



Modélisation UML

Aurore Blot – GRTgaz

aurore.blot@insa-rouen.fr



Chapitre 04

Classes & Associations particulières

Rappel

Associations
multiples &
réflexives

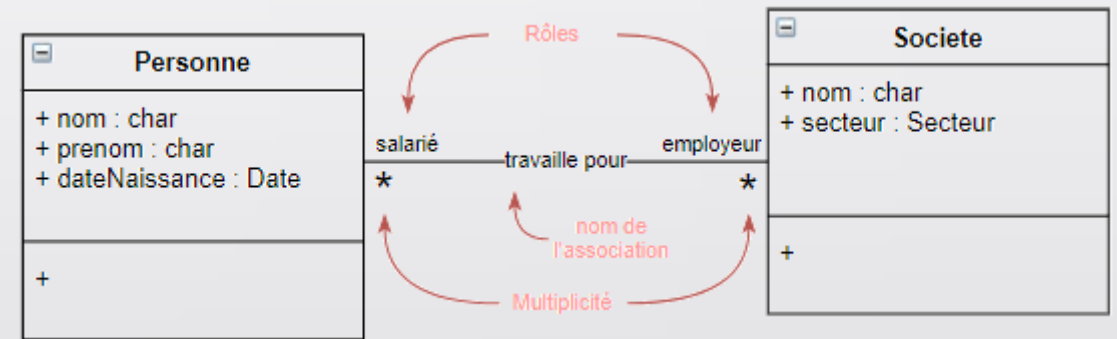
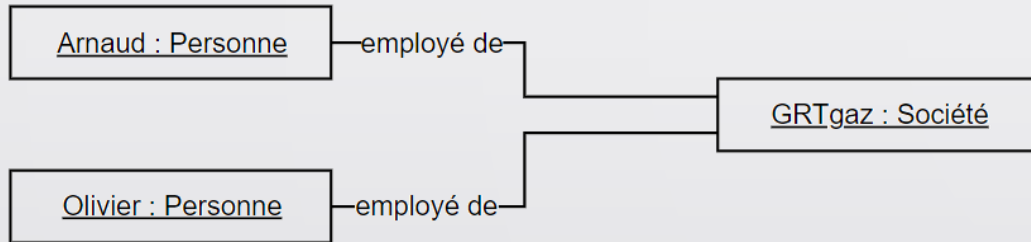
Classe
association

Agrégation
&
Composition

Héritage

Classe
abstraite &
interface

Rappel des bases de diagramme d'objets/ de classe





Chapitre 04

Classes & Associations particulières

Rappel

Associations
multiples &
réflexives

Classe
association

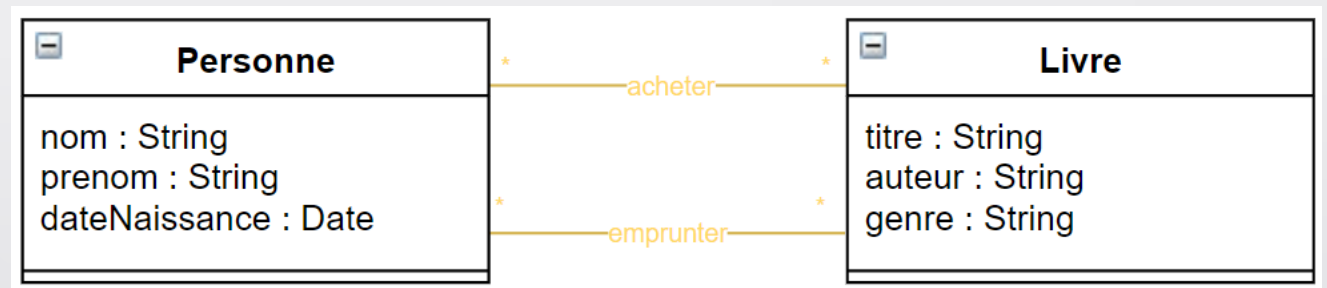
Agrégation
&
Composition

Héritage

Classe
abstraite &
interface

Associations multiples

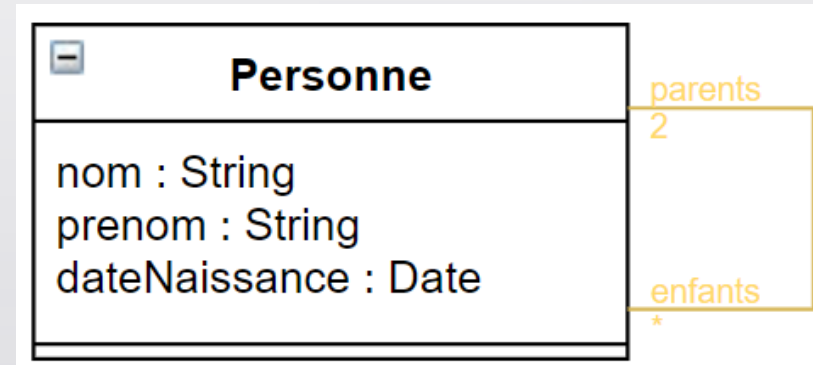
- Association la plus utilisée = binaire
- Parfois, les associations sont multiples



Associations réflexives

*Exercice : exemple de
diagramme d'objets associé*

- Parfois, une association pointe sur la classe d'origine.
- Exemple





Chapitre 04

Classes & Associations particulières



Rappel

Associations
multiples &
réflexives

Classe
association

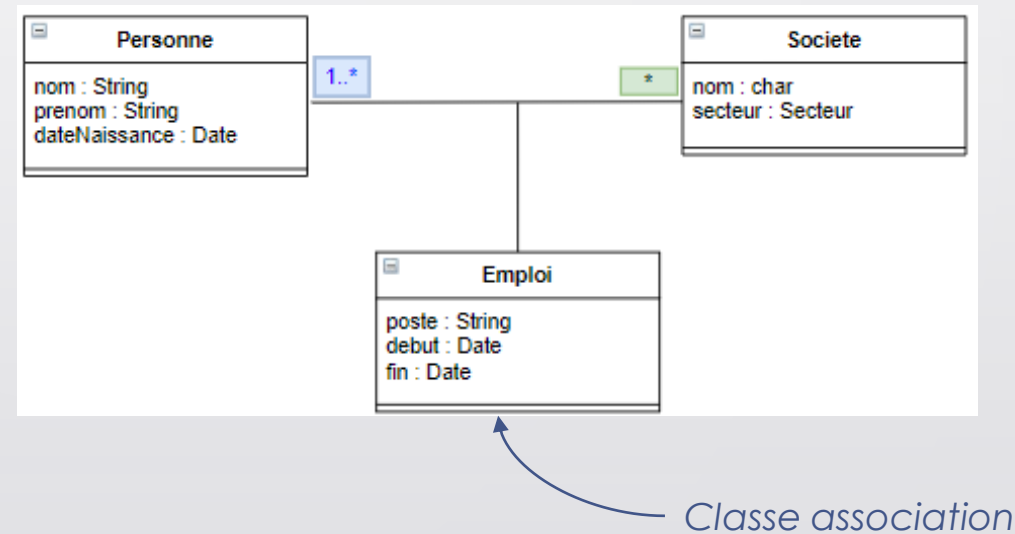
Agrégation
&
Composition

Héritage

Classe
abstraite &
interface

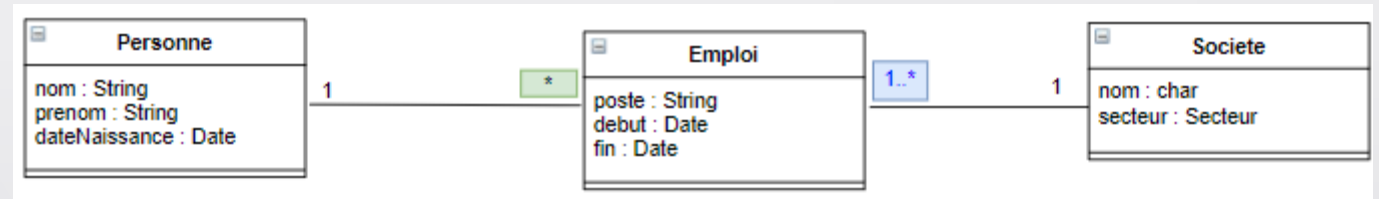
- Cas où une classe en associe deux autres
- On a une unique instance de la classe association qui lie chaque objet.

Classe-association



- On peut toujours établir une correspondance en association purement binaire

Classe-association





Chapitre 04

Classes & Associations particulières



Rappel

Associations
multiples &
réflexives

Classe
association

Agrégation
&
Composition

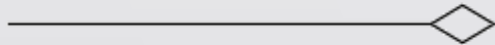
Héritage

Classe
abstraite &
interface



Agrégation

Représentation

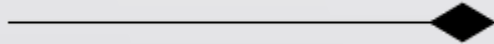


- L'association est une relation qui relie une (ou plusieurs) classes.
- **Agrégation = association non symétrique** ; une classe joue un rôle prédominant par rapport à l'autre



Composition

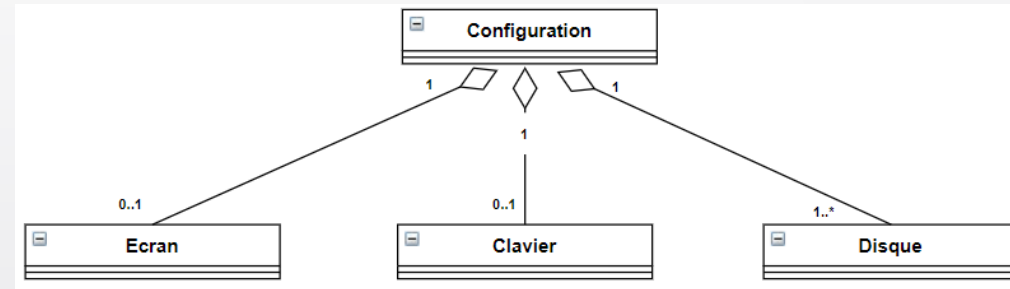
Représentation



- **Composition = agrégation forte**
- **= association non symétrique avec coïncidence des durées de vie** des composants
- La destruction du composé implique la destruction de tous les composants liés.
- La cardinalité côté composant ne peut être que de 1.

Agrégation

Exemple



- Une configuration comporte :
 - Éventuellement un écran
 - Éventuellement un clavier
 - Au moins un disque

Composition

Exemple



- Une application est composée éventuellement de fenêtre (aucune ou plusieurs).
- La fenêtre n'existe pas en dehors de l'application.



Agrégation & Composition

- Relation entre un tout et des parties
= relation d'agrégation ou de composition
- **Quelle relation choisir ?**
- Est-ce que la destruction du tout implique la destruction des parties ?
- Un composant peut-il être réutilisé ?



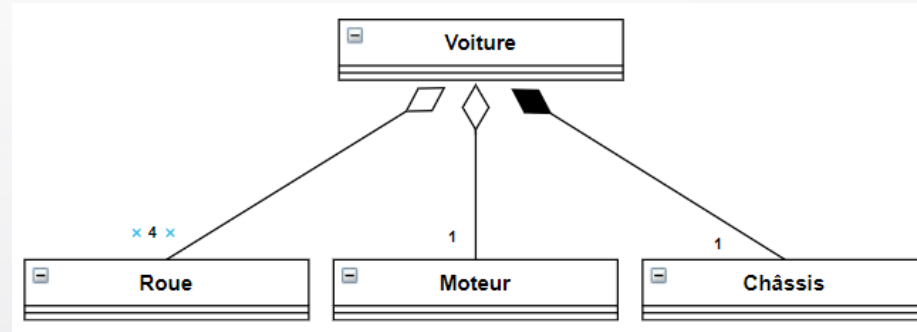
Agrégation & Composition

Exemple

- Une voiture est composé de 4 roues, d'un moteur et d'un châssis.

Agrégation & Composition

Exemple



- Une voiture est composée de 4 roues, d'un moteur et d'un châssis.
- Les roues et le moteur peuvent être utilisés pour d'autres voitures.
- Le châssis est indissociable de la voiture.



Chapitre 04

Classes & Associations particulières



Rappel

Associations
multiples &
réflexives

Classe
association

Agrégation
&
Composition

Héritage

Classe
abstraite &
interface



Héritage

- Principe : Regrouper en structure arborescente des classes partageant une structure et un comportement
- Permet de factoriser des propriétés communes



Héritage

Représentation :



- Relation de **spécialisation / généralisation**
- On parle de **sous-classe** et de **super-classe**, ou de classe enfant et classe parent
- Dire qu'**une sous-classe hérite d'une super-classe** signifie que toutes les instances de la sous-classe hérite de :
 - **Hérite de la structure** (les attributs*) de la super-classe
 - **Hérite du comportement** (les opérations) de la super-classe
 - Possède **ses propres propriétés en plus**

** /!\ visibilité des attributs*



Héritage vs Composition

- Exemple :
- Il y a des voitures bleues, des voitures vertes et des voitures rouges.



Héritage vs Composition

- Exemple :
- Il y a des voitures bleues, des voitures vertes et des voitures rouges.
- Héritage : classe abstraite voiture et 3 classes enfants VoitureBleue, VoitureVerte, VoitureRouge
- Composition : une classe voiture, une classe couleur et une association entre les deux



Chapitre 04

Classes & Associations particulières

Rappel

Associations
multiples &
réflexives

Classe
association

Agrégation
&
Composition

Héritage

Classe
abstraite &
interface



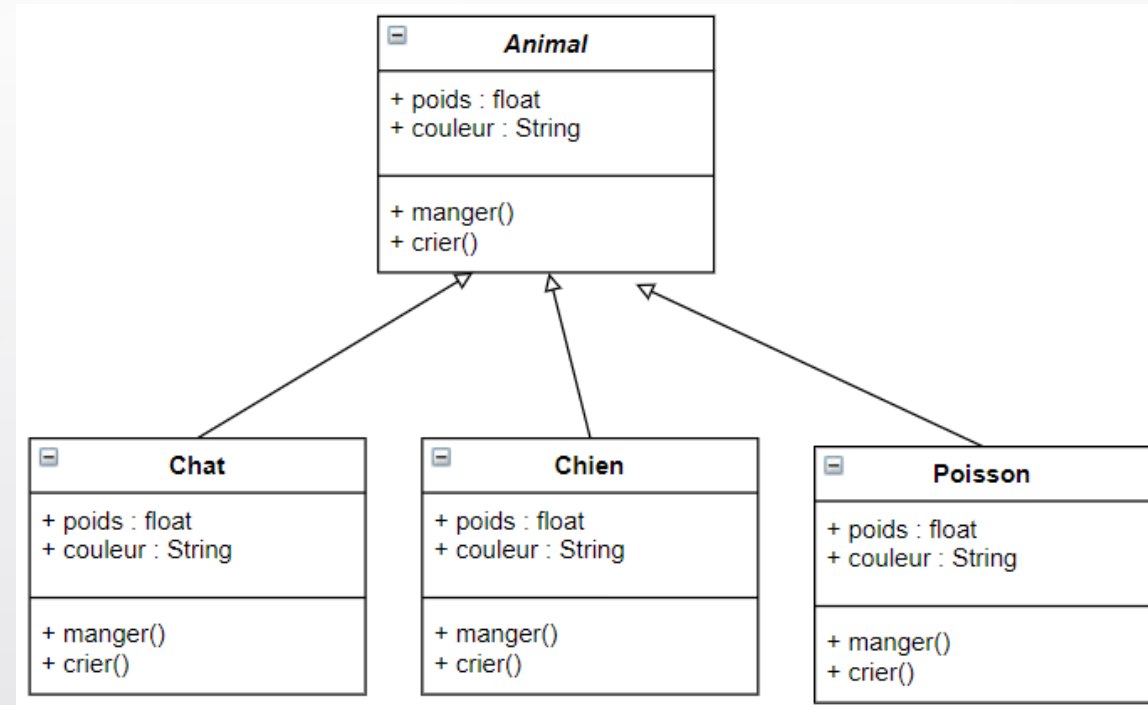
Classe abstraite

- Classe sans **aucune instance**
- Contient des **opérations non définies**
- Doit nécessairement être spécialisée en une (ou plusieurs) classes héritées
- Intérêt :
 - **Spécifier un comportement commun** à plusieurs classes
 - Manipuler des instances de classes différentes de façon uniforme

Classe abstraite

Exemple

NB : s'applique très bien à la modélisation de jeu...



- On définira dans chaque classe enfant les opérations `manger()` et `crier()`
 - « des sardines », « des croquettes », « des vers »
 - « je miaule », « j'aboie », « je ne crie pas vraiment »



Interface

- Principe : **regrouper uniquement les opérations utiles à la classe « cliente »**

Interface =

- Liste d'**opérations abstraites**
- **Sans attribut**
- Dont les opérations doivent être instanciées par des classes héritées



Interface

- **Une classe peut « réaliser » une interface**, i.e. **définir toutes les opérations** abstraites de l'interface. Elle est alors dite « classe cliente de l'interface ».
- Relation de dépendance entre la classe cliente et l'interface requise.

NB

- Une classe peut implémenter plusieurs interfaces (contournement de l'héritage multiple qui est impossible en Java)
- On verra en TD qu'une interface peut permettre de créer un supertype

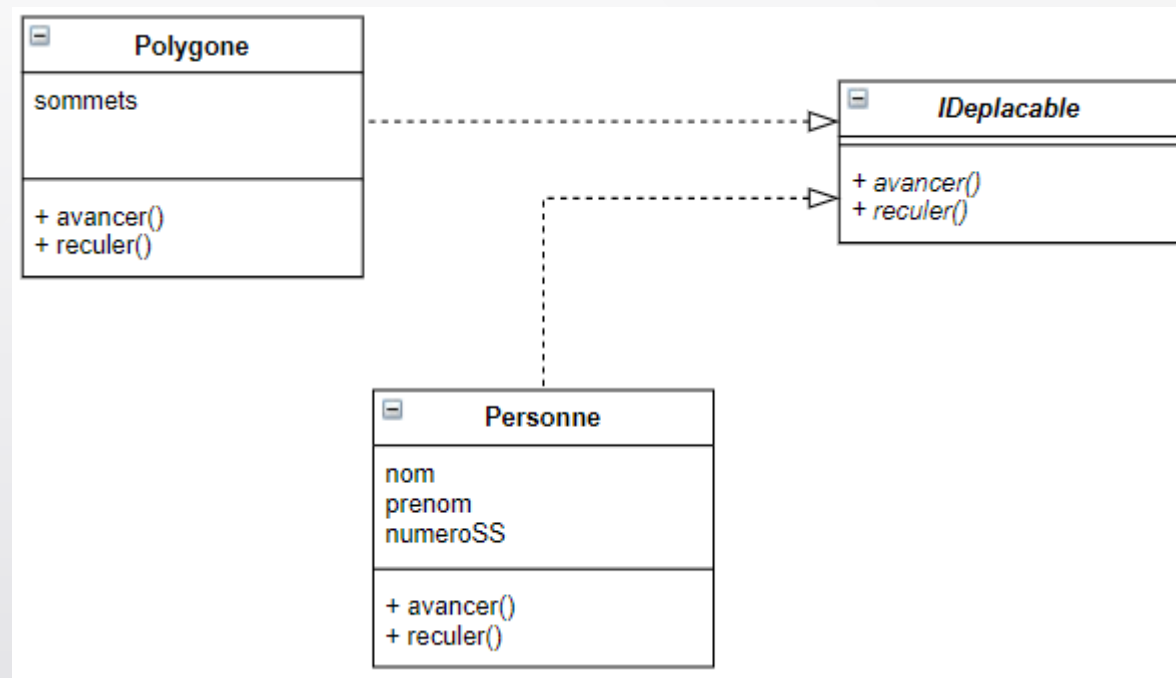
Interface

Représentation



Interface

Exemple



Chapitre 04

Classes & Associations particulières

Synthèse des relations
entre classes

Association
(binaire, multiple, réflexive)

Composition

Agrégation

Généralisation/Spécification

Réalisation d'une interface



Next

Exemples

Mise en
pratique

Projet