Uml

Il ne s'agit ici de faire un cours de mérise ou d'uml. Cependant, les éléments de mérise et uml qui permettent de modéliser un objet seront présentés.

1. Rappels sur Mérise

Définition : Mérise

Mérise est une méthode d'analyse, de conception et de réalisation de systèmes d'informations

Modèle conceptuel de données (MCD)

Un MCD est constitué des :

- noms des entités
- noms des associations
 - un à plusieurs
 - un à un
 - plusieurs à plusieurs

Exemple : Exemple d'un MCD

L'exemple ci dessous représente un modèle conceptuel de données d'une PME.

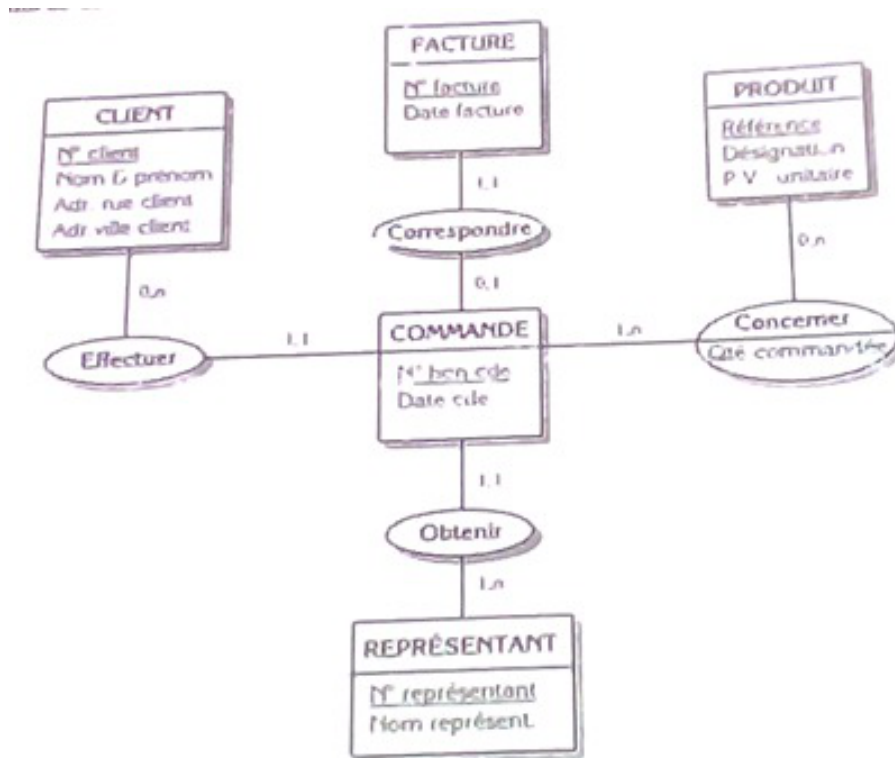


Figure 1 - MCD

Dans cette figure,

- CLIENT, FACTURE, PRODUIT, COMMANDE, REPRESENTANT sont des les noms des entités.
- Effectuer, Correspondre, Obtenir Concerner sont les noms des associations.

Chaque entité est composé d'un ensemble d'éléments appelé propriétés. L'entité CLIENT a pour propriétés les éléments suivants :

- N° client
- Nom et Prénoms
- Adr. rue client
- Adr. ville client

Chaque entité d'un MCD peut avoir plusieurs propriétés qui lui sont spécifiques.

Les entités sont reliées entre elles à travers les associations et de leurs cardinalités.

La notion d'objet

Avec Mérise, on peut dire que "les *entités* représentent les *objets* et les *propriétés* des entités sont les *attributs* de ces objets". Cependant, avec Mérise les méthodes qui manipulent les attributs n'apparaissent pas dans le modèle conceptuel de données.

2. Définition de quelques éléments de UML

Qu'est ce que UML ?

UML (Unified Modeling Language) est un langage de modélisation unifié.

En fait, avec UML, vous pouvez modéliser toutes les étapes du développement d'une application informatique, de sa conception à la mise en route, grâce à des diagrammes.

Parmi les diagrammes proposés par uml, nous allons nous intéresser au diagramme de classes. En effet, c'est de celui-là que nous allons nous servir. Il permet de modéliser des classes ainsi que les

interactions entre elles.

Un diagramme de classes est par exemple ce qu'on appelle "MCD en Mérisse" sauf que c'est pas totalement vrai. L'une des différences est que ce qui est appelé entité avec Mérisse représente une classe avec UML.

Modéliser un objet en UML

En UML, une entité ou une classe est modélisée par :

- le nom de la classe
- les propriétés ou attributs de la classe
- les méthodes de la classe

Exemple : Une entité ou une classe

En UML, une entité ou une classe CLIENT peut être représentée par la figure 2.

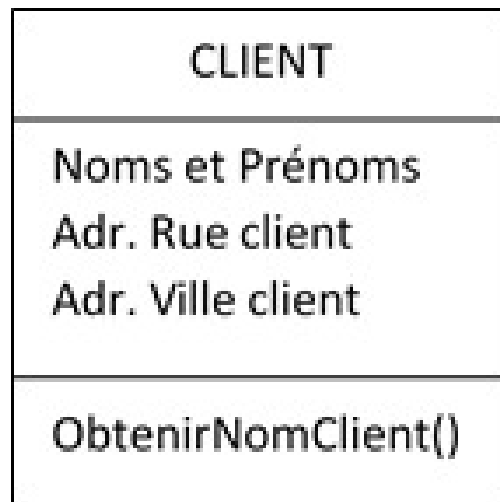


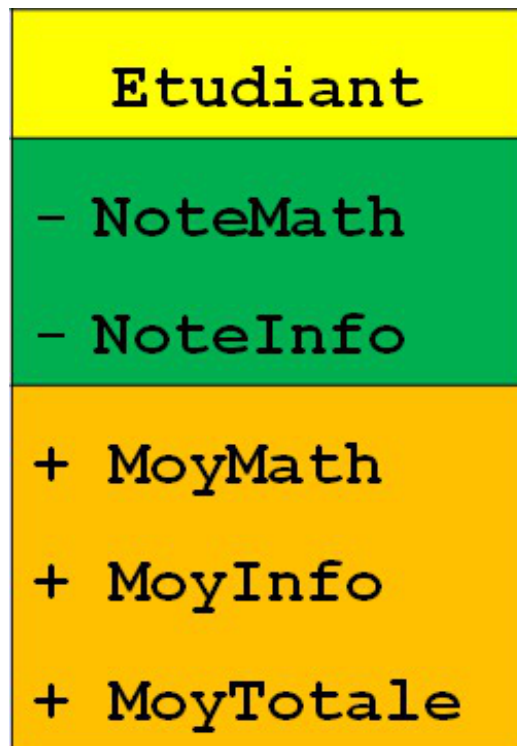
Figure 2 - Une classe

Dans cet exemple,

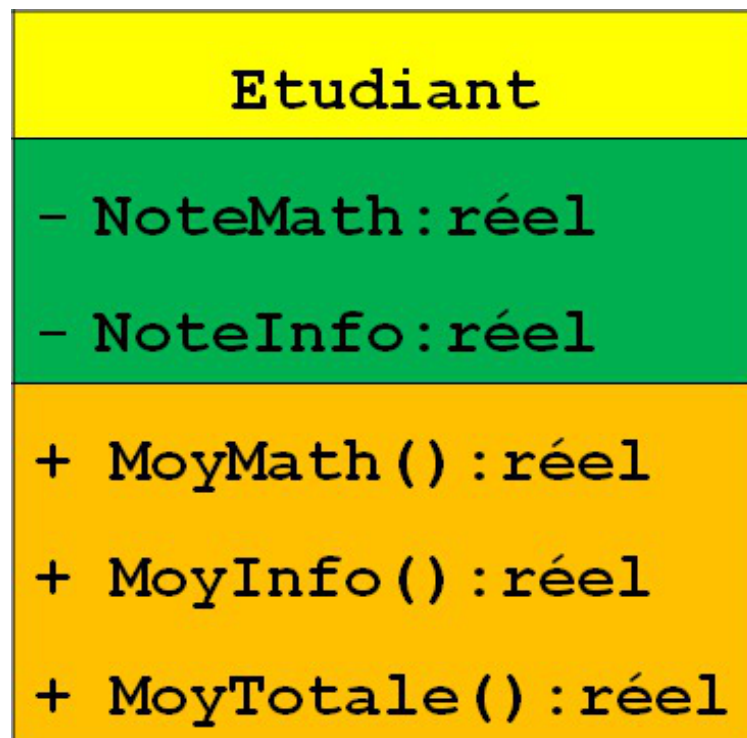
- CLIENT représente le nom de la classe ;
- Noms et Prénoms, Adr. Rue client, Adr. Ville client représentent les attributs de la classe ;
- ObtenirNomClient() représente une méthode de la classe.

Exemple : Exemple de classe plus complète

Ici, nous allons modéliser une classe Etudiant avec toutes ses caractéristiques comme UML le fait. La figure 3 est une représentation un peu proche de la figure 2.

*Figure 3 - Classe Etudiant*

Ici, Etudiant est le nom de la classe ; NoteMath et NoteInfo sont les attributs de la classe ; et enfin MoyMath, MoyInfo et MoyTotale sont les méthodes de la classe. La figure 4 ci-dessous est plus complète que celle de figure 3.



Ici, chaque élément de la classe représenté a son type qui est spécifié. en plus des types, le signe "-" spécifie que les attributs sont privés et le signe "+" spécifie que les méthodes sont publiques.

Remarque : Modélisation d'une classe

Jusqu'ici, la modélisation d'une classe en UML montre qu'il y a un lien entre l'entité du MCD et la classe avec UML.

Lien ou association entre les classes

Comme Mérisse, UML aussi permet d'associer plusieurs classes à travers un lien ou une association. Dans la figure 5, il existe une association entre l'entité Etudiant et l'entité Classe.

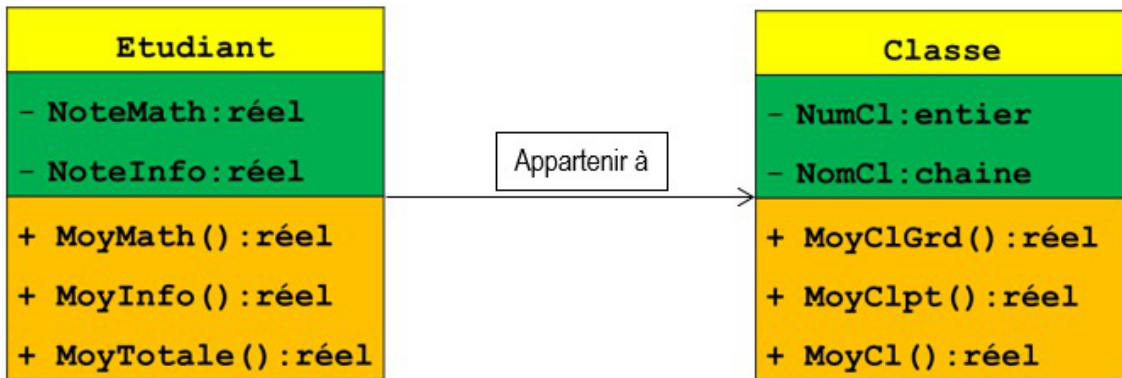


Figure 5 - association entre deux classes

3. Héritage à l'aide de Mérisse et UML

Représentation de l'héritage avec Mérisse

La figure 6 ci-dessous présente l'héritage selon la méthode Mérisse.

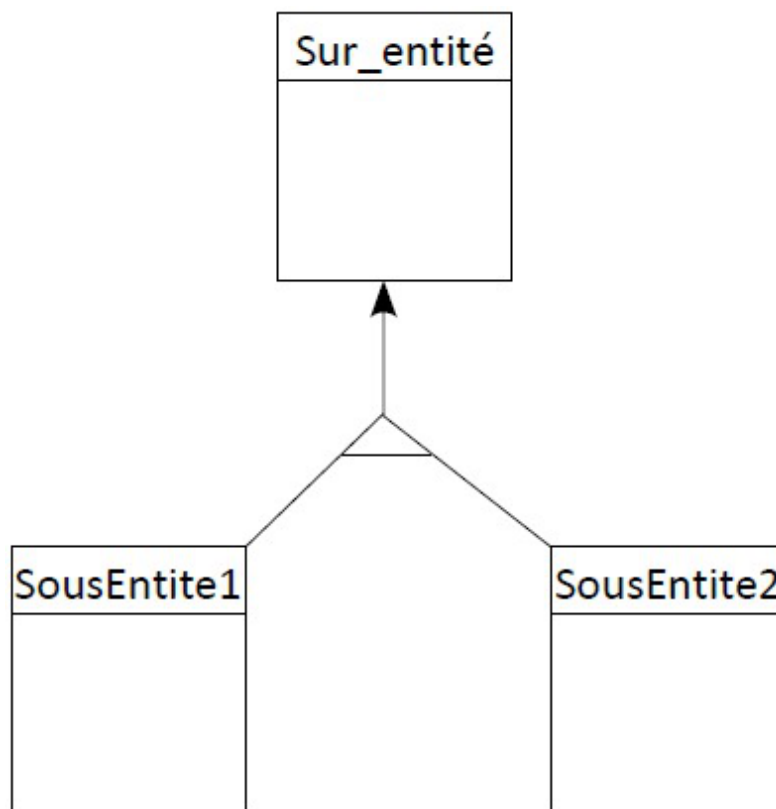


Figure 6 - héritage avec mérisse

Ici, Sur_entité représente une *superclasse* et Sousentité1, Sousentité2 représente des *sousclasses*.

Représentation héritage selon UML

La figure 7 ci-dessous présente l'héritage selon le langage UML.

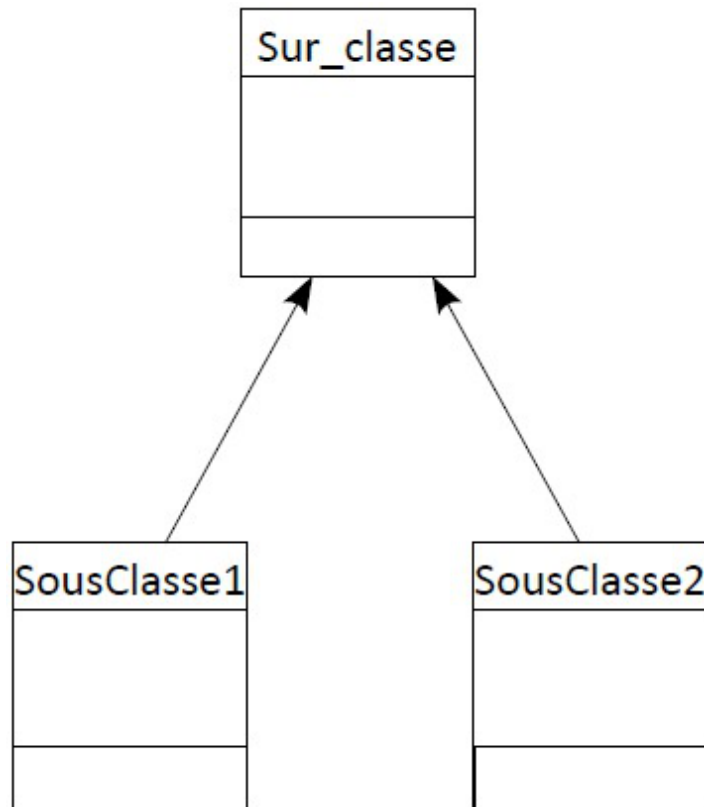


Figure 7 - héritage avec UML

Ici, Sur_Classe représente une *superclasse* et SousClasse1, SousClasse2 représente des *sousclasses*.

4. Exercice : Exercice 1

[Solution n°1 p 15]

Soit la figure suivante : 14.png. Décrivons cette classe.

Dans la classe les deux NoteMath et NoteInfo sont de type et sont .

Les trois de calcul de moyennes sont .

Les entités des MCD de la méthode Mérise représentent les et les propriétés des entités sont les de ces objets"

5. Exercice : Exercice 2

[Solution n°2 p 15]

En UML, une entité ou une classe est modélisée par :

- ☐ le nom de la classe
- ☐ les propriétés ou attributs de la classe
- ☐ les méthodes de la classe

Génération de code



Objectifs

Générer un code en JAVA et C++

1. Code Java

Génération du code Java

Supposons qu'on dispose des entités définies par la figure 8 ci-dessous

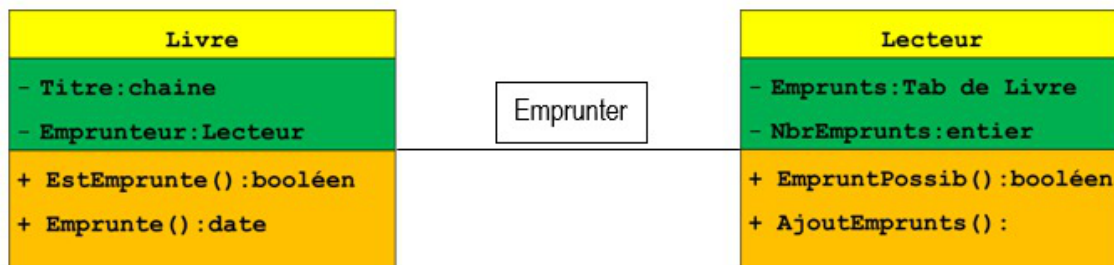


Figure 8 - association de l'entité Livre et Lecteur


Cette représentation est la représentation uml. Cette représentation définit deux classes :

- La classe Livre ;
- La classe Lecteur

La classe Livre est définie par :

- ses attributs Titre et Emprunteur ;
- ses méthodes EstEmprunte() et Emprunte().

Le code Java de cette classe est présenté comme suit :



```

public class Livre {
    private
        Lecteur Emprunteur;
        String Titre;

    public
        boolean EstEmprunte() {
            return true;
        };
        Date Emprunte() {
            return
        };
}

```


Figure 9 - code Java de la classe Livre

Vous aurez plus d'explication lorsque le cours de Java commencera.

La classe Lecteur est définie par :

- ses attributs Emprunts et NbrEmprunts ;
- ses méthodes AjoutEmprunts() et EmpruntPossib().

Le code Java de cette classe Lecteur est présenté comme suit :



```

public class Lecteur {
    private
        int NbrEmprunts;
        Livre [] Emprunts;

    public
        boolean EmpruntPossib() {
            return true;
        }
        void AjoutEmprunts() {

        }
}

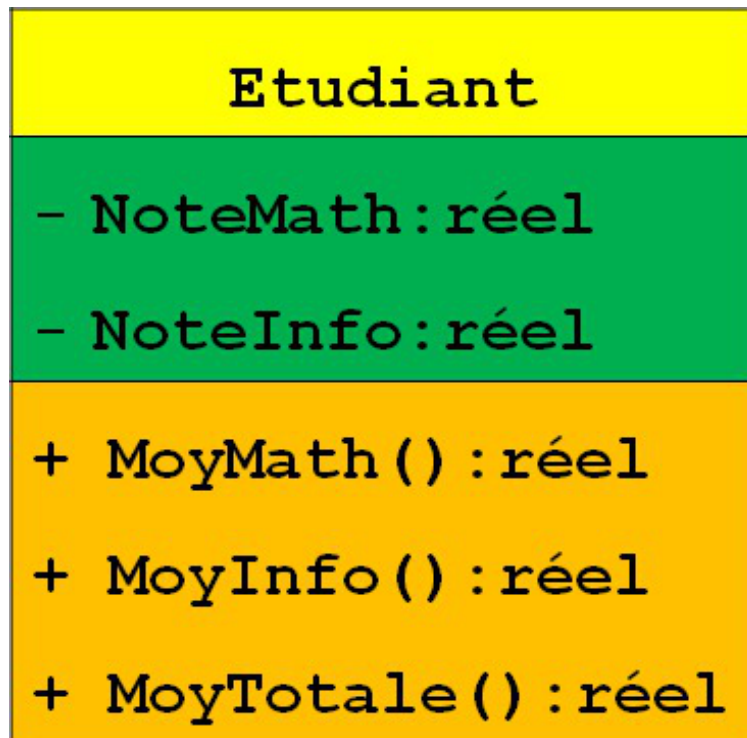
```

Figure 10 - Code Java de la classe Lecteur

2. Code C++

Génération de code C++

Supposons qu'on dispose des entités définies par la figure ci-dessous



C'est la même figure que la figure 4. Le code C++ est :

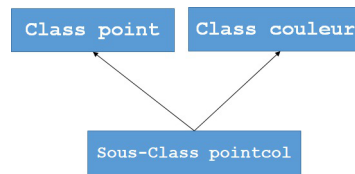
```

class Etudiant{
    private:
        float NoteMath;
        float NoteInfo;
    public:
        float MoyInfo();
        float MoyMath();
        float MoyTotale();
};
  
```

Figure 12 - code c++ de la classe Etudiant

Code héritage

Soit l'image suivante :



Le code C++ de l'héritage est :

```
class pointcol : public point, public couleur {  
} ;
```

Exercice : Exercice 1



[Solution n°3 p 15]

Le code de l'héritage simple suivant :15.png est représenté en C++ par :

```
class [ ] : public [ ] { };  
class [ ] : public [ ] { };  
class [ ] : public [ ] { };
```



Solutions des exercices

> Solution n° 1

Exercice p. 9

Soit la figure suivante : *14.png*. Décrivons cette classe.

Dans la classe `Eleve` les deux `attributs` `NoteMath` et `NoteInfo` sont de type `réel` et sont `privés`.

Les trois `méthodes` de calcul de moyennes sont `publiques`.

Les entités des MCD de la méthode Mérise représentent les `objets` et les propriétés des entités sont les `attributs` de ces objets"

> Solution n°2

Exercice p. 9

- ☒ le nom de la classe
- ☒ les propriétés ou attributs de la classe
- ☒ les méthodes de la classe

> Solution n° 3

Exercice p. 14

Le code de l'héritage simple suivant : *15.png* est représenté en C++ par :

```
class Sous-Classe1 : public superclasse { };  
class Sous-Classe2 : public superclasse { };  
class Sous-Classe3 : public superclasse { };
```