

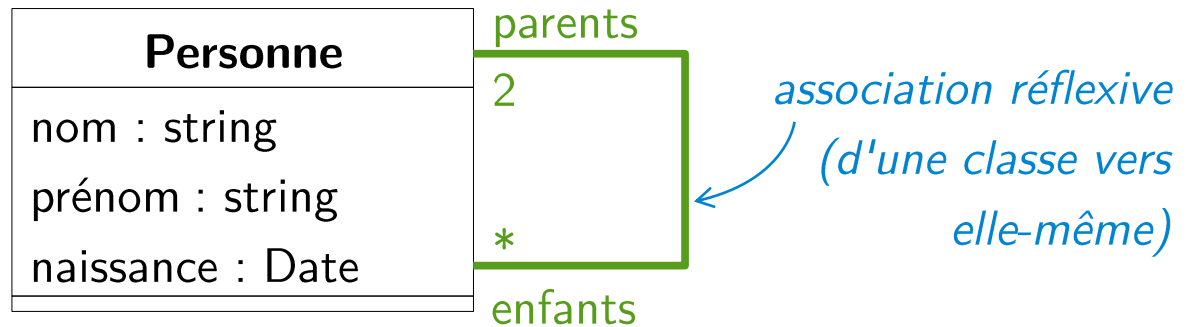
UML

Diagrammes de classes

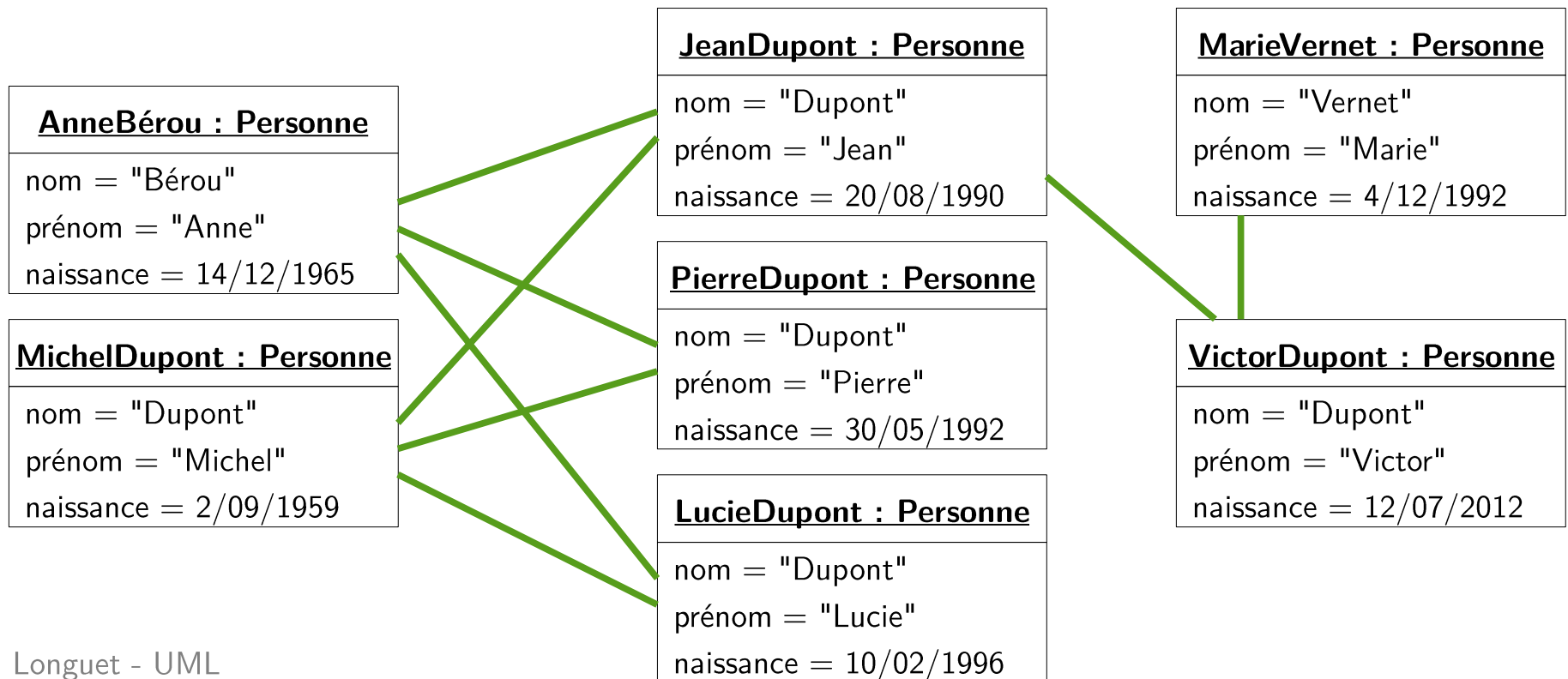
2. Associations particulières, héritage

Association réflexive

Diagramme de classes



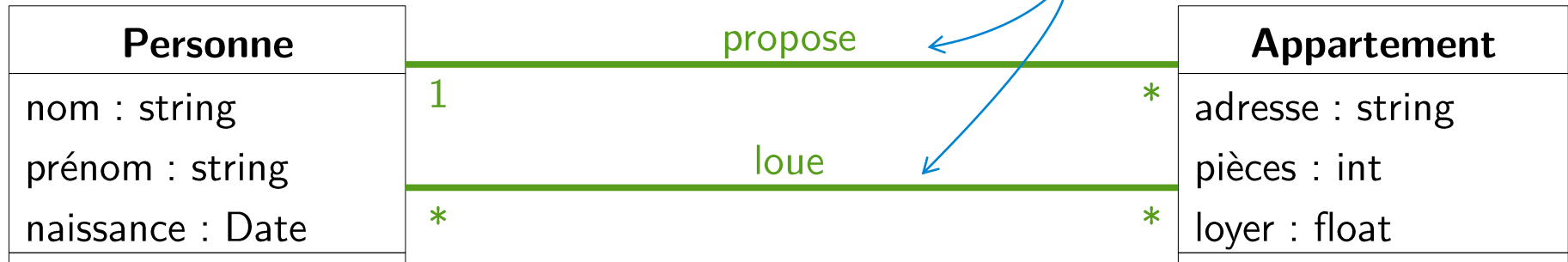
Exemple de diagramme d'objets



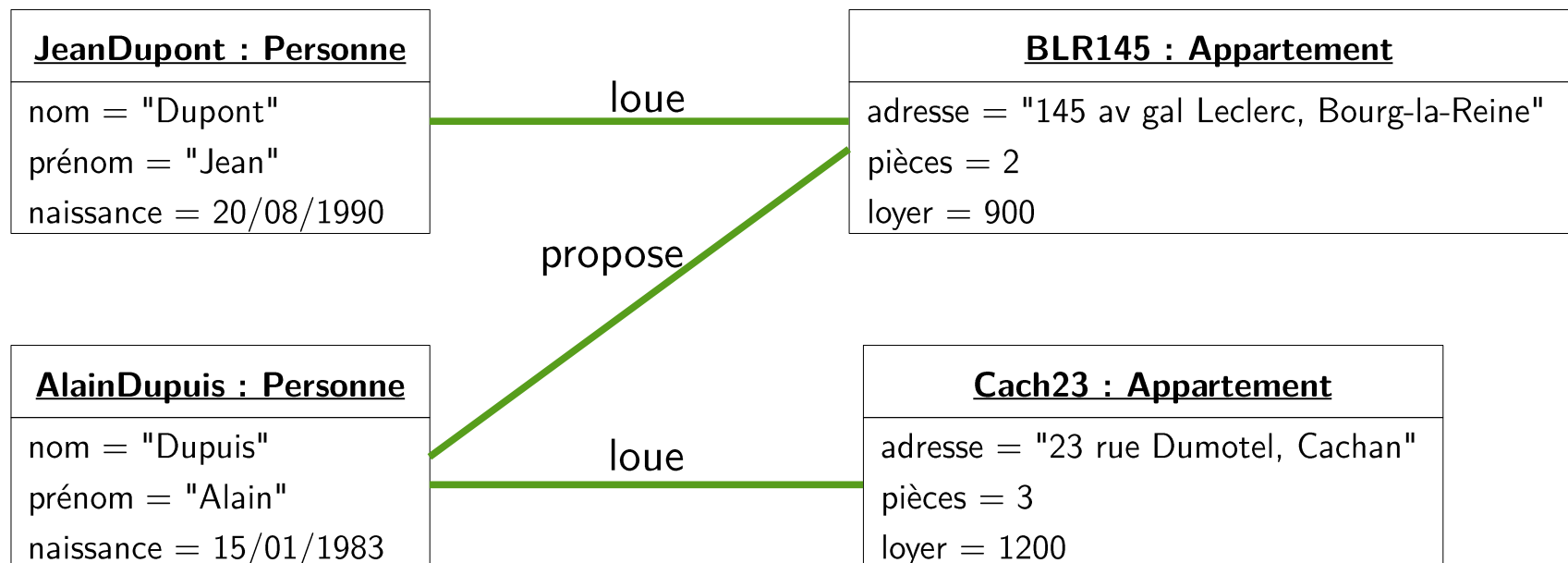
Associations multiples

Diagramme de classes

*plusieurs associations
entre deux classes*

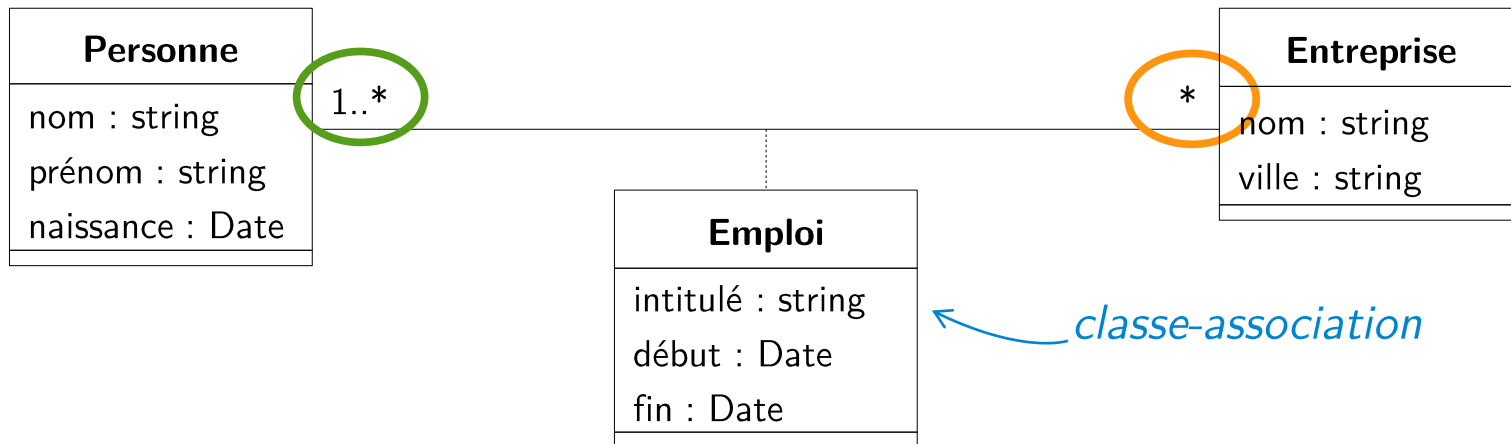


Exemple de diagramme d'objets



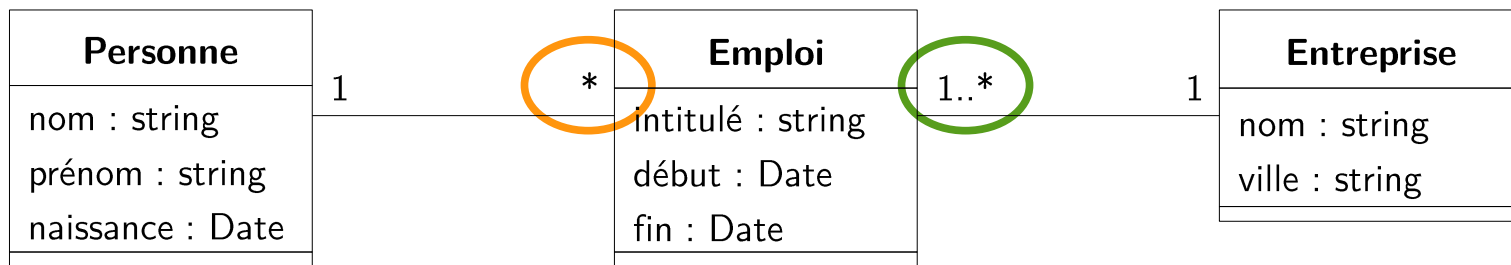
Classe-association

Permet de paramétrer une association entre deux classes par une classe



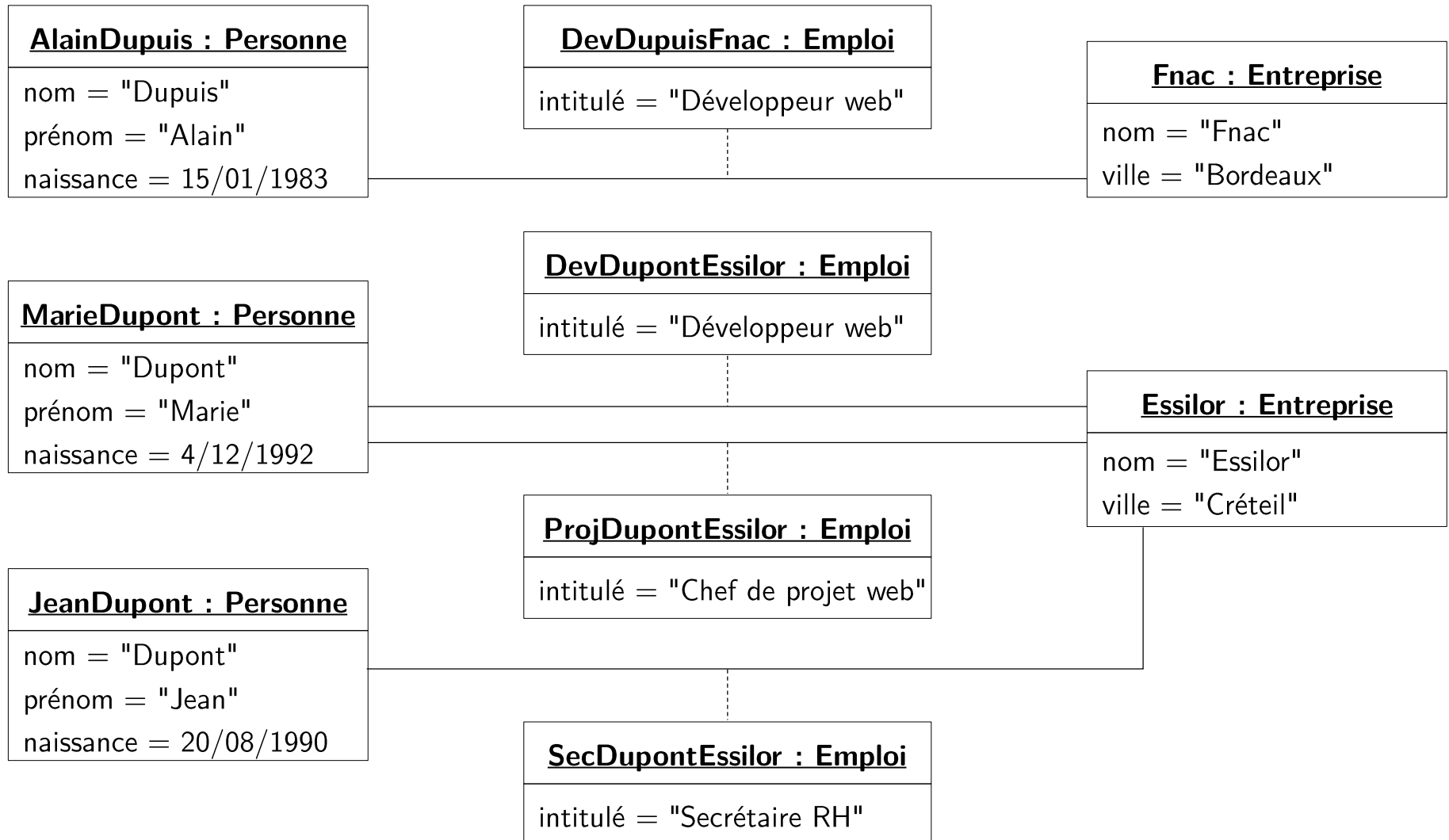
Instance unique de la classe-association pour chaque lien entre objets

Équivalence en termes de classes et d'associations :



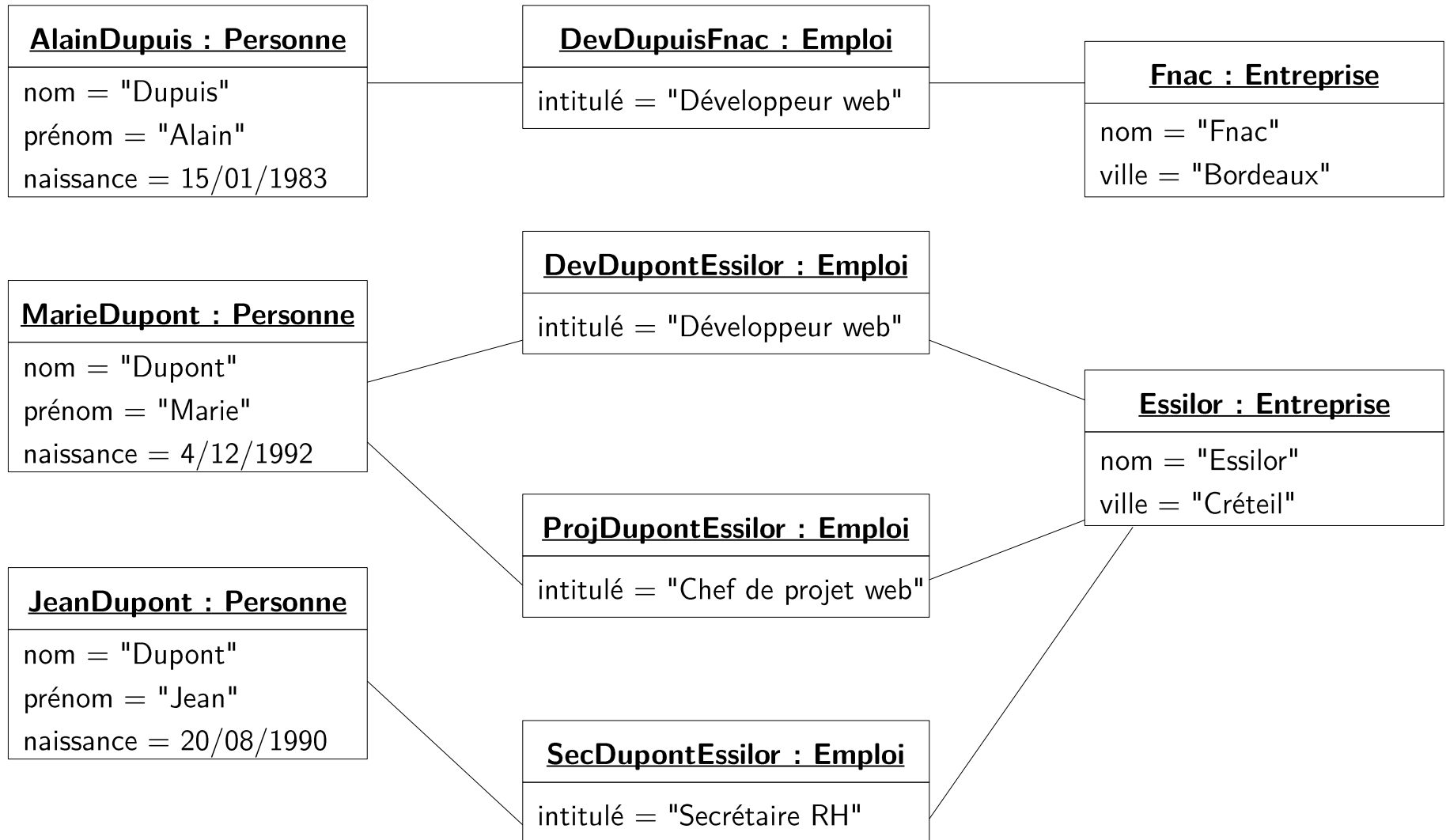
Classe-association

Exemple de diagramme d'objets

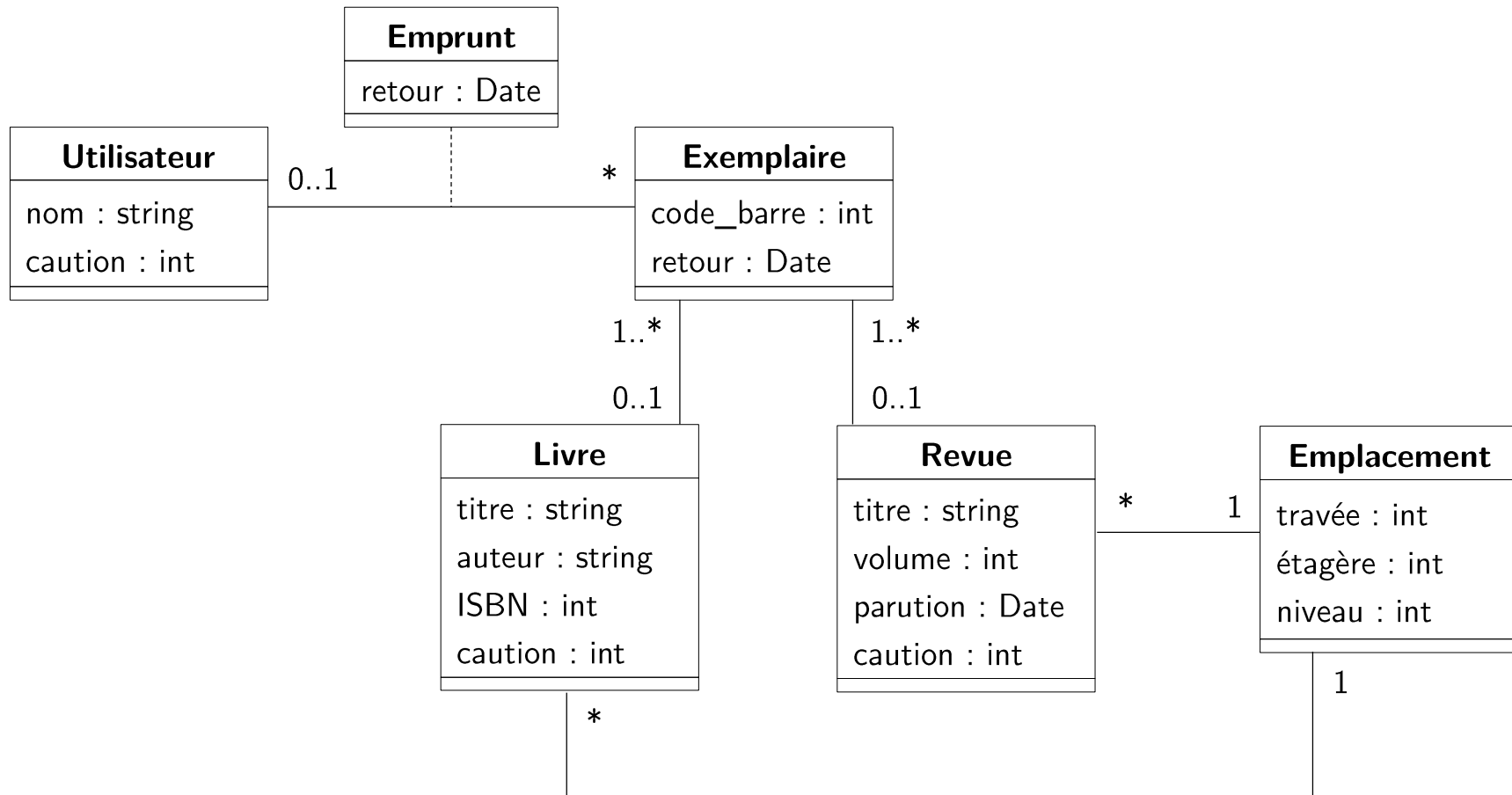


Classe-association

Diagramme d'objets du diagramme de classes équivalent



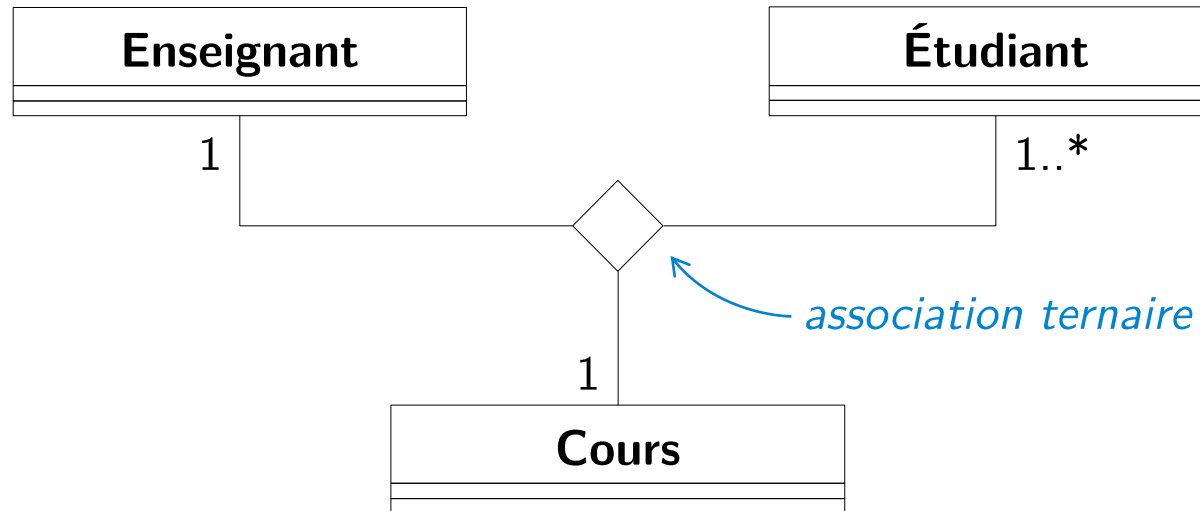
Exemple de la bibliothèque (3)



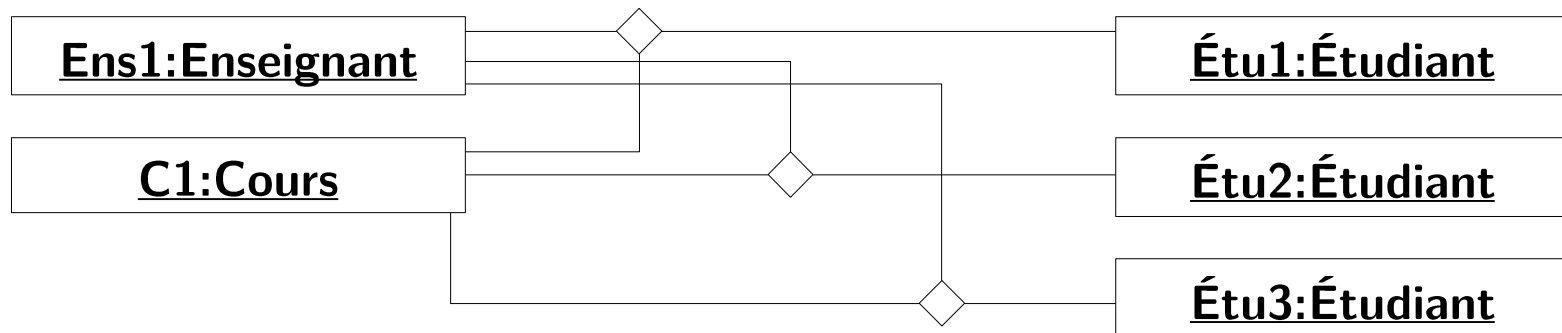
Note : Un exempleire est un exempleire d'un livre ou d'une revue
Si un exempleire n'est pas emprunté, retour a la valeur *null*

Association n -aire

Association reliant plus de deux classes

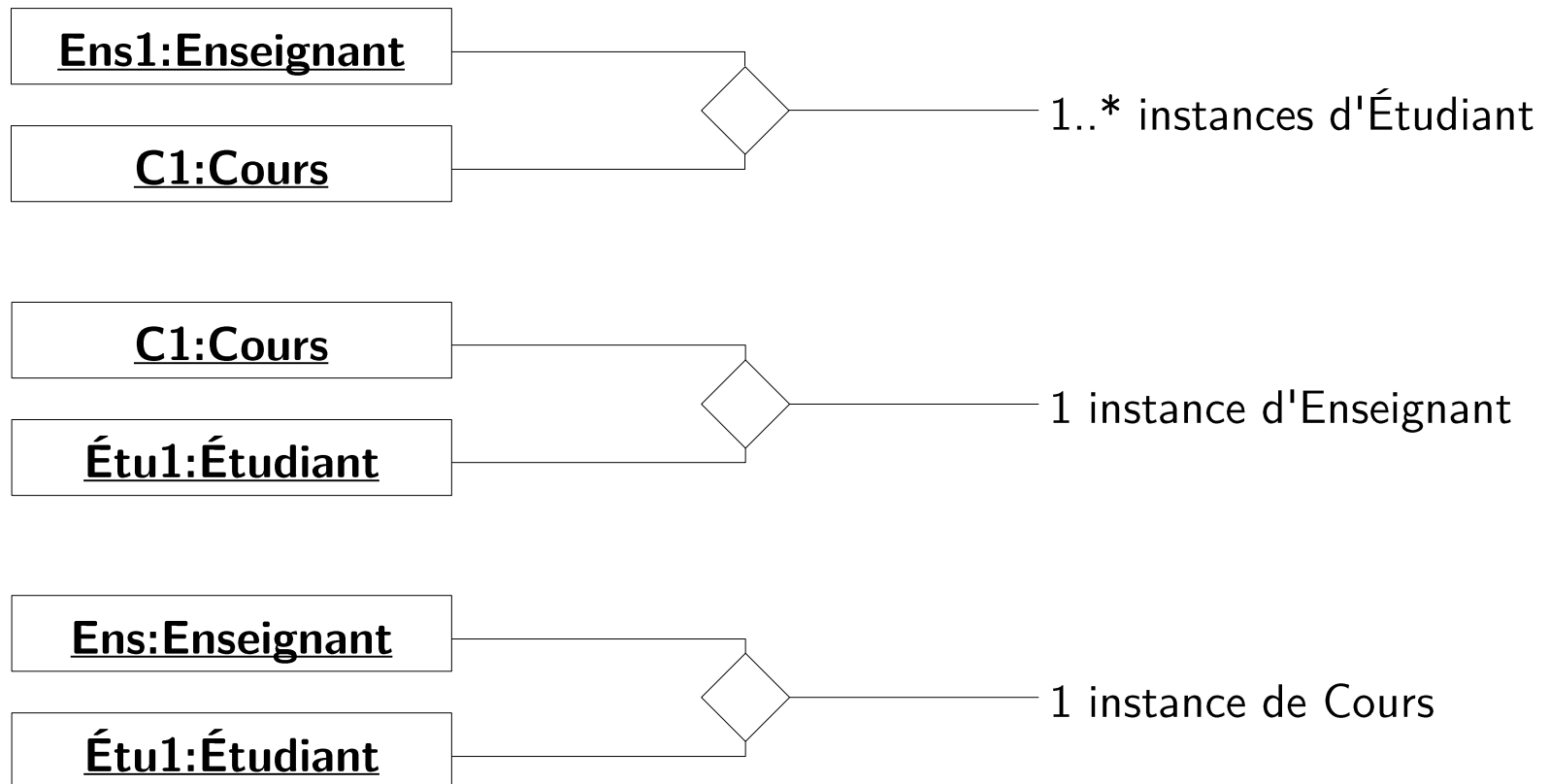


Instance d'une association n -aire = lien entre n objets



Association n -aire

Multiplicités : pour chaque $(n-1)$ -uplet d'objets, contraint le nombre d'objets qui lui sont associés



Hiérarchie de classes

Principe : Regrouper les classes partageant des attributs et des opérations et les organiser en arborescence

Spécialisation : raffinement d'une classe en une sous-classe

Généralisation : abstraction d'un ensemble de classes en super-classe

CompteCourant
numéro : int
devise : Devise
solde : float
découvertAutorisé : float
fraisDécouvert : float
déposer(montant : float)
retirer(montant : float)
solde() : float

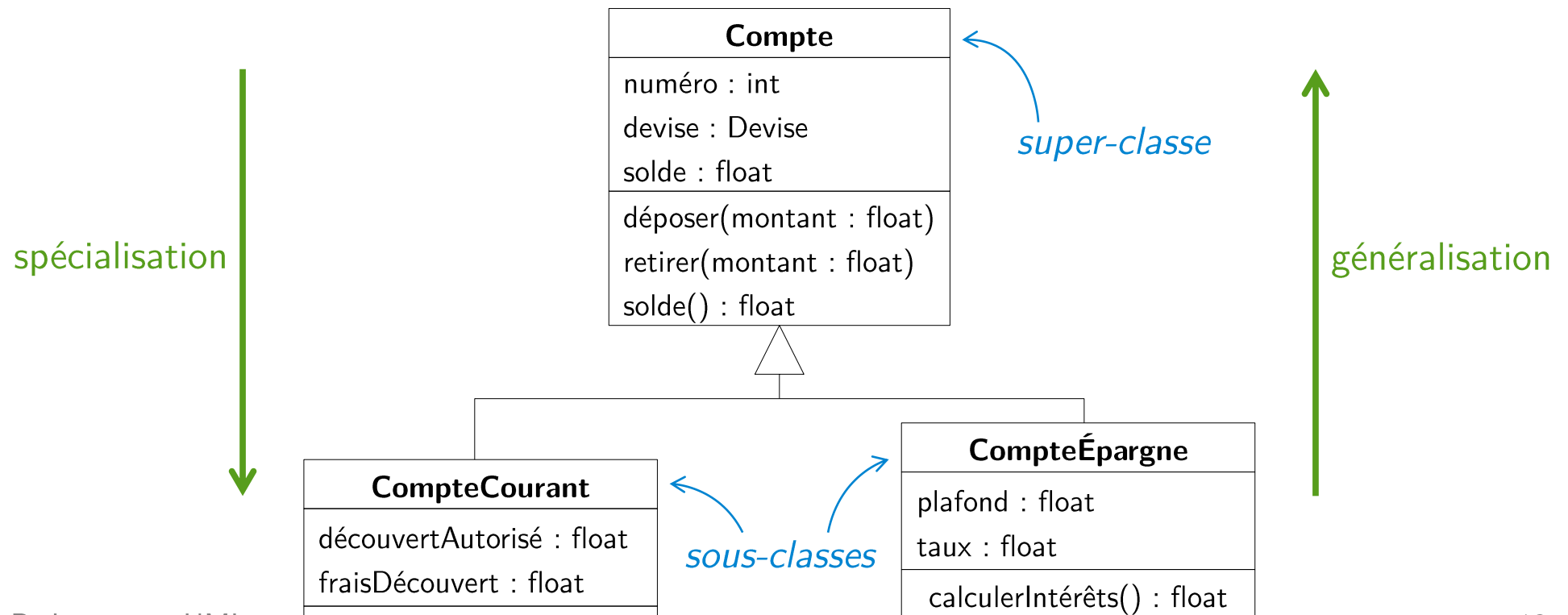
CompteÉpargne
numéro : int
devise : Devise
solde : float
plafond : float
taux : float
déposer (montant : float)
retirer(montant : float)
solde() : float
calculerIntérêts() : float

Hiérarchie de classes

Principe : Regrouper les classes partageant des attributs et des opérations et les organiser en arborescence

Spécialisation : raffinement d'une classe en une sous-classe

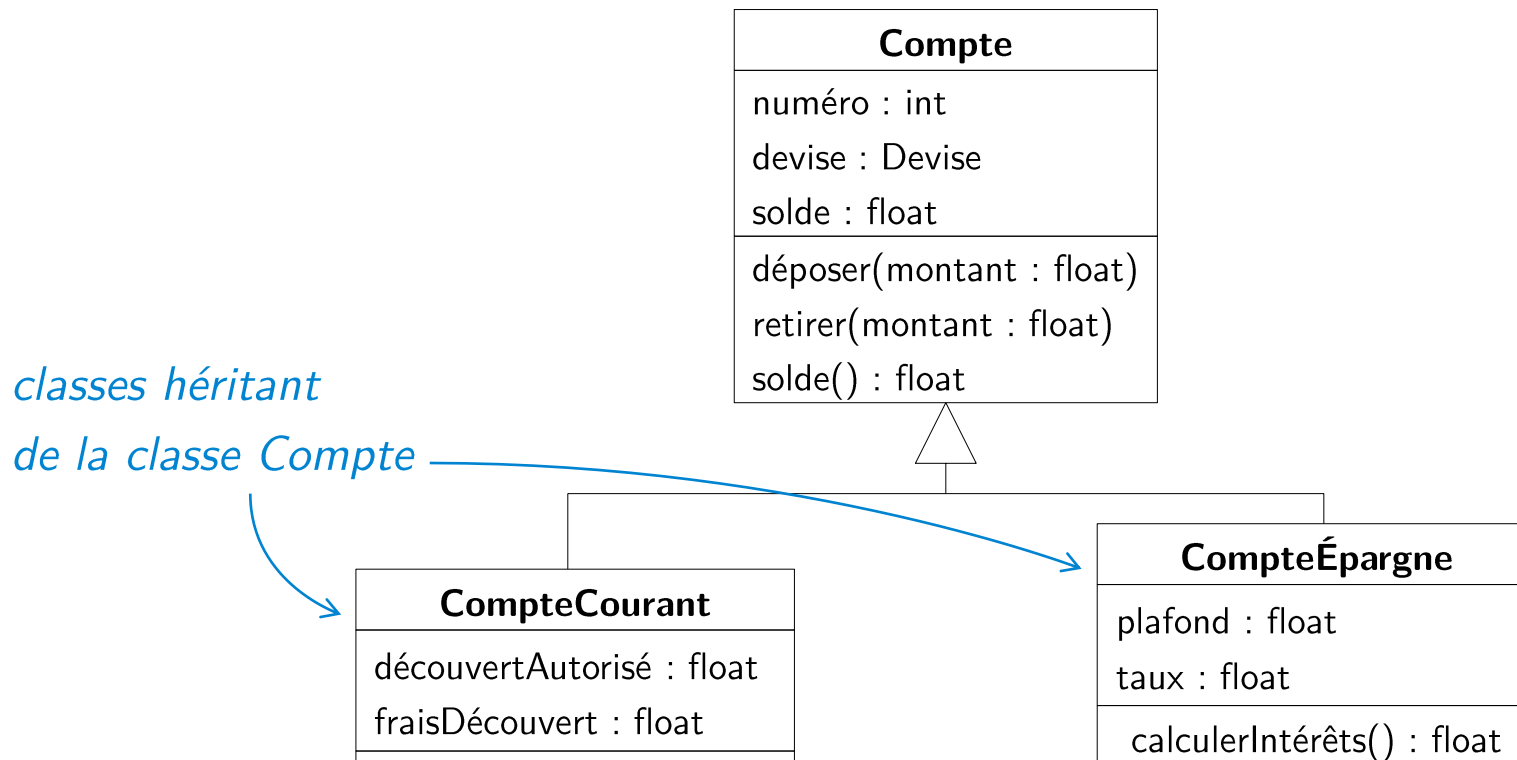
Généralisation : abstraction d'un ensemble de classes en super-classe



Hiérarchie de classes

Principe : Regrouper les classes partageant des attributs et des opérations et les organiser en arborescence

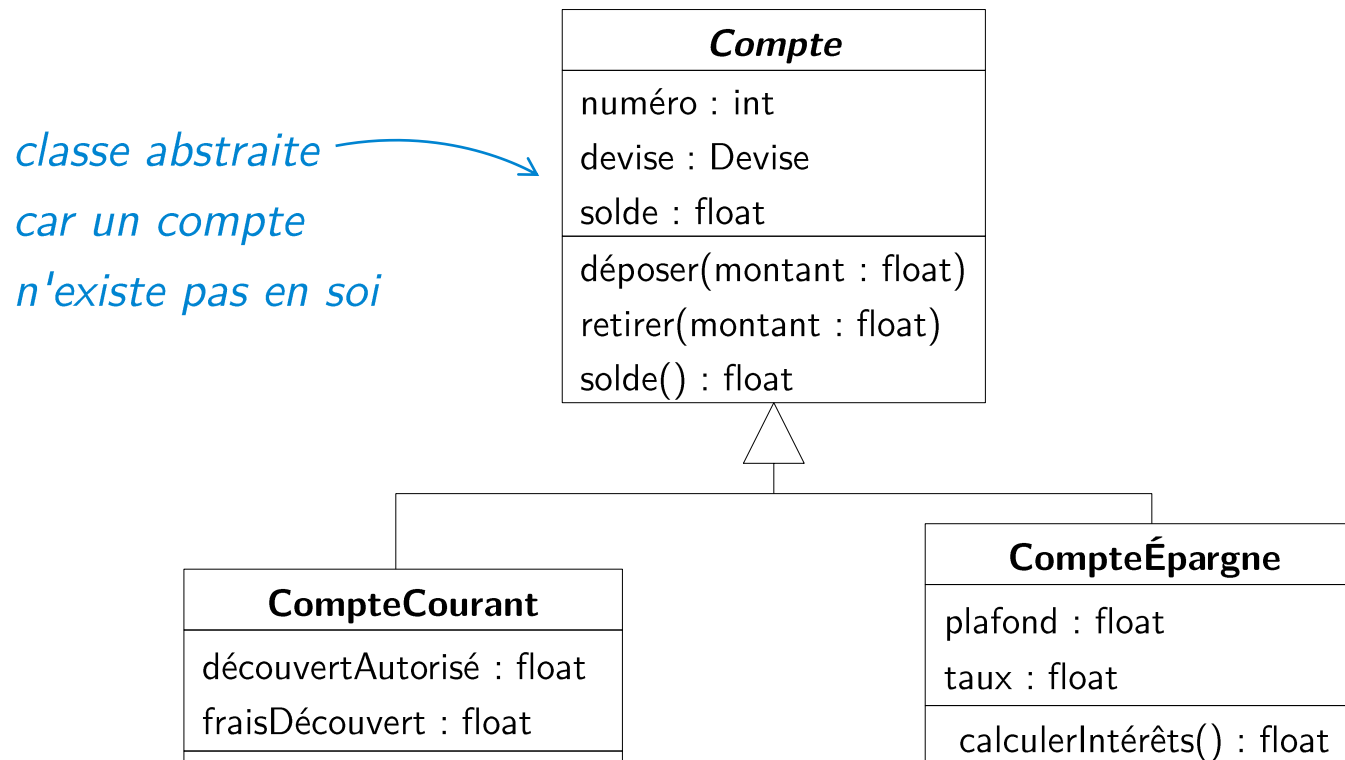
Héritage : Construction d'une classe à partir d'une classe plus haute dans la hiérarchie (partage des attributs, opérations, contraintes...)



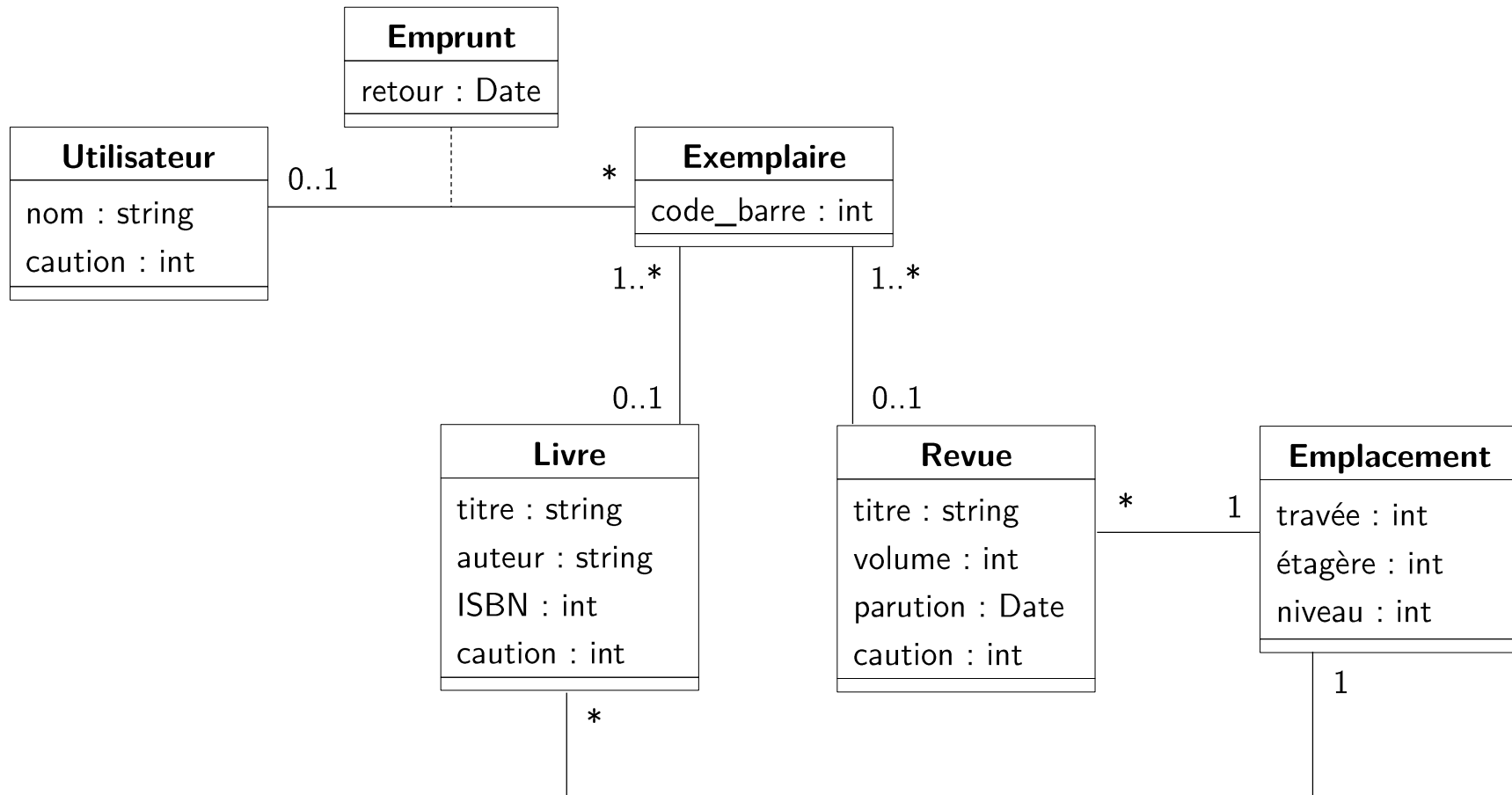
Classe abstraite

Classe sans instance, seulement une base pour classes héritées

Notation : nom de la classe en italique (ou stéréotype « abstract »)

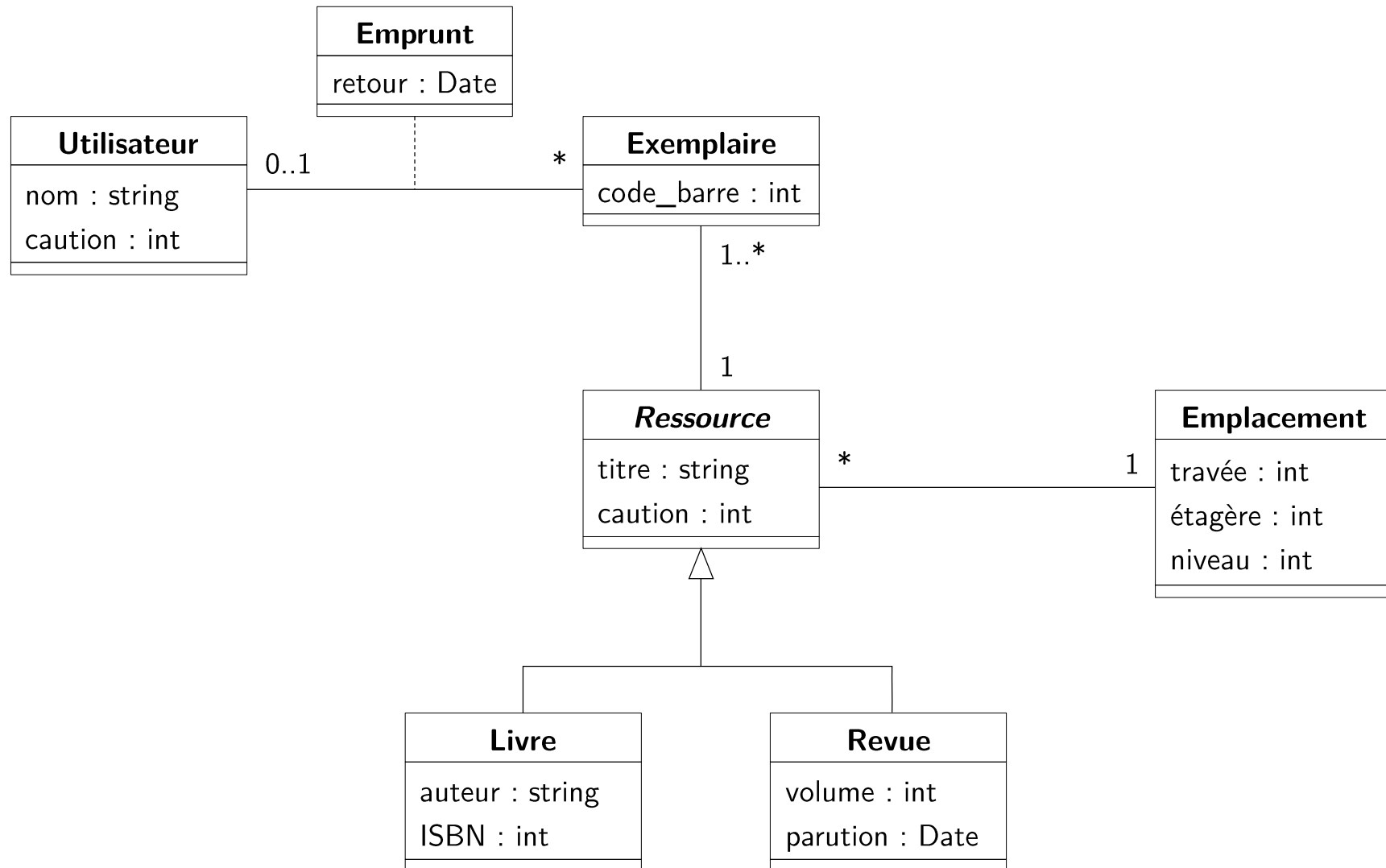


Exemple de la bibliothèque (4)



Note : Un exemplaire est un exemplaire d'un livre ou d'une revue

Exemple de la bibliothèque (4)



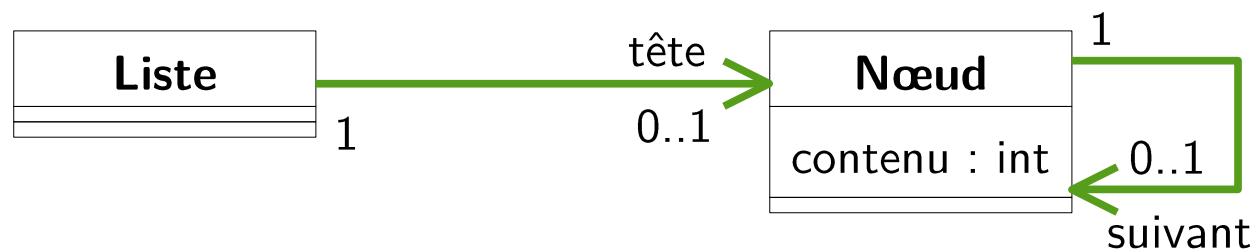
Navigabilité

Orientation d'une association

- Restreint l'accessibilité des objets
- Depuis un A, on a accès aux objets de B qui lui sont associés, mais pas l'inverse



Exemple (listes chaînées)



Par défaut, associations navigables dans les deux sens (pas de flèche)

Agrégation

Association particulière entre classes

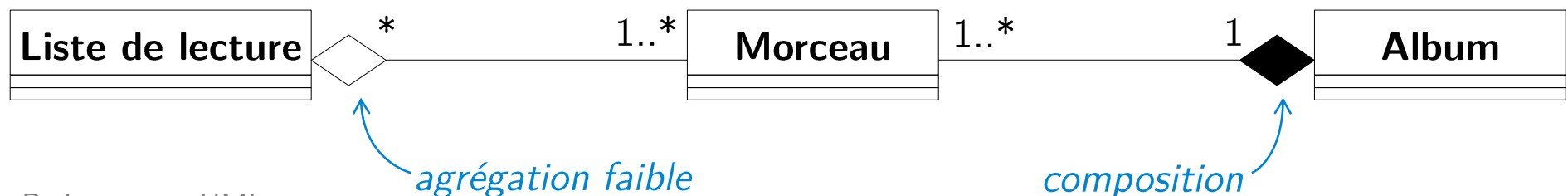
- **Dissymétrique** : une classe prédominante sur l'autre
- Relation de type **composant-composite**

Deux types d'agrégation

- Agrégation faible
- Composition

Exemple

Lecteur de contenu audio permettant de créer des listes de lecture



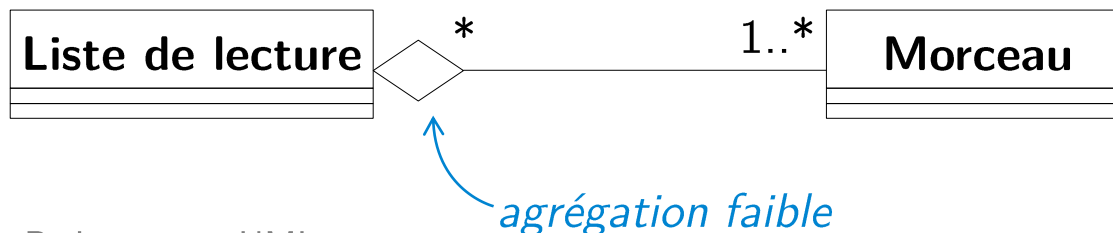
Agrégation faible

Agrégation par référence

- Le composite **fait référence** à ses composants
- La création ou destruction du composite est **indépendante** de la création ou destruction de ses composants
- Un objet peut **faire partie de plusieurs composites** à la fois

Exemple

- Une liste de lecture est composée d'un ensemble de morceaux
- Un morceau peut appartenir à plusieurs listes de lecture
- Supprimer la liste ne supprime pas les morceaux



Composition

Agrégation par valeur

- Le composite **contient** ses composants
- La création ou destruction du composite **entraîne** la création ou destruction de ses composants
- Un objet ne **fait partie que d'un composite** à la fois

Exemple

- Un morceau n'appartient qu'à un album
- La suppression de l'album entraîne la suppression de tous ses morceaux



Exemple de la bibliothèque (5)

