UML : LES CONCEPTS OBJETS

Par PAUL EDIDE
EMAIL : paul.edide@gmail.com

PLAN :

I. Objet
II. Classe
III. Instance d'une classe
IV. Différentes conceptualisation
V. Les Méthodes
VI. Les diagrammes de classes
VII. Les constructeurs
VII. Cas pratique
VIII. Liens et Associations
IX. Les associations
X. Multiplicités des associations
XI. Classe-association
XII. Agrégation
XIII. Héritage
XIV. Cas pratique

INTRODUCTION :

Dans ce chapitre nous parlerons du concept objet, des classes et types associées.

II. Objets

Le but de la modélisation objet est de décrire les objets.

La description d'un objet est une abstraction ayant des limites claires et un sens précis dans le contexte du problème étudié.

Un objet possède une identité et peut être distingué des autres.

En regroupant les instances en classes, on décrit les instances par leurs propriétés générales, de manière abstraite.

Dans une classe, on ne décrit qu'une fois la structure et le comportement commun d'un ensemble d'objets.

Une classe précise :

- les propriétés (attributs) des instances

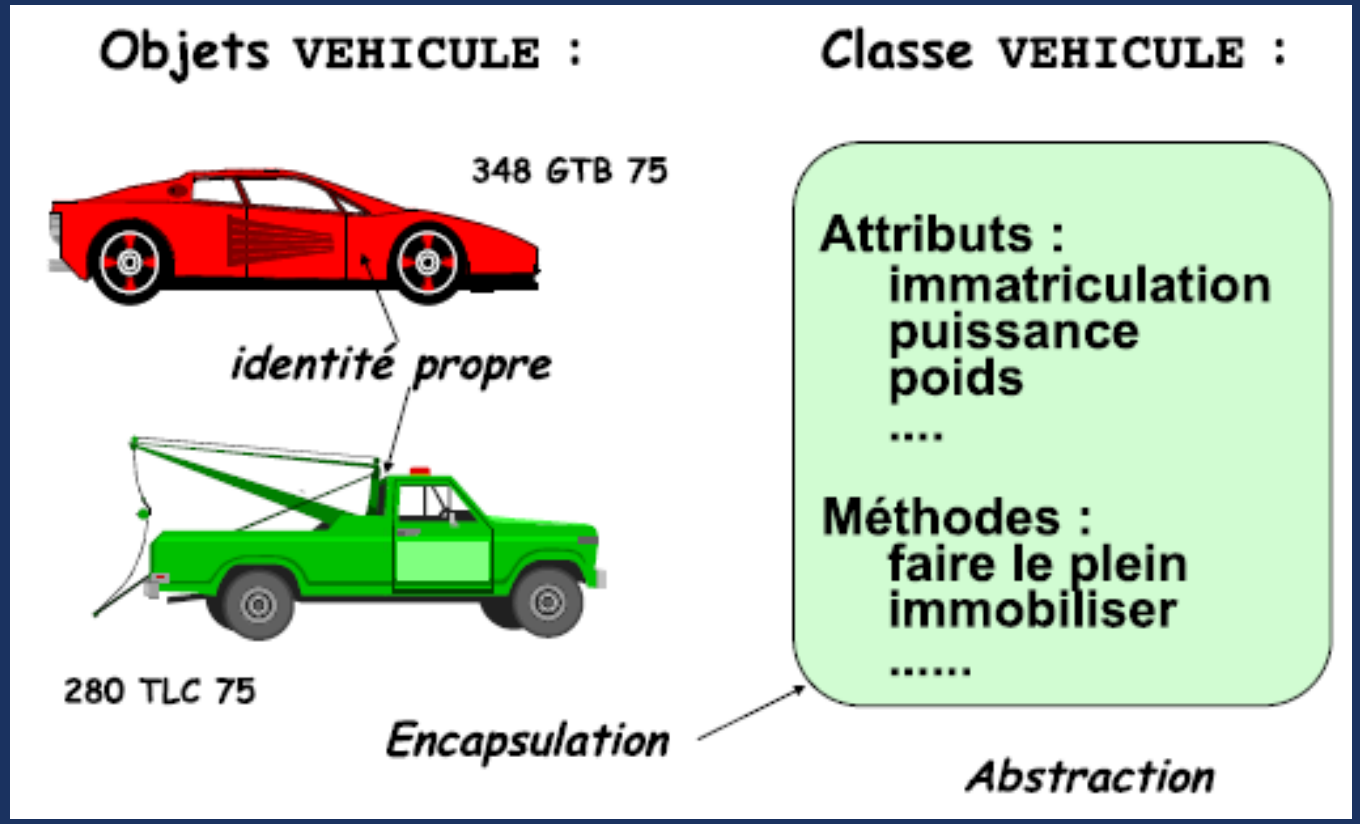
- le comportement (méthodes) des instances.

II. Classe

Une classe définit les caractéristiques et le comportement communs à un ensemble d'objets (les instances de la classe).

Remarque : Toute objet est instance d'une classe (d'une seule) c'est-à-dire, un objet ne peut avoir qu'une seule définition.

Exemple :



III. Instance d'une classe

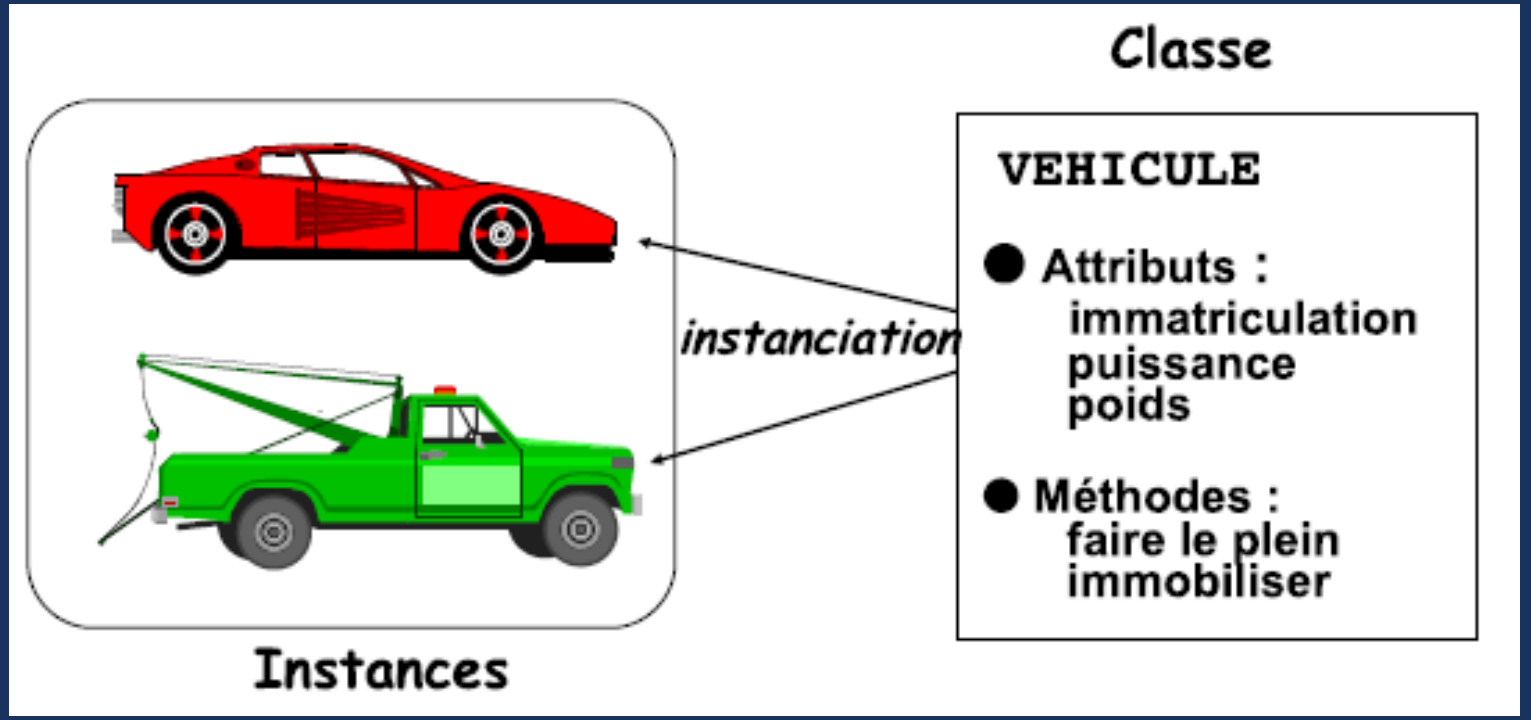
Une instance est la concrétisation d'une classe.

Les instances d'une classe partagent un objectif sémantique commun qui dépend du point de vue de modélisation.

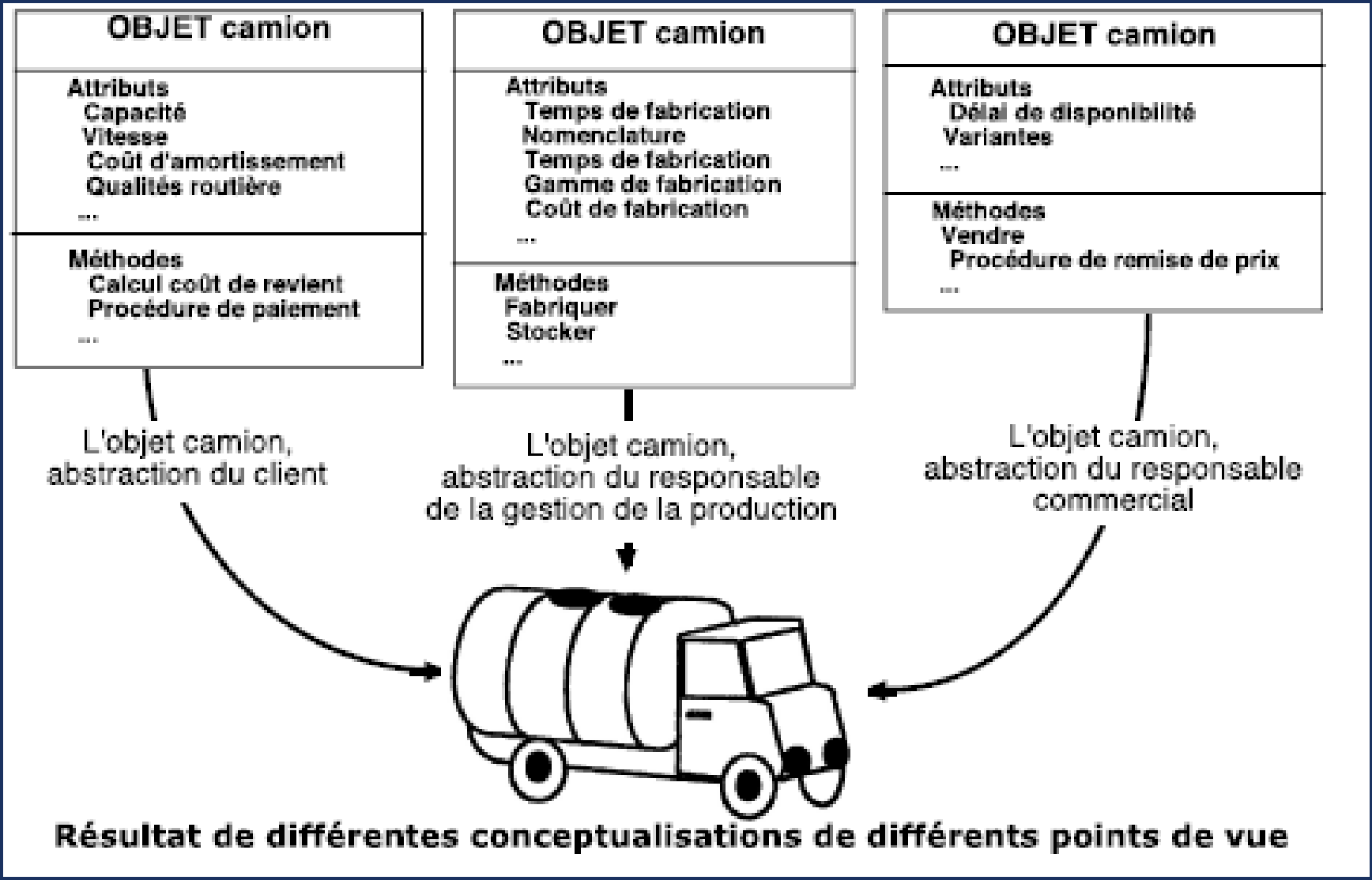
Exemple :

- Du point de vue des actifs financiers : véhicule et micro-ordinateur peuvent appartenir à une même classe.

- Du point de vue inventaire matériel : véhicule et le micro-ordinateur n'appartiennent pas à une même classe.



IV. Différentes conceptualisation



Résultat de différentes conceptualisations de différents points de vue

V. Les Méthodes

La classe décrit :

- les méthodes (comportement),

Une méthode est une fonction pouvant comporter des paramètres et une valeur de retour.

Le nom de la méthode, le type de valeur retournée et le type des paramètres (et les exceptions) forment la signature de la méthode.

Méthodes particulières :
- Constructeur : création d'instances

- Destructeur : suppression d'instances.

- le type des variables (structure) et valeur initiales,

- les valeurs partagées (variables de classes).

VI. Les diagrammes de classes

Les diagrammes de classes ("class diagram") permettent de modéliser les classes et les relations entre classes.

Les classes sont représentées par un rectangle.

Il est possible de les représenter avec plus ou moins de "détail" :

- uniquement par leur nom,

- par leur nom et noms d'attributs, et au besoin d'opérations

- de manière complète, avec indication du type des attributs, de valeur par défaut et des signatures d'opérations.

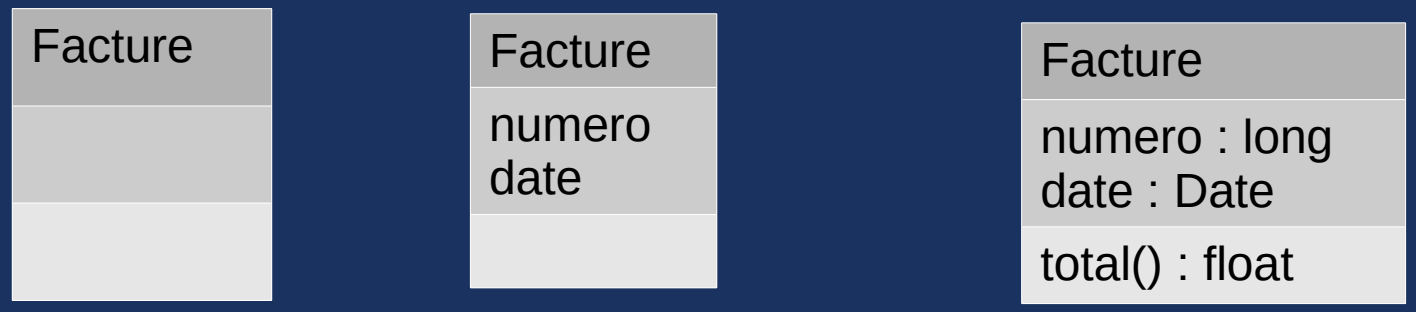
VI. Les diagrammes de classes

Notation pour les classes :

- Le nom de la classe commence par une majuscule

- et les autres par une minuscule.

Exemple :



Juste le nom de la classe

Le nom de la classe et des attributs

Le nom de la classe et des attributs avec leur type et les noms des méthodes avec leur signature

VII. Cas pratique :

Exercice : Calev Devise

VIII. Liens et Associations

Un lien représente une relation entre deux objets.

Un lien est une connexion physique ou conceptuelle entre des instances d'objets : un lien est un tuple d'instances.

Un lien est une instance d'association.

Une association décrit un groupe de liens ayant une structure et une sémantique commune.

Une association réunit des classes. Tous les liens d'une association mettent en relation des objets de même classe.

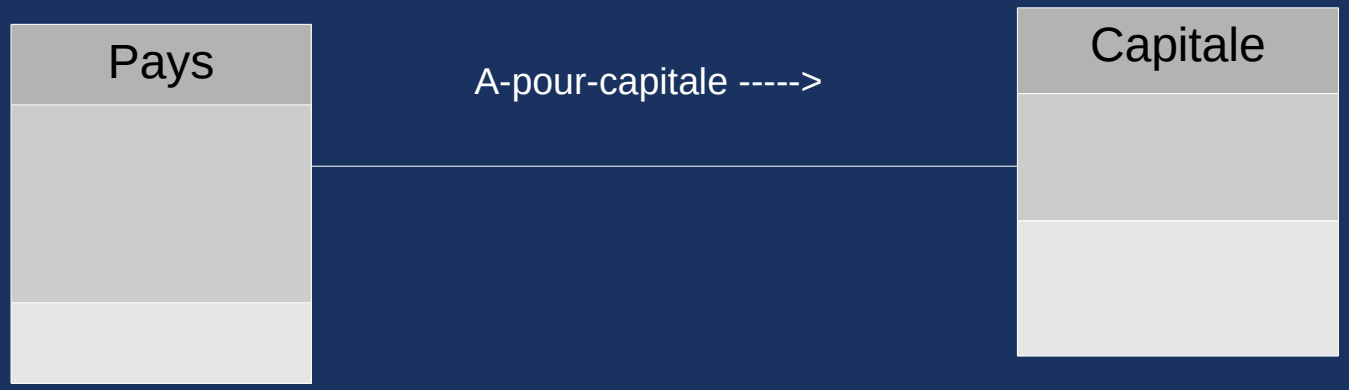
Les associations sont bidirectionnelles (binaires).

IX. Les associations

Notation pour les associations.

Les associations peuvent être nommées :

"a-pour-capitale" nomme l'association dans le sens "Pays" vers "Capitale"

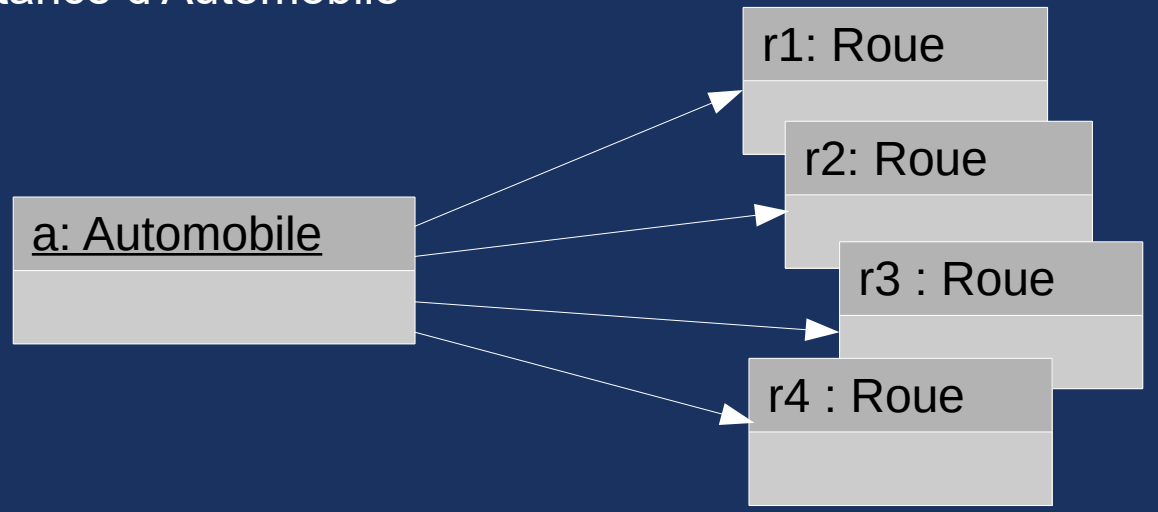


X. Multiplicités des associations

La multiplicité d'une association exprime le nombre de liens entre les instances de chaque classe de L'association.

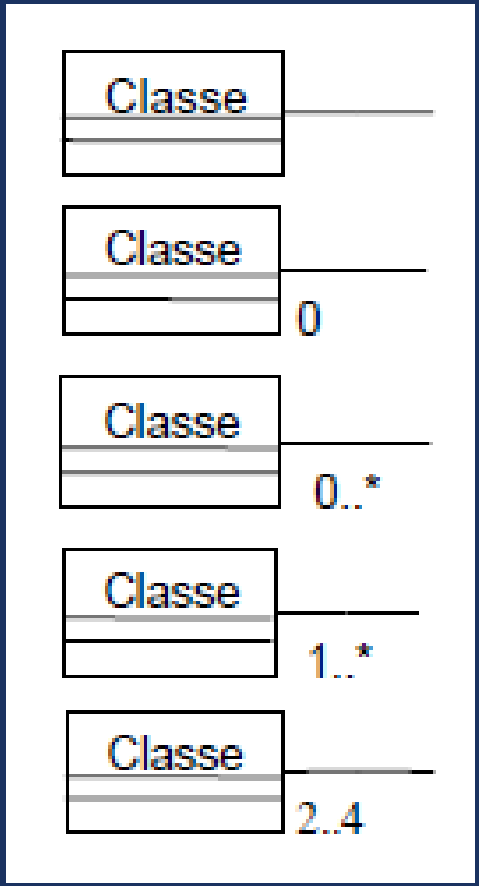
Exemple: dans l'association "Automobile" à "Roue", une instance d'Automobile est liée à 4 instances de Roues, et, une instance de Roue est liée à une seule instance d'Automobile

Association



Liens

XI. Multiplicités des associations



Exactement un

Optionnel (zéro ou un)

Plusieurs (zéro ou plus)

Un ou plus

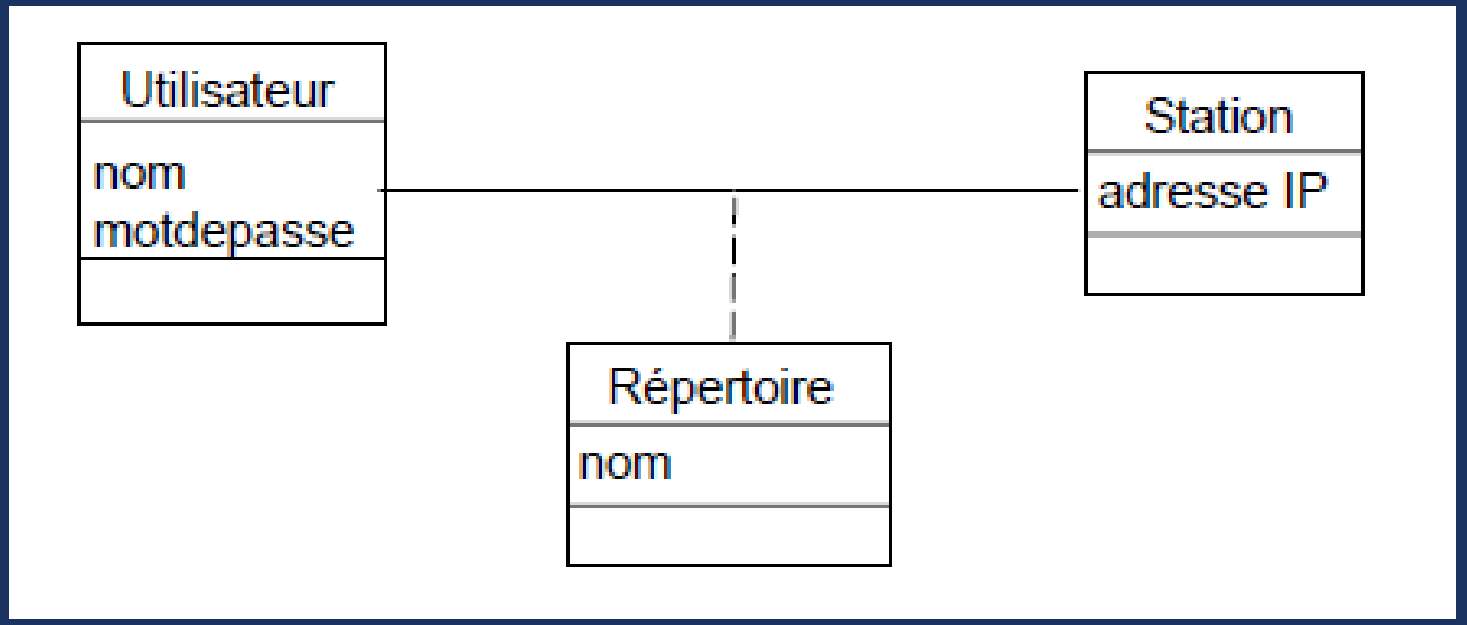
Spécifié numériquement (1.X, mais pas 2.0)

XI. Classe-association

Les associations avec attributs de liens peuvent être modélisé sous la forme d'attributs d'une classe : l'association est considérée comme une classe ou nommée "Classe-association" (Link Association).

Dans une association modélisée sous la forme d'une "classe-association", chaque lien est une instance de cette "classe-association".

Exemple de modélisation d'une classe association.



XII. Agrégation

L'agrégation est la forme particulière d’association entre deux classes indiquant que des instances d’une classe (par ex. LIGNE_FACTURE) sont contenues (ou agrégées) dans une instance d’une autre classe (par ex. FACTURE).

Dans l’agrégation les deux objets en relation sont distingués : l’un est un composant de l’autre.

Au niveau logique, on considère que le composé est responsable de la gestion de ses composants (c’est le composé qui crée, modifie, ou détruit ses composants).

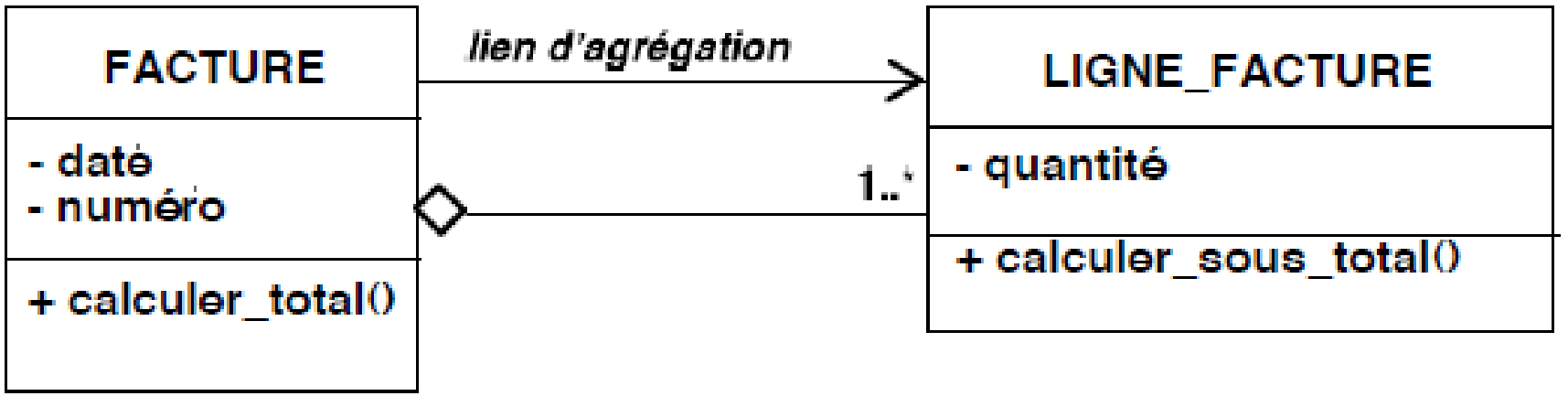
Une agrégation est une relation "composé-composant" ou "partie-de".

Un des objets participant à l'agrégation est un composé, un assemblage de composants ou de parties

Exemple, une facture est un composé de lignes factures

XII. Agrégation

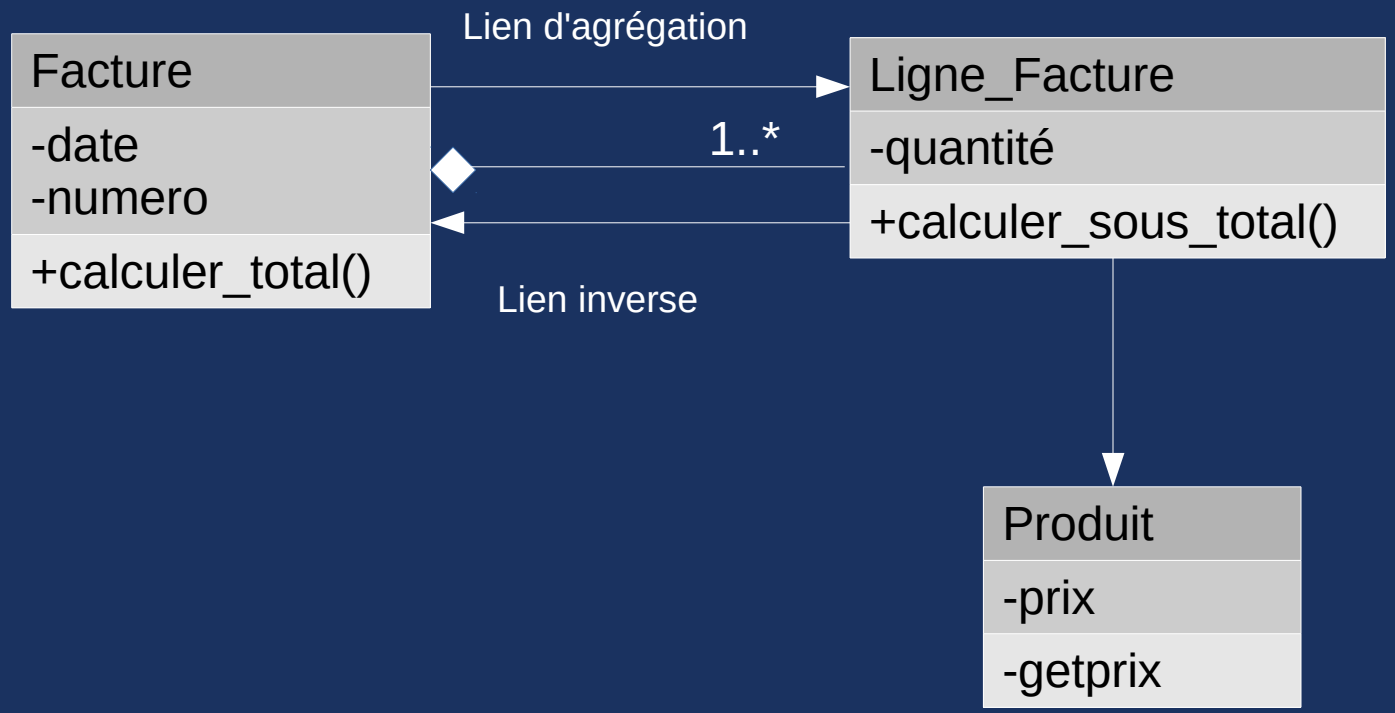
Notation :



XII. Agrégation

L'agrégation n’est pas porteuse de sens.

L'agrégation est orientée du composé vers le composant (création d'un lien inverse si nécessaire)



XII. Agrégation

L'agrégation est une association dotée d'une sémantique additionnelle.

Elle est :
- antisymétrique (à la différence de l'association qui est bidirectionnelle)

- L'agrégation peut être transitive : un objet composant
peut lui même être un objet composé.

Exemple, une automobile se compose (entre autres) d’un bloc-moteur et d’un châssis.
Le bloc-moteur se compose d’une boîte de vitesse, d’un carburateur…

- L’agrégation n'est ni symétrique ni bi-directionnelle :
car l’agrégation distingue un composé d’un composant

XII. Agrégation

Comment reconnaître une agrégation ?

Une association est une agrégation si :

- Peut-on utiliser l'expression partie-de ?

- Des opérations appliquées sur le composé sont elles appliquées automatiquement aux composants (le composé gère ses composants) ?

Exemple : totaliser une facture

- Des valeurs d'attributs sont-elles propagées du composé vers des composants ?

Exemple : la date d'une facture "concerne" les lignes factures

- Y a-t-il une asymétrie intrinsèque dans l'association dans laquelle une classe d'objet est subordonnée à l'autre ?

XIII. Héritage

L'héritage est la transmission de caractéristiques à ses descendants

L'héritage correspond à la relation "est-un" :

- Un CONDUCTEUR est une PERSONNE

- Un CONDUCTEUR est une spécialisation de PERSONNE

- Une PERSONNE est une généralisation d’un CONDUCTEUR.

La classe qui hérite dispose :

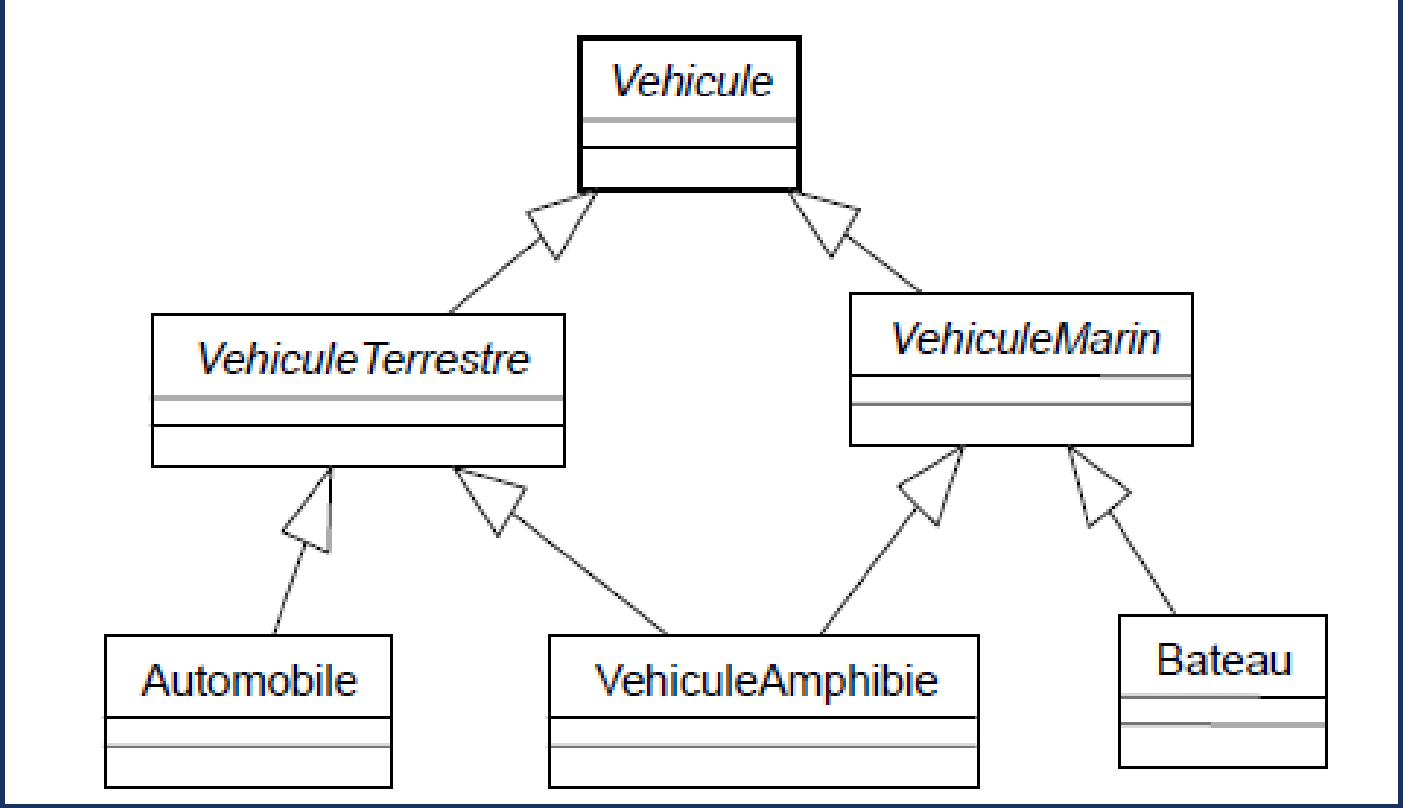
- Des méthodes de niveau public et protégé

- Des attributs de niveau public et protégé

Par contre les attributs et méthodes de niveau privé ne sont pas hérités.

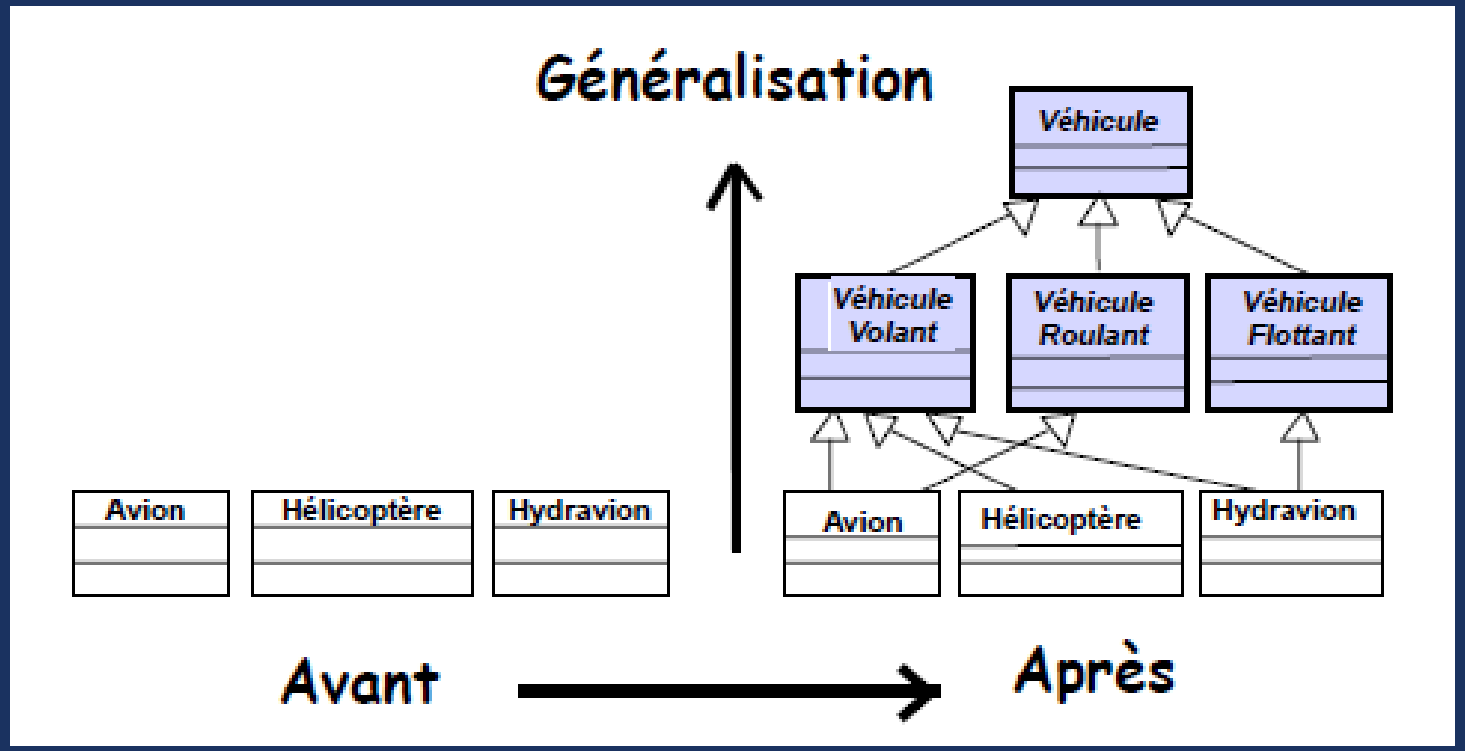
XIII. Héritage

Notation :



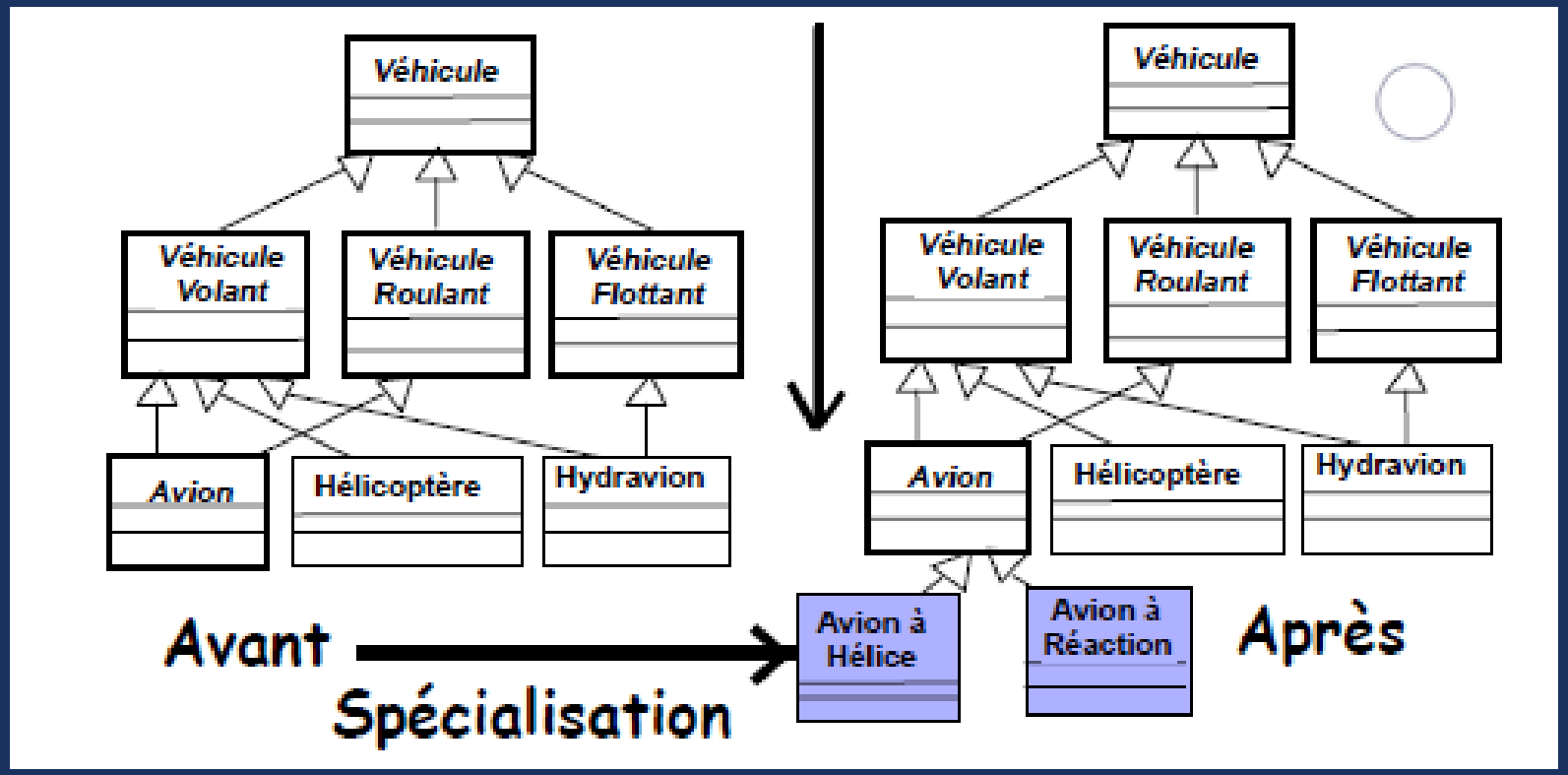
XIII. Héritage

La généralisation est l'abstraction des caractéristiques communes à plusieurs classe d'objets.



XIII. Héritage

La spécialisation est la dérivation de nouvelles abstractions contenant des caractéristiques supplémentaires.



XIII. Héritage

Une classe définit un type.

- La classe CONDUCTEUR héritant de la classe PERSONNE, le type CONDUCTEUR est un sous-type du type PERSONNE.

Pour que l'affectation d'un sous-type à un type soit valide il faut que :

- une instance de CONDUCTEUR soit aussi une instance de PERSONNE

- l'ensemble des instances de CONDUCTEUR soit inclus dans l'ensemble des instances de PERSONNE.

Une classe B hérite d'une classe A si l'ensemble des instances de la classe B peut être inclus dans l'ensemble des instances de la classe A.

- la classe CONDUCTEUR héritera de la classe PERSONNE si l'ensemble des instances de CONDUCTEUR peut être inclus dans l'ensemble des instances de PERSONNE.

XIV. Cas pratique :

Exercice : Calev Devise