

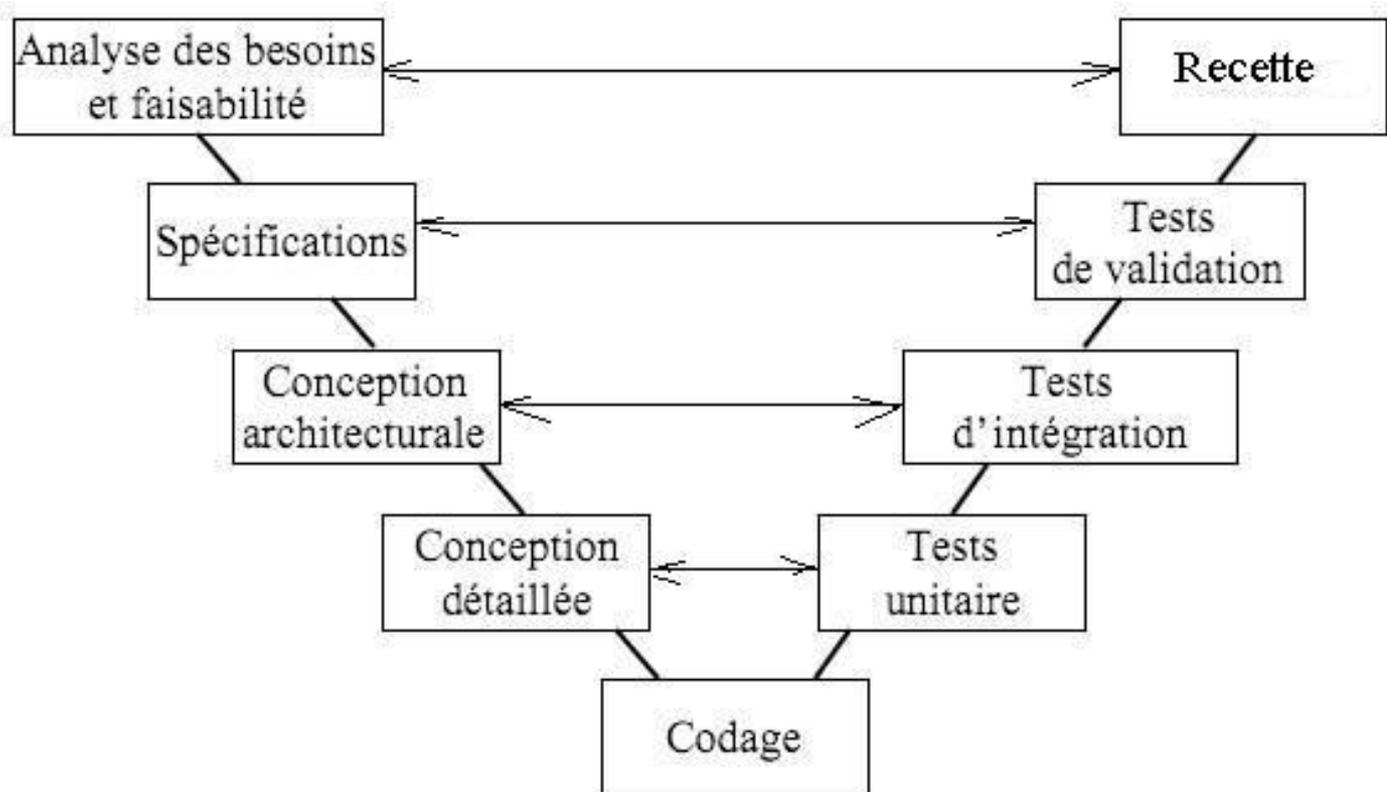


Analyse et Conception Orientées Objets

Cours I : Analyse Orientée Objet

Cycle de vie d'un projet logiciel

- Cycle en V :



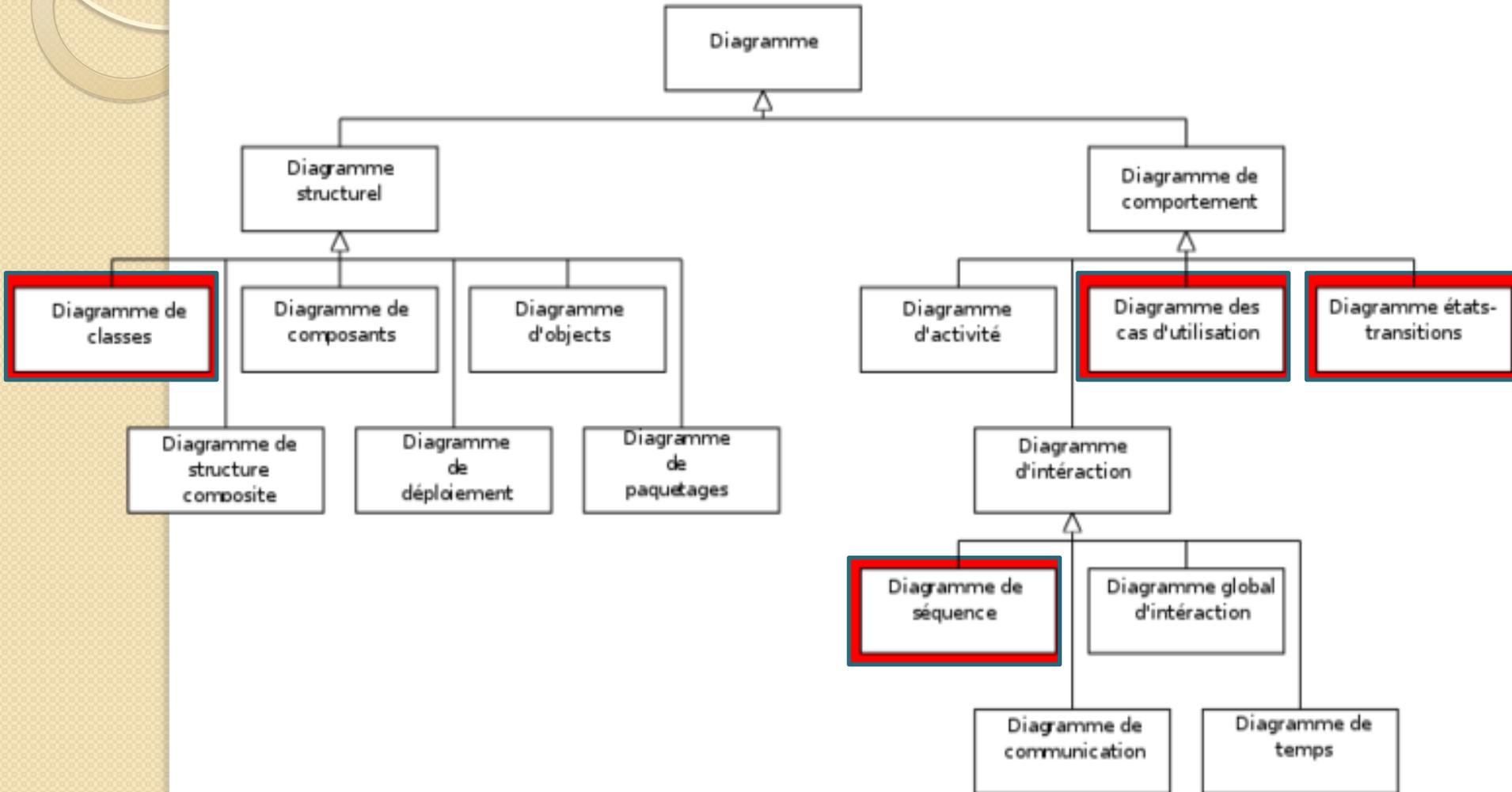
Définitions

- **Analyse** : de la perception réelle vers la représentation
 - L'analyse consiste sur la base d'un cahier de charge de modéliser les acteurs et le système en ne s'intéressant pas à l'implémentation.
 - On parle d'approche métier :
QUI (acteurs) fait QUOI (système)
- **Conception** : de la représentation vers la représentation du prototype
 - La conception consiste sur la base de l'analyse à modéliser la réalisation informatique.
 - On parle d'approche informatique :
COMMENT ?

Historiques d'UML

- UML : Unified Modeling Language, standard d'OMG
- Fusion de 3 langages de modélisation OO (Booch, OMT, OOSE) en 1995.
- Dernière version UML 2.4.1 août 2011
- UML offre un standard de modélisation pour spécifier, visualiser, modifier et construire les documents nécessaires au bon développement d'un logiciel orienté objet

Taxonomie des diagrammes



L'approche orientée objet

- Modélisation du monde réel à l'aide d'un système formé d'objets
- Un objet est une représentation :
 - soit d'une réalité vivante comme un être humain, un animal, ..
 - soit d'une réalité matérielle comme une voiture, une bouteille
 - soit d'une réalité immatérielle ou abstraite comme une idée, une dette, la sécurité sociale

L'approche orientée objet

Cette modélisation est composée de 3 parties :

- **l'identité de l'objet** : ce qui permet de le distinguer d'un autre objet
- **l'état de l'objet** : ensemble des valeurs d'attributs
- le **comportement** : ensemble des méthodes qui décrivent ce qu'on peut faire avec.

Identité de l'objet

```
Porsche 911 : Voiture
```

Etat de l'objet

```
45789 : Numéro de série  
911 : Modèle  
1500 Kg : Poids  
32 litres : Quantité d'essence
```

Comportement de l'objet

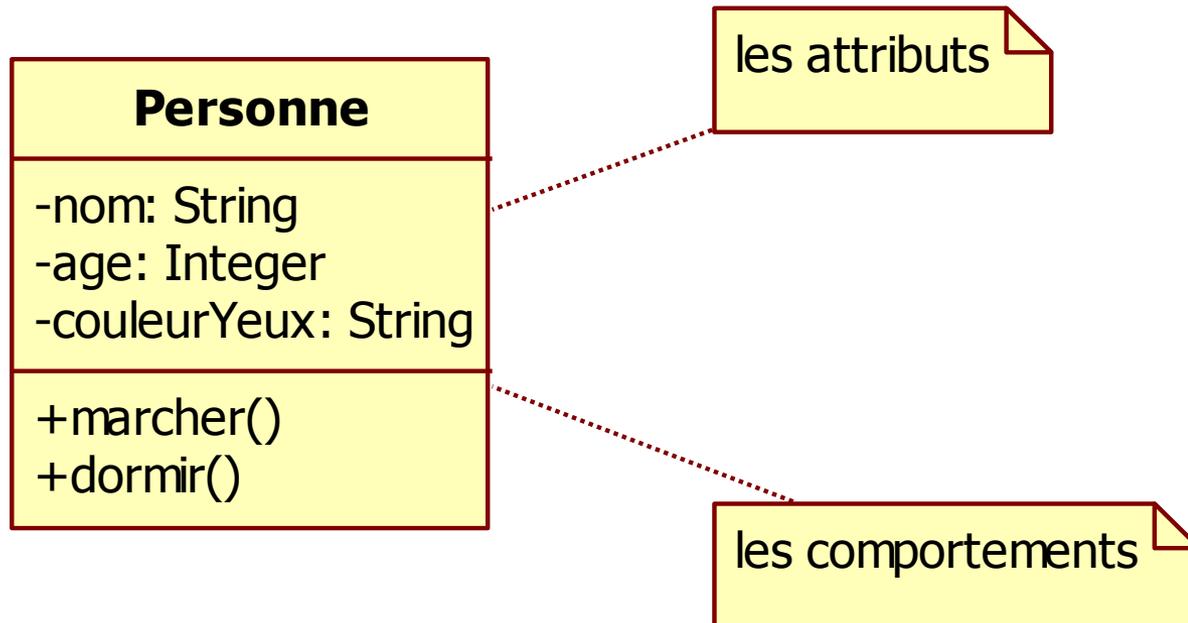
```
démarrer()  
arrêter()  
accélérer()  
rouler()
```

Classe et Objet

- Une Ferrari peut se décrire de la même façon qu'une Porsche 911.
- Le nom et les valeurs des attributs permettent de différencier la Porsche 911 de la Ferrari
- On introduit un nouveau concept : **classe**.
- Une classe est une description commune d'objets.
- Un objet est une instance d'une classe. Seuls les objets auront une existence dans les programmes

UML et classes

- En UML, on décrit une classe comme l'indique le schéma qui suit :



Etats d'un objet

- Les valeurs des différents attributs d'un objet déterminent l'état de l'objet
- Un objet doit toujours être dans un état cohérent. Un état est cohérent quand il est prévu dans les spécifications.

Un objet Evaluation a un attribut **note**. Si une note doit être comprise entre 0 et 20 alors un objet Evaluation qui a son attribut **note** égal à -1 n'est pas dans un état cohérent.

Etat et comportement

- Le comportement dépend de l'état. Une opération ne s'exécutera pas nécessairement de la même façon pour deux états de l'objet.
- L'effet de l'opération **accélérer** dépend de la quantité d'essence dans la Ferrari.

Etat et comportement

- L'état d'un objet ne peut être changé que par le comportement.

C'est le concept d'**encapsulation**

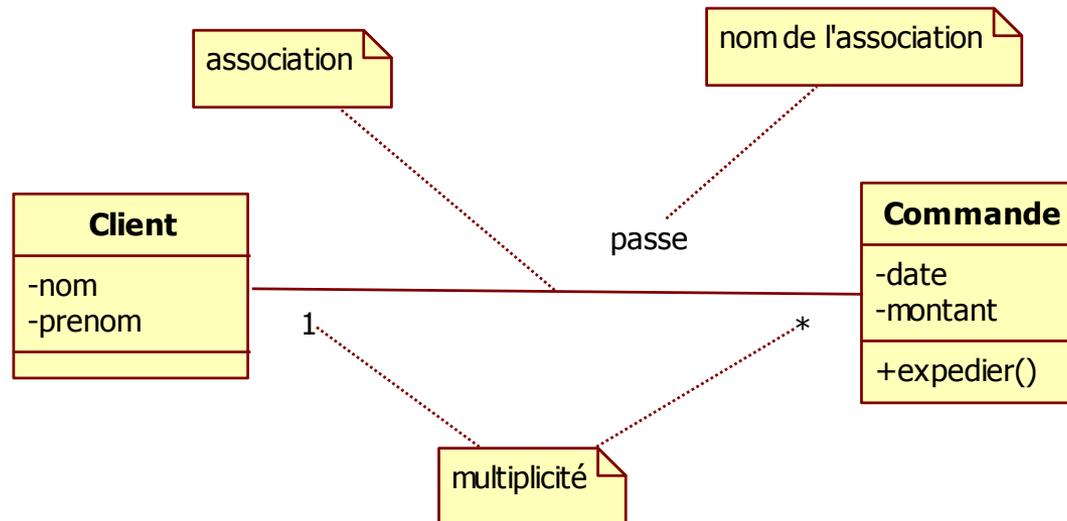
L'exécution de l'opération **accélérer** diminue la quantité d'essence dans la Ferrari.

Diagramme de classes

- Dans un système non trivial, on gèrera plusieurs objets d'une mème classe et/ou de classes différentes .
- Il faut donc non seulement dècrire les objets avec des classes mais aussi les liens qu'il peut y avoir entre objets de différentes classes

Association

- L'association est le lien le plus pauvre entre deux classes.
- Il traduit le fait qu'un objet d'une classe est associé à un ou plusieurs autres objets d'une classe

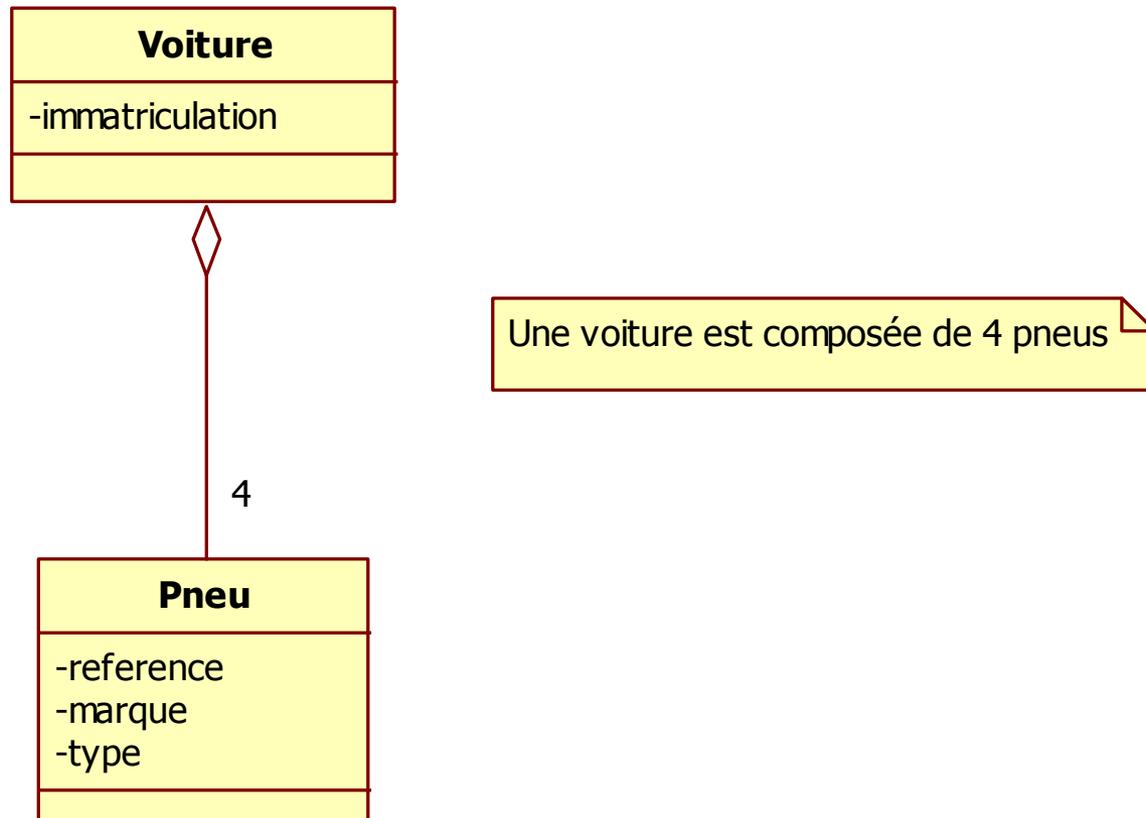


Association et Agrégation

- Une agrégation est une forme contrainte d'une association dans laquelle l'une des classes décrit un tout alors que la classe associée décrit des parties.
- On appelle le tout un composé (ou composite) et une partie un composant.

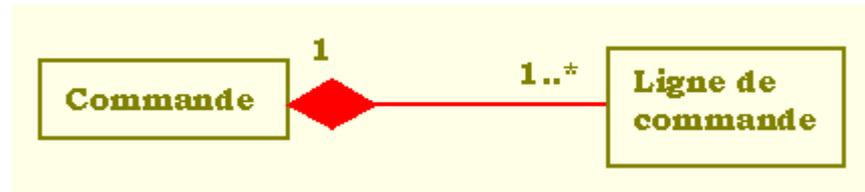
Association et Agrégation

- Une agrégation est une association dissymétrique

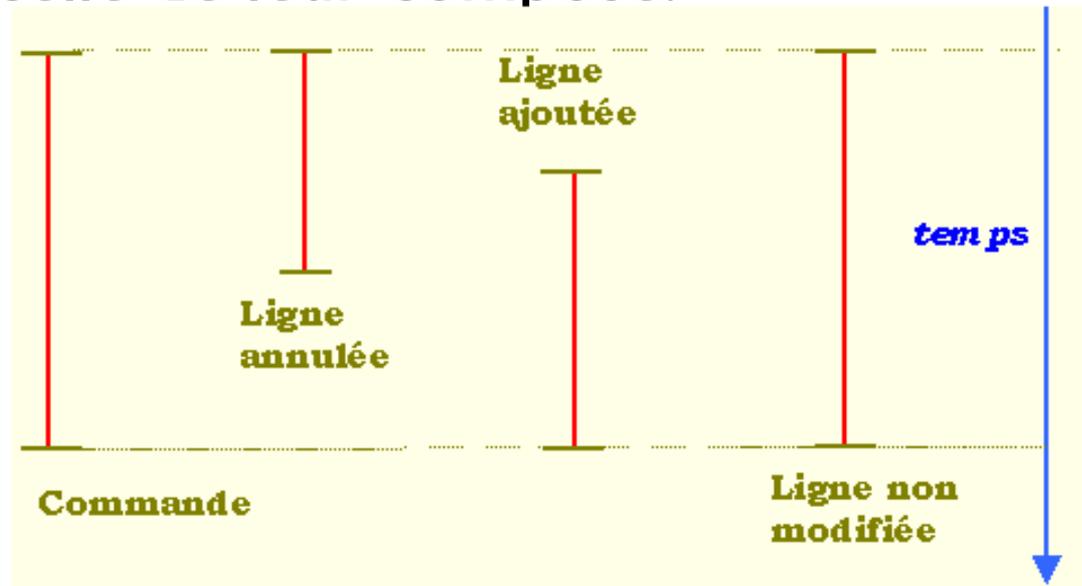


Agrégation et Composition

- Une composition est une forme contrainte d'une agrégation.



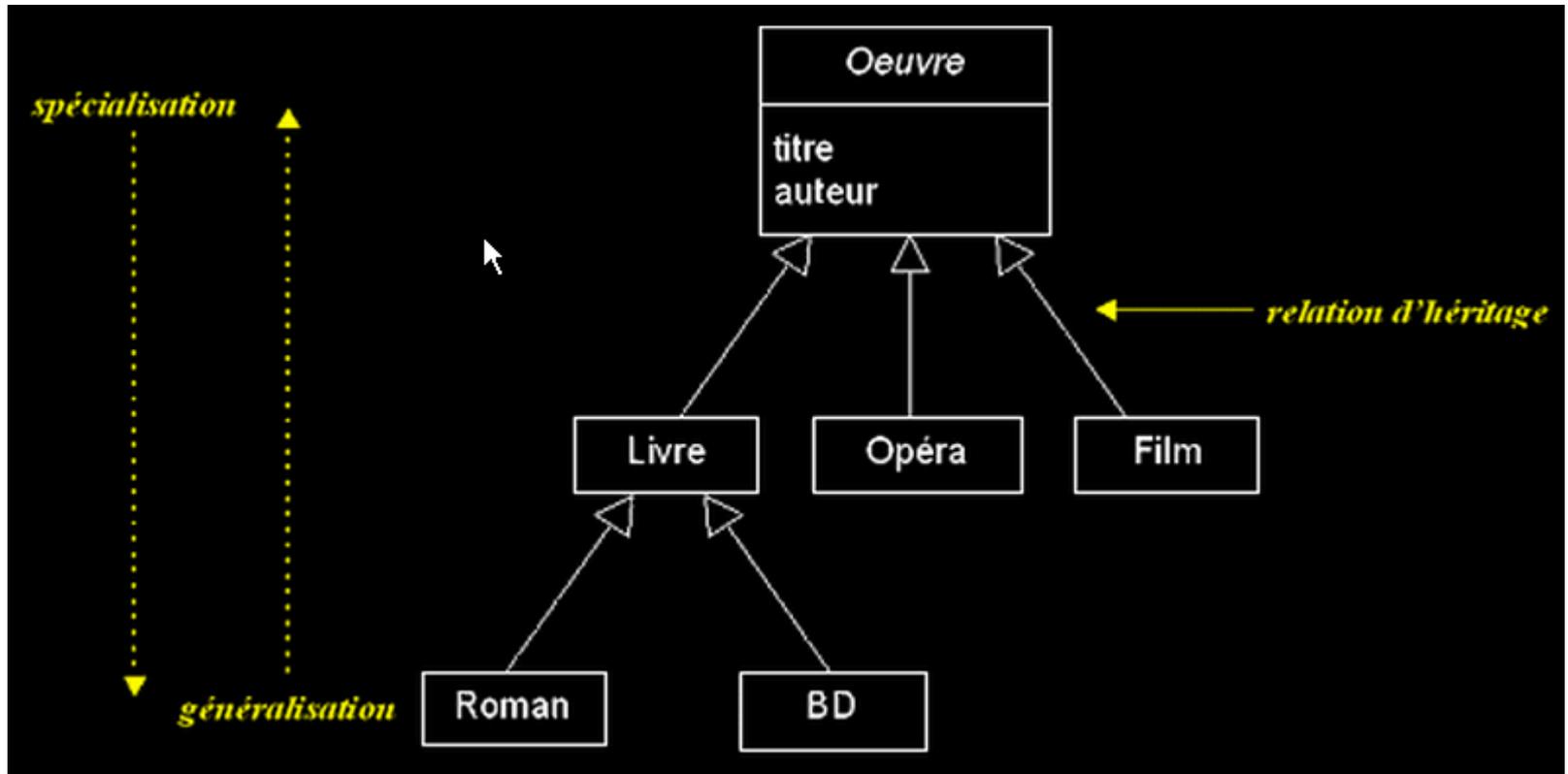
- Les composants ont une durée de vie incluse dans celle de leur composé.



Héritage : le concept

- Une généralisation est une relation conceptuelle entre classes, respectivement parent, enfant(s), telle que la classe enfant :
 - 1) possède les mêmes descriptions (attributs, opérations, associations) que sa classe parent
 - 2) ajoute des descriptions (attributs, opérations, associations) qui lui sont spécifiques.

Héritage : un exemple (uml.free.fr)



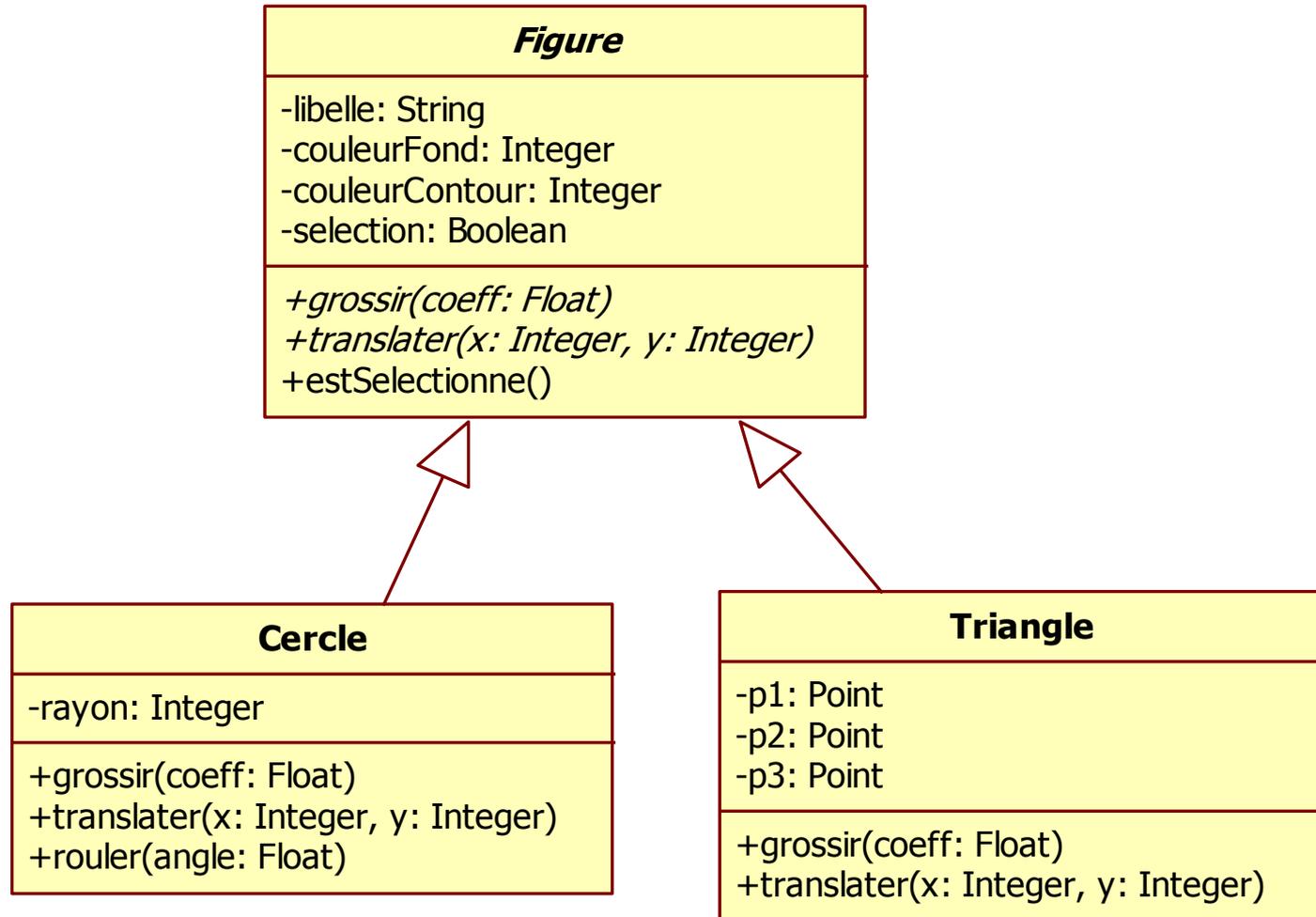
Héritage : remarques

- Dans une classe **Enfant**, certaines opérations de la classe **Mère** peuvent être redéfinies
- Un piège sémantique de l'expression :
 - Titi **est un** canari
 - Un canari **est un** oiseau
 - La première phrase est un classement. Titi est une instance de la classe **Canari**.
 - La deuxième phrase est une catégorisation. **Canari** est une sous classe de la classe **Oiseau**.

Héritage et Classes abstraites

- Certaines classes peuvent ne pas être entièrement définies pour donner lieu à des instances. On dit qu'elles sont **abstraites**.
- Elles ne seront utilisées qu'en tant que classes mères dans un héritage.
- Exemple : dans un logiciel de dessins, on fabrique des dessins. Un dessin est un ensemble de figures de formes différentes.

Héritage et Classes abstraites



Héritage et Classes abstraites

- La classe **Figure** est abstraite.
- Les instances seront des triangles ou des cercles mais pas de figures.
- Par contre, on constate que les deux classes Triangle et Cercle ont en commun une partie de leur comportement. C'est le concept de **polymorphisme**.

Les quatre concepts de l'objet

- **Etat + comportement**
- **Encapsulation**
- **Héritage**
- **Polymorphisme**



Démonstration StarUML