

DIAGRAMME DE CLASSES

Formateur : Mahdi KELLOUCH

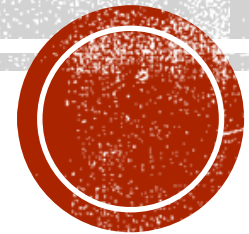


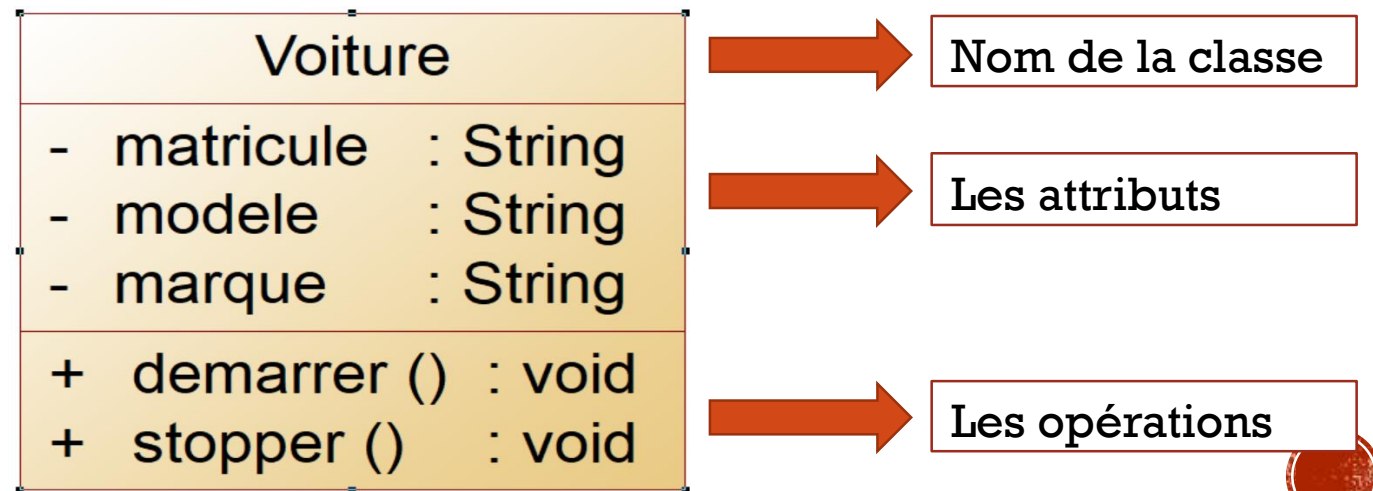
DIAGRAMME DE CLASSES

- Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation.
- Alors que le diagramme de cas d'utilisation montre un système du point de vue des acteurs, le diagramme de classes en montre la structure interne.
- Il permet de fournir une représentation abstraite des objets du système qui vont interagir pour réaliser les cas d'utilisation.



CLASSE

- Une classe est une description d'un ensemble d'objets ayant une sémantique, des attributs, des méthodes et des relations en commun.
- Un objet est une instance d'une classe.
- Une classe est composé de trois sections :
 - Nom de classe
 - Les attributs
 - Les opérations



LES ATTRIBUTS

- Les attributs définissent des informations qu'une classe ou un objet doivent connaître.
- Ils représentent les données encapsulées dans les objets de cette classe.
- Chacune de ces informations est définie par un nom, un type de données, une visibilité et peut être initialisée.
- Le nom de l'attribut doit être unique dans la classe.



LES ATTRIBUTS

- La syntaxe de la déclaration d'un attribut est la suivante :

```
<visibilité>  [/]  <nom_attribut>  ['['<multiplicité>']']  :  <type>  [  =  
<valeur_par_défaut> ]
```

[] : champs optionnel.

< > : champs obligatoire.



LES ATTRIBUTS

- Visibilité : il existe 4 niveaux de visibilité :
 - **Public ou +** → Accessible par n'importe quelle autre classe.
 - **Private ou -** → inaccessible par toute autre classe.
 - **Protected ou #** → Accessible uniquement par les classes dérivées.
 - **Sans clé ou ~** → Accessible uniquement dans le package de la classe.

Classe
+Attribut public - Attribut privé #Attribut protégé
+Opération public() -Opération privée() #Opération protégée()



LES ATTRIBUTS

- **Nom attribut** : identificateur de l'attribut, unique au sein de la classe.
- **Multiplicité** : Nombre de valeurs stockées dans l'attributs; exemple : tableau
Parents[1..2]: Personne
- **Valeur par défaut** : valeur prise par l'attribut lors de l'instanciation de la classe.
- **Type attribut** : peut être un nom de classe, un nom d'interface ou un type de donnée prédéfini.

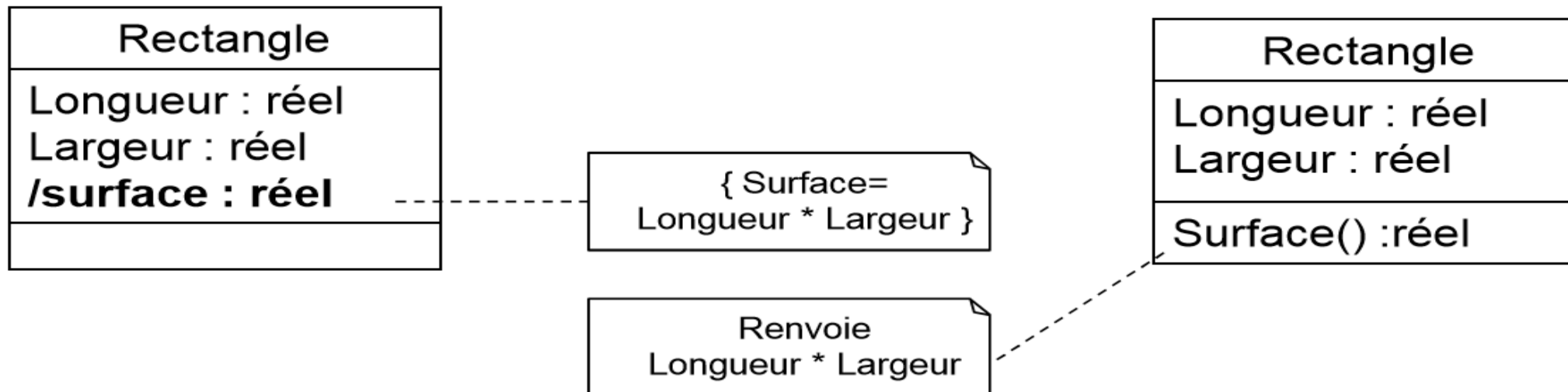
Télévision
+Marque : chaîne
Télétexte : Booléen=vrai
Couleur : enum { gris,noir }
ChaîneTv: canal

Canal
Fréquence: réel



ATTRIBUT DÉRIVÉ

- Les attributs dérivés peuvent être calculés à partir d'autres attributs et de formules de calcul.
- Les attributs dérivés sont symbolisés par l'ajout d'un « / » devant leur nom.
- Les attributs dérivés peuvent être traduits par une opération.



ATTRIBUT DE LA CLASSE

- Attribut qui garde une valeur unique et partagée par toutes les instances de la classe.
 - Les instances ont accès à cet attribut, mais n'en possèdent pas une copie.
 - Un attribut de classe n'est pas une propriété d'une instance, mais une propriété de la classe.
 - L'accès à cet attribut ne nécessite pas l'existence d'une instance.

triangle
Angle[1..3]: réel
<u>Pi : réel= 3.14</u>
/surface : réel



LES OPÉRATIONS

- Les opérations permettent de définir le comportement des instances de la classe.
- La syntaxe de déclaration d'une operation est :

`<visibilité> <nom_méthode> ([<paramètre_1>, ... , <paramètre_N>]) : [<type_renvoyé>]`

- **Nom_opération** : identificateur de l'opération, unique au sein de la classe.
- **Type_retourné**: la valeur retournée par l'opération; si omis l'opération ne retourne aucune valeur.

Remarque : Une opération peut apparaître plusieurs fois avec des paramètres différents, on dit que cette opération est surchargée.



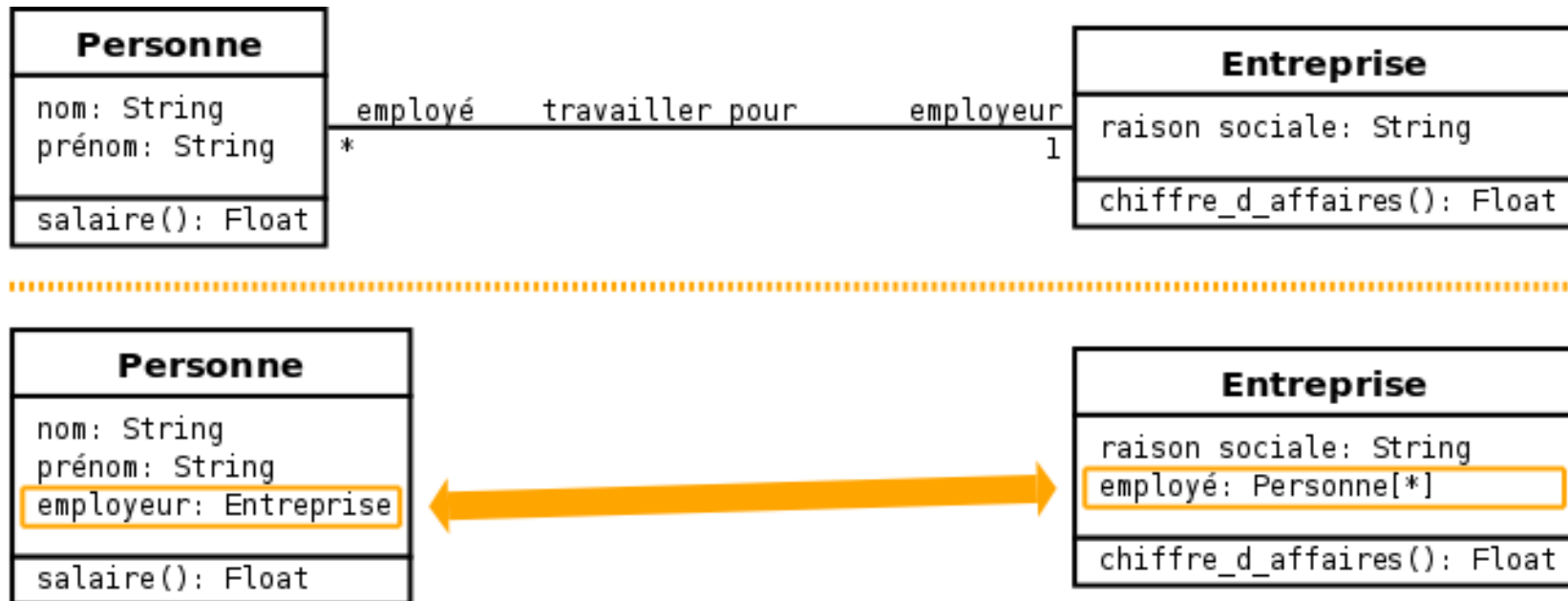
RELATIONS ENTRE LES CLASSES

- Les relations les plus utilisées sont :
 - Association
 - Agrégation
 - Composition
 - Héritage



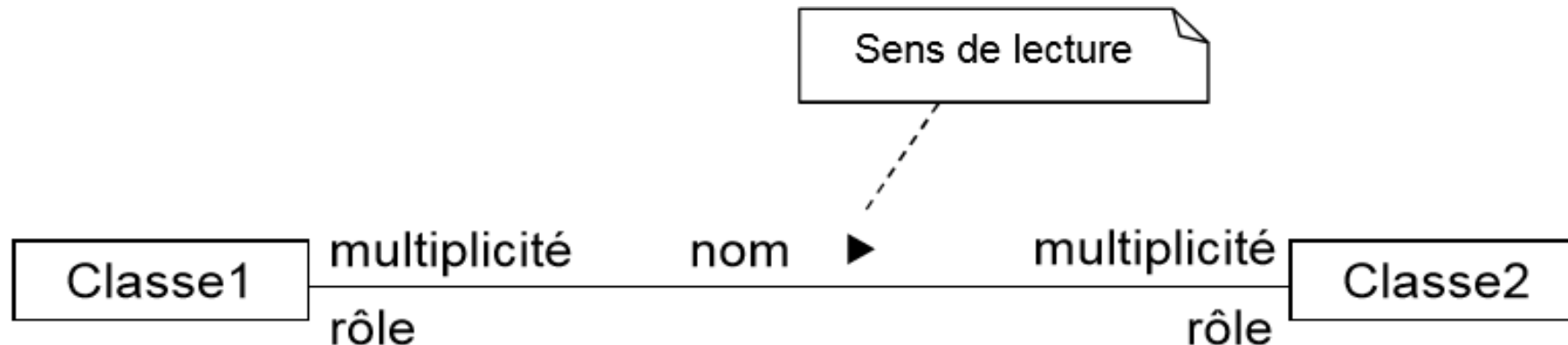
RELATIONS ENTRE LES CLASSES - ASSOCIATION

- Une association est une relation entre deux classes (association binaire) ou plus (association n-aire).
- Une association indique donc qu'il peut y avoir des liens entre des instances des classes associées.



RELATIONS ENTRE LES CLASSES - ASSOCIATION

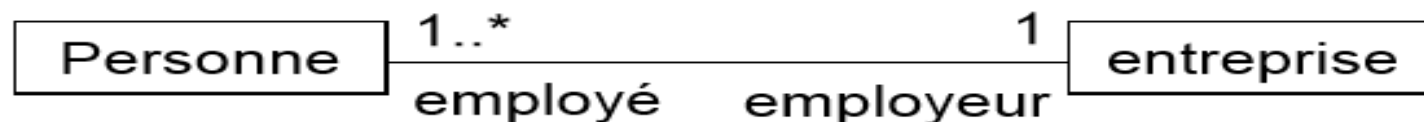
- une relation sémantique entre les objets d'une classe.
 - Représentée graphiquement par un trait plein entre les classes associées.
 - Complétée par un nom qui correspond souvent à un verbe à l'infinitif, avec une précision du sens de lecture en cas d'ambiguïté.
 - Chaque extrémité de l'association indique le rôle de la classe dans la relation et précise le nombre d'objets de la classe qui interviennent dans l'association.



ASSOCIATION – MULTIPLICITÉ

- **Multiplicité:** indique nombre d'objets de la classe apparaissant à cette extrémité pouvant s'associer à un seul et unique objet de la classe apparaissant à l'autre extrémité

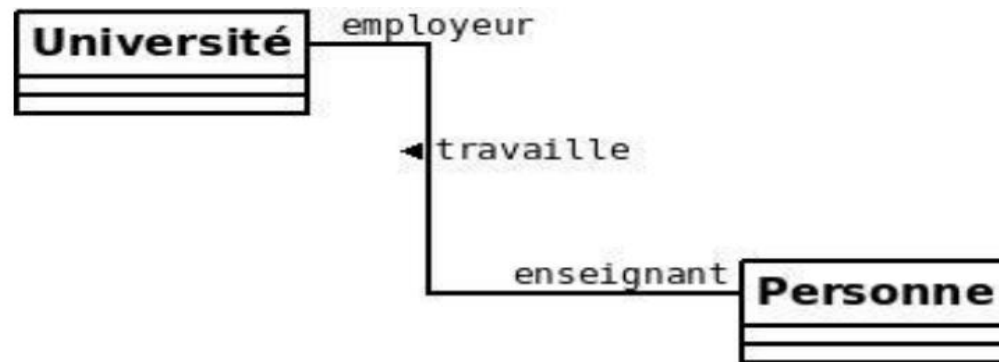
Multiplicité	Signification
1	un et un seul (multiplicité par défaut)
0..1	zéro ou un
N	exactement N
M..N	de M à N
*	zéro ou plus, c-à-d. de 0 à $+\infty$
1..*	un ou plus, c-à-d. de 1 à $+\infty$



ASSOCIATION - ROLE

- Il est possible de préciser le rôle joué par une ou plusieurs des classes composant une association afin d'en faciliter la compréhension. On ajoute pour cela ce rôle à côté de la classe concernée.

- Exemple 1 :

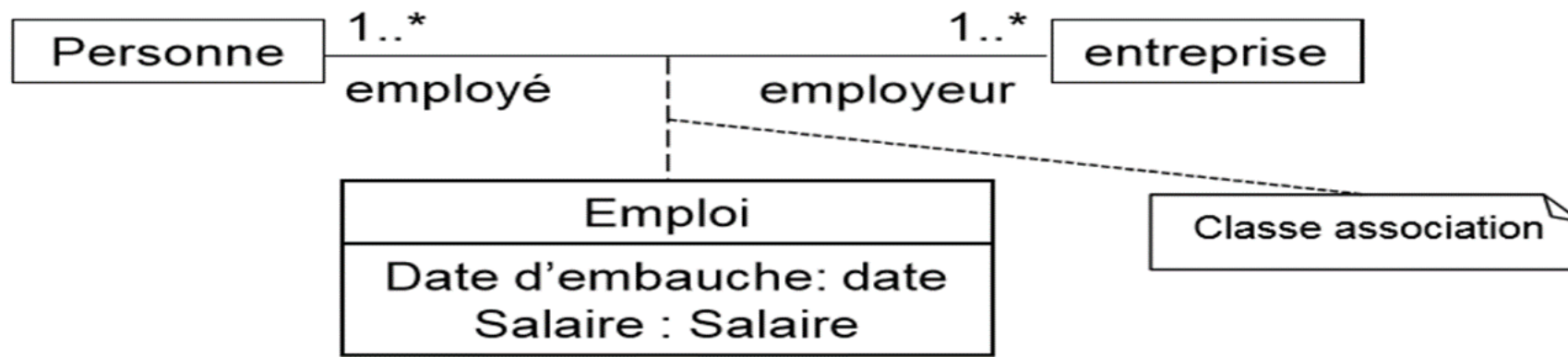


- Exemple 2:



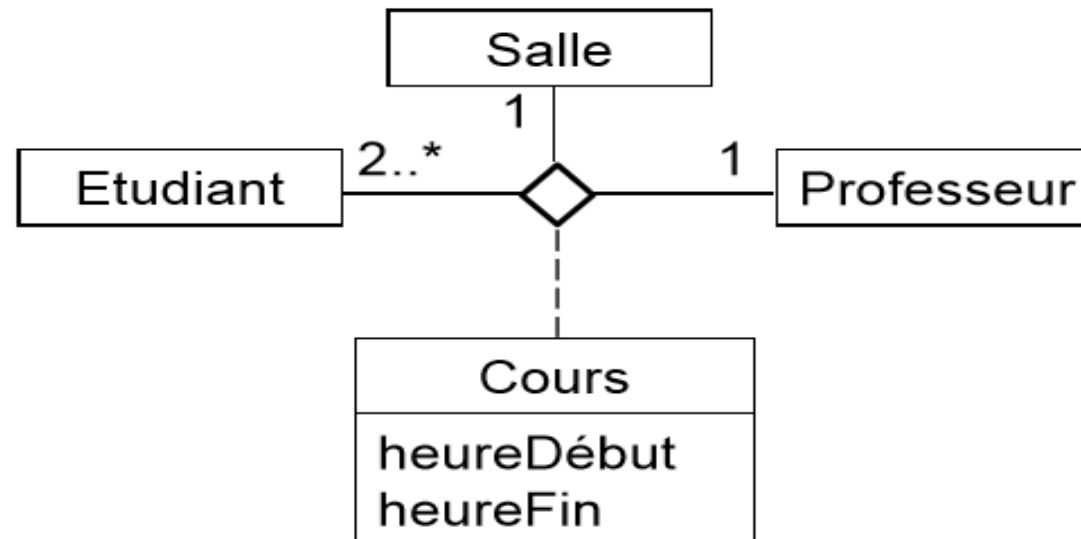
CLASSE ASSOCIATION

- Une association peut avoir ses propres propriétés, qui ne sont disponibles dans aucune des classes liée.
- Seules les classes peuvent avoir des propriétés, cette association devient alors une classe appelée «classe association».
- À partir du moment où elle est définie, elle est considérée comme toutes les autres classes du modèle.



ASSOCIATION N-AIRE

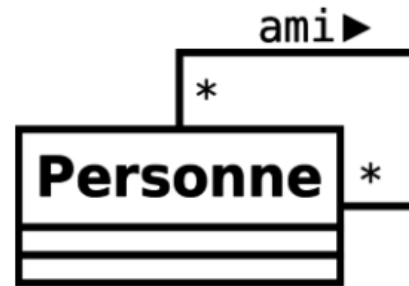
- Une association n-aire lie plus de deux classes. Les traits pleins qui viennent de toutes les classes associées convergent vers un losange central pouvant éventuellement accueillir une classe association.
- La multiplicité apparaissant sur le lien de chaque classe s'applique sur une instance du losange.



ASSOCIATION REFLEXIVE

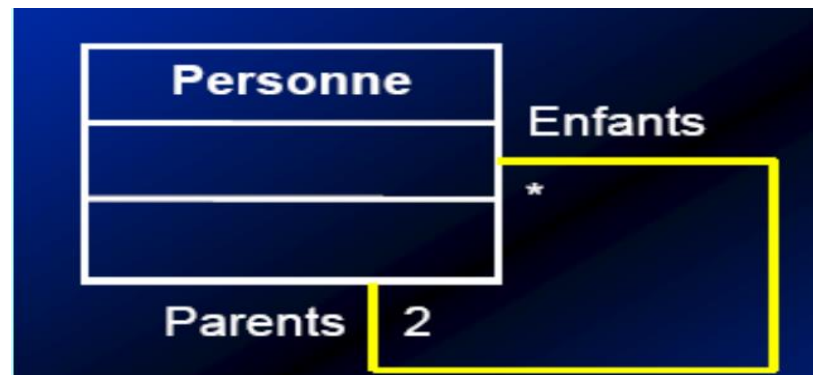
- Une association réflexive est une association qui associe une classe avec elle-même.

- Exemple 1 :



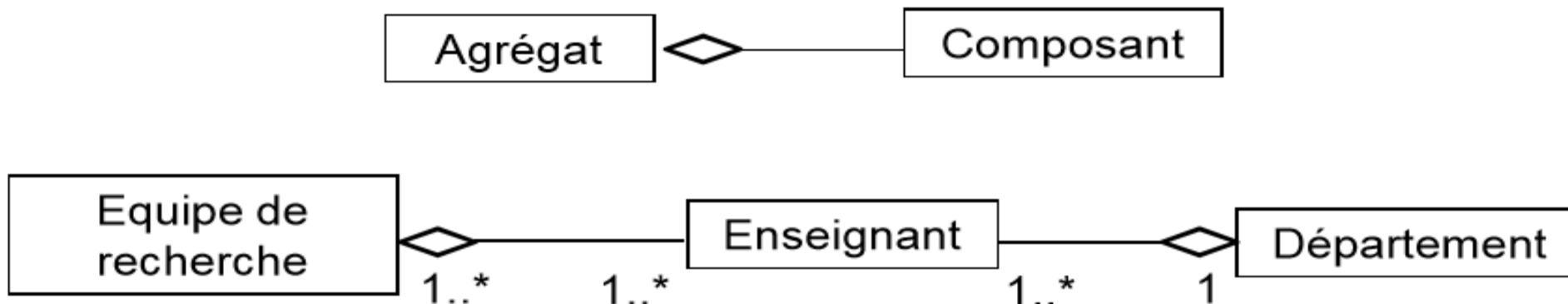
Association réflexive « ami »

- Exemple 2 :



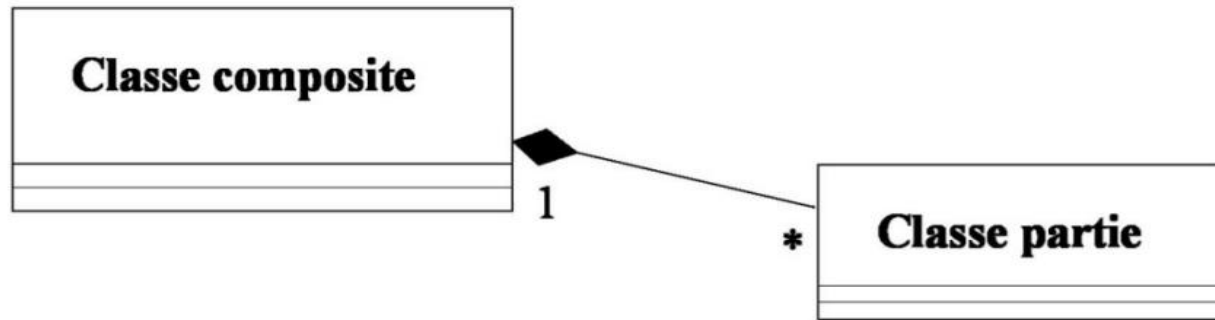
AGRÉGATION

- Association non symétrique d'inclusion d'un élément dans un ensemble.
 - Dans un diagramme, une agrégation se distingue d'une association par l'ajout d'un losange vide du côté de l'agrégat.
 - Une classe peut être liée à plusieurs classes agrégats.
 - Destruction de l'agrégat n'implique pas destruction des classes associées.



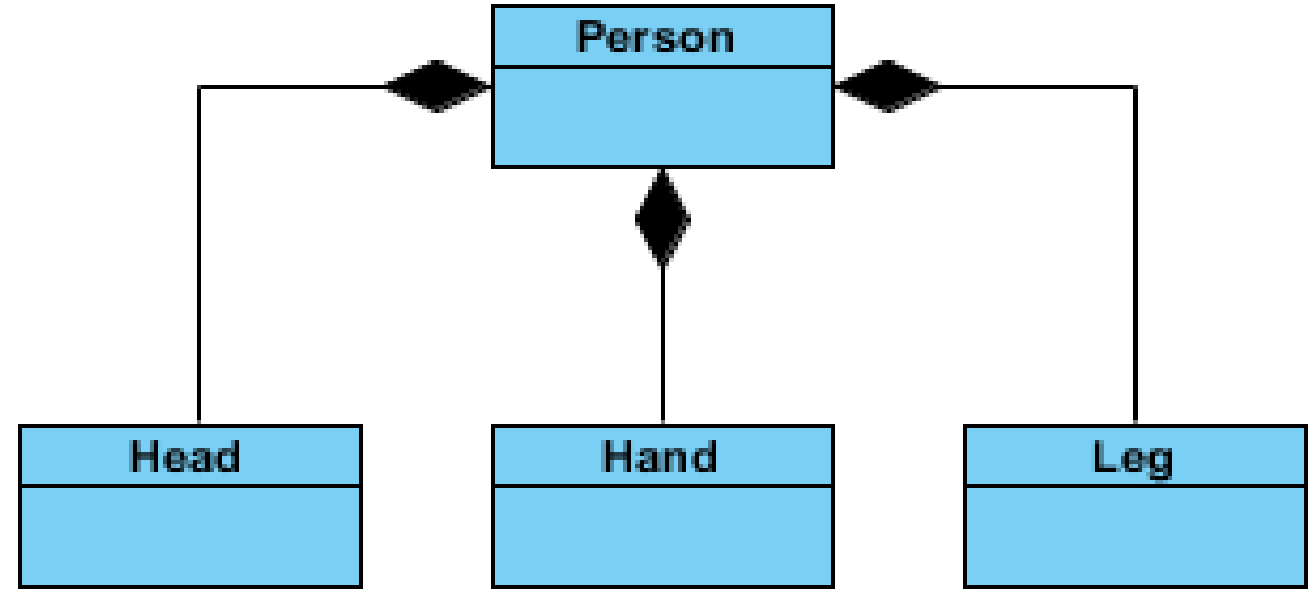
COMPOSITION

- Un lien de composition symbolise l'existence d'une agrégation particulière, dite 'forte', entre deux entités (classes).
- Dans une relation de type composition, la destruction de l'objet composite implique la destruction de ses composants.

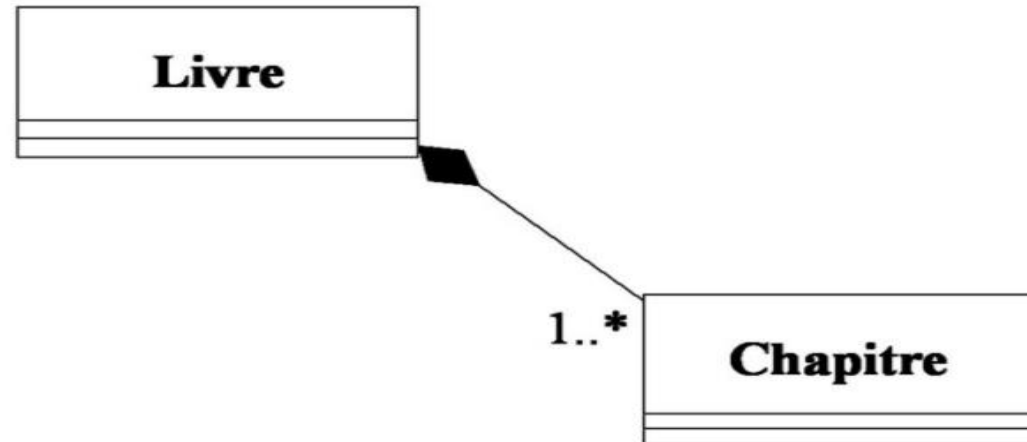


COMPOSITION

- Exemple 1 :

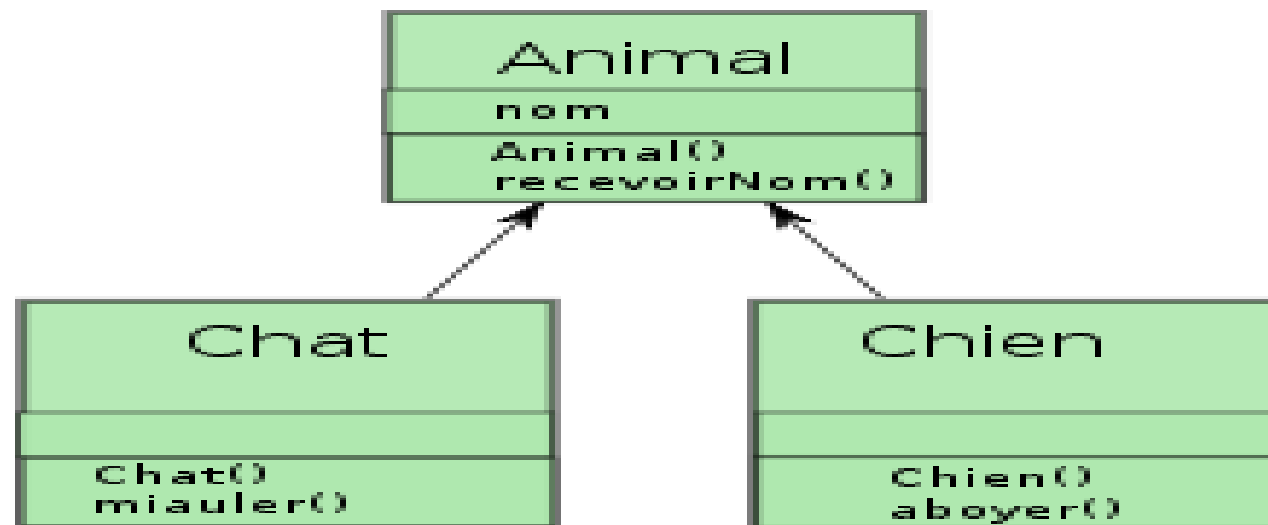


- Exemple 2 :

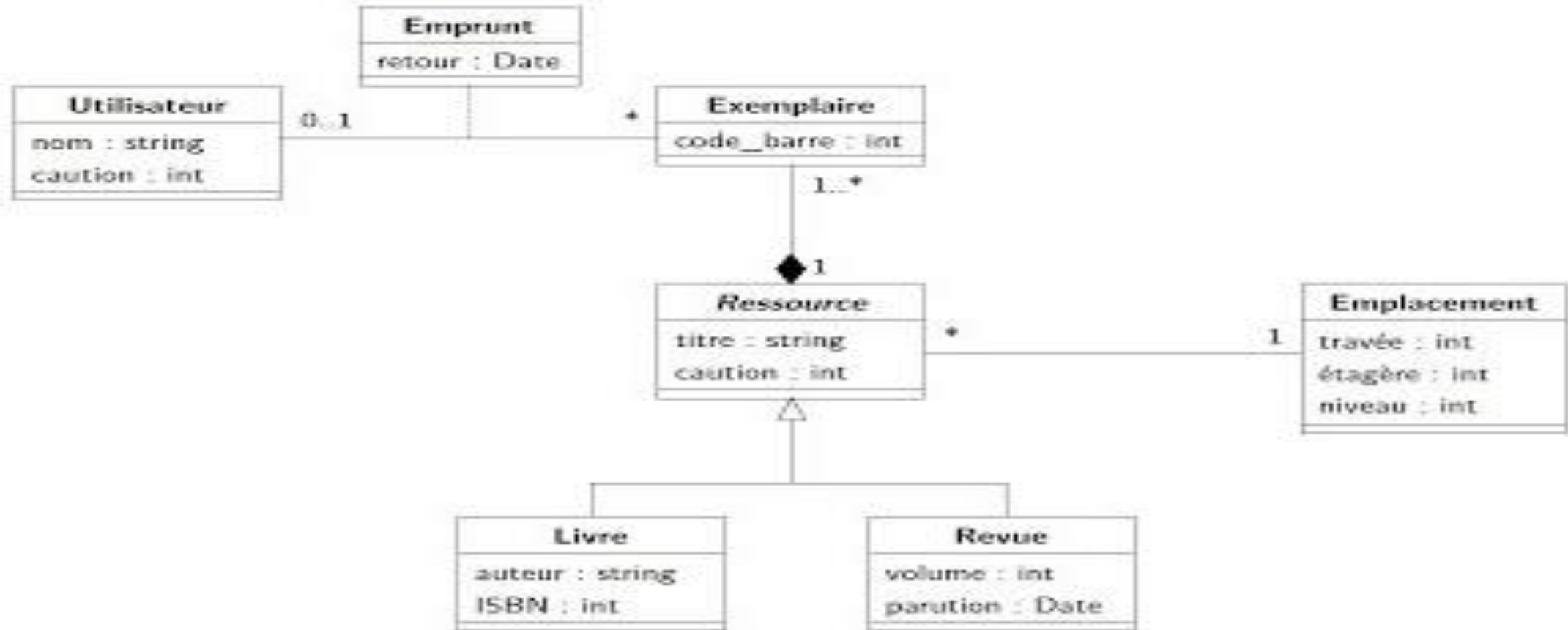


HÉRITAGE

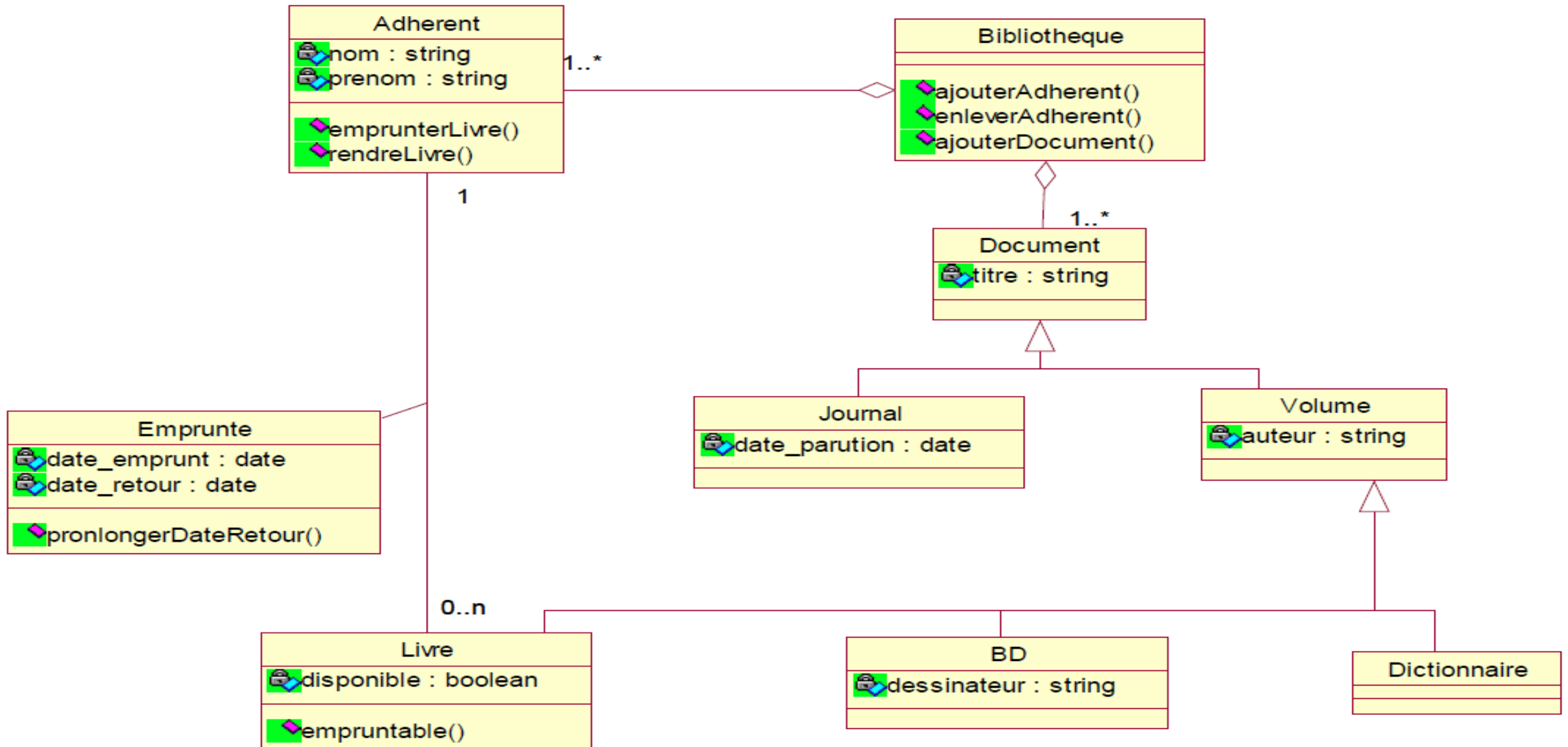
- L'héritage établit une relation de généralisation-spécialisation qui permet d'hériter dans la déclaration d'une nouvelle classe (appelée classe dérivée, classe fille, classe enfant ou sous-classe) des caractéristiques (propriétés et méthodes) de la déclaration d'une autre classe (appelée classe de base, classe mère, classe parent ou super-classe).



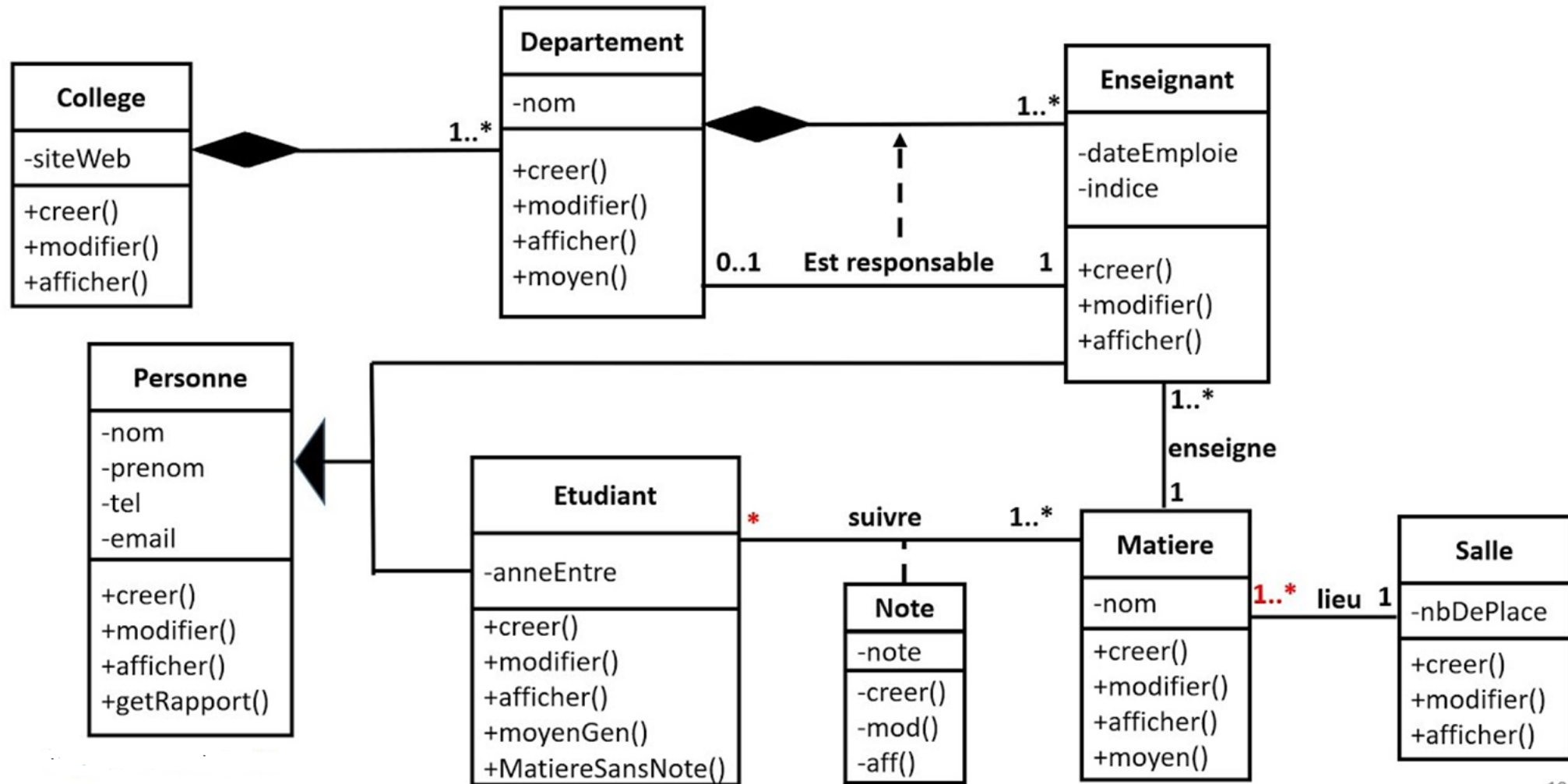
EXEMPLE 1 DE DIAGRAMME DE CLASSES (BIBLIOTHÈQUE)



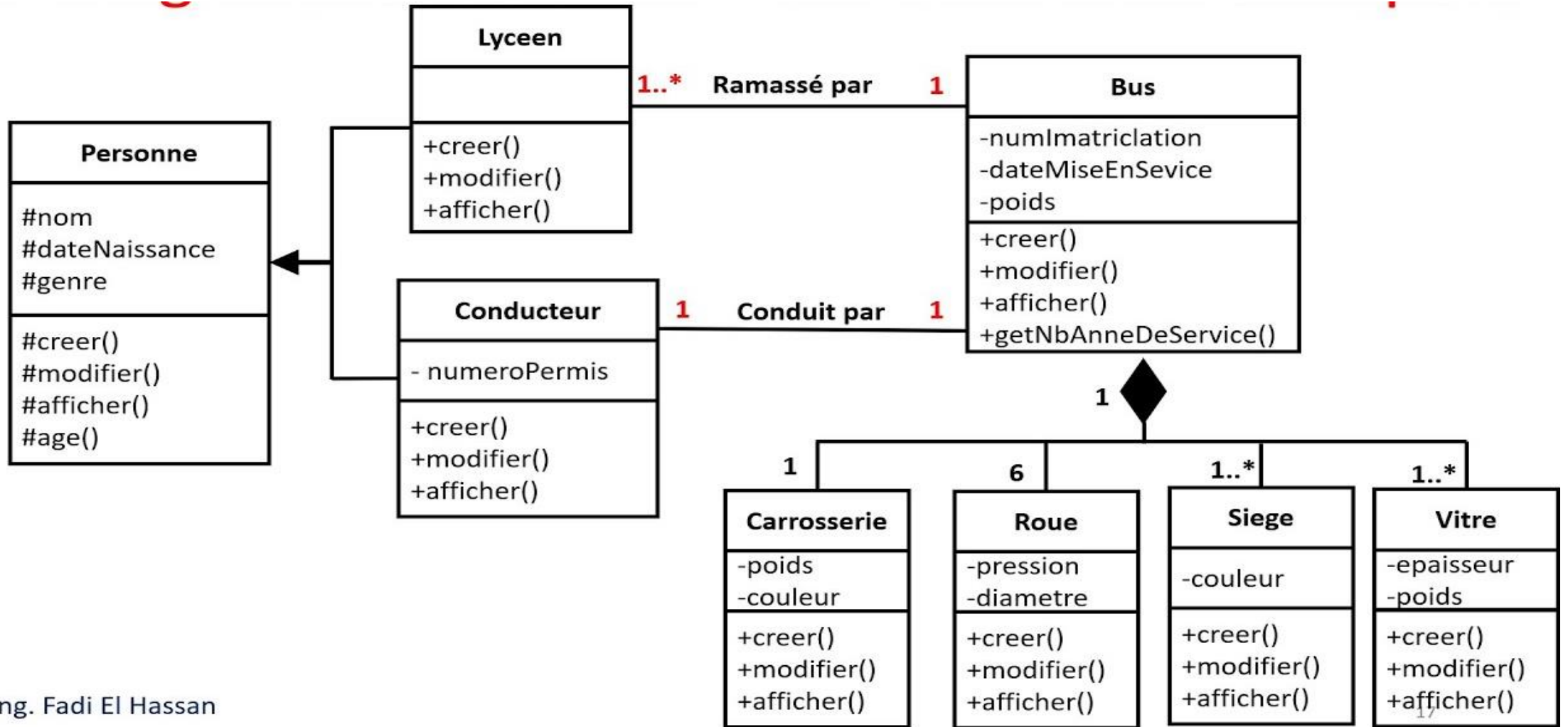
EXEMPLE 2 DIAGRAMME DE CLASSE



GESTION DES COURS UNIVERSITAIRES



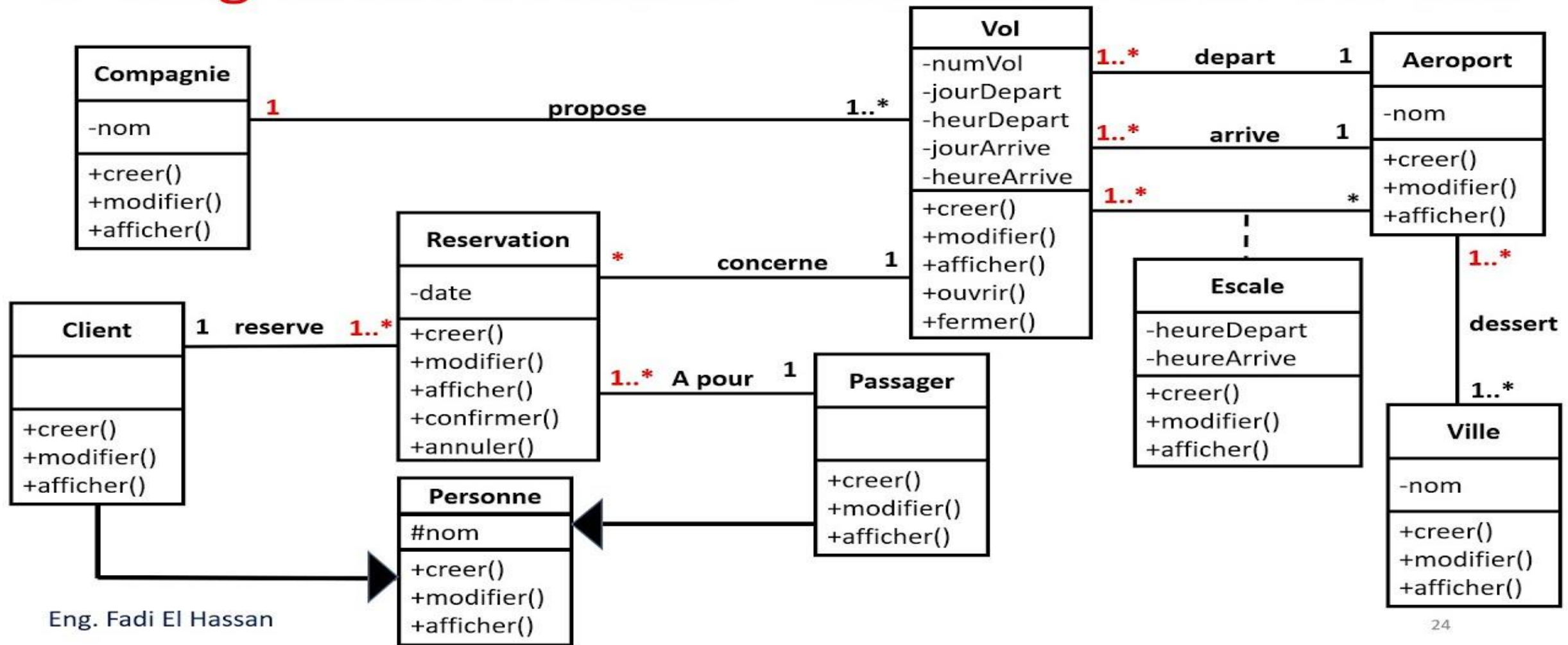
GESTION SOCIÉTÉ DE TRANSPORT



Eng. Fadi El Hassan



GESTION RÉSERVATION DES VOLS



Eng. Fadi El Hassan

