

UML

Leçon 3 : Diagramme de classes

Objets et classes

Conception orientée objet: Représentation du système comme un ensemble d'objets interagissant

Diagramme de classes

- Représentation de la **structure interne** du logiciel
- Utilisé surtout en conception mais peut être utilisé en analyse

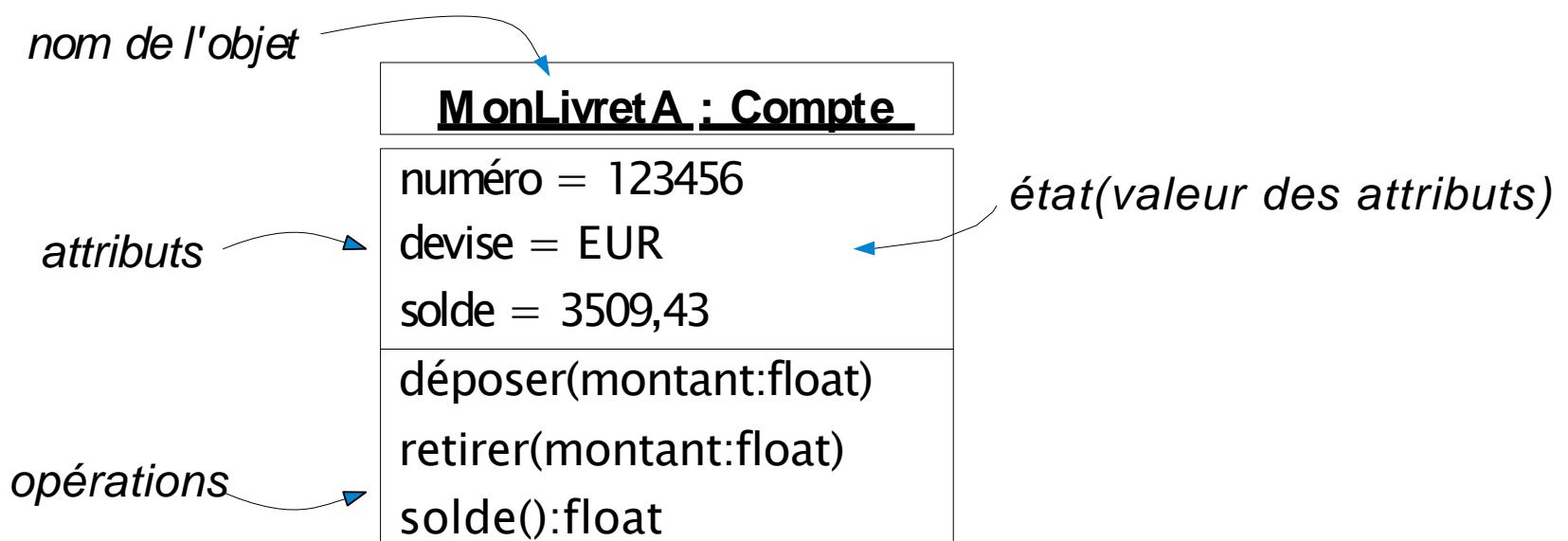
Diagramme d'objets

- Représentation de l'**état** du logiciel (objets + relations)
- Diagramme **évoluant avec l'exécution** du logiciel
 - création et suppression d'objets
 - modification de l'état des objets (valeurs des attributs)
 - modification des relations entre objets

Objets et classes

Objet

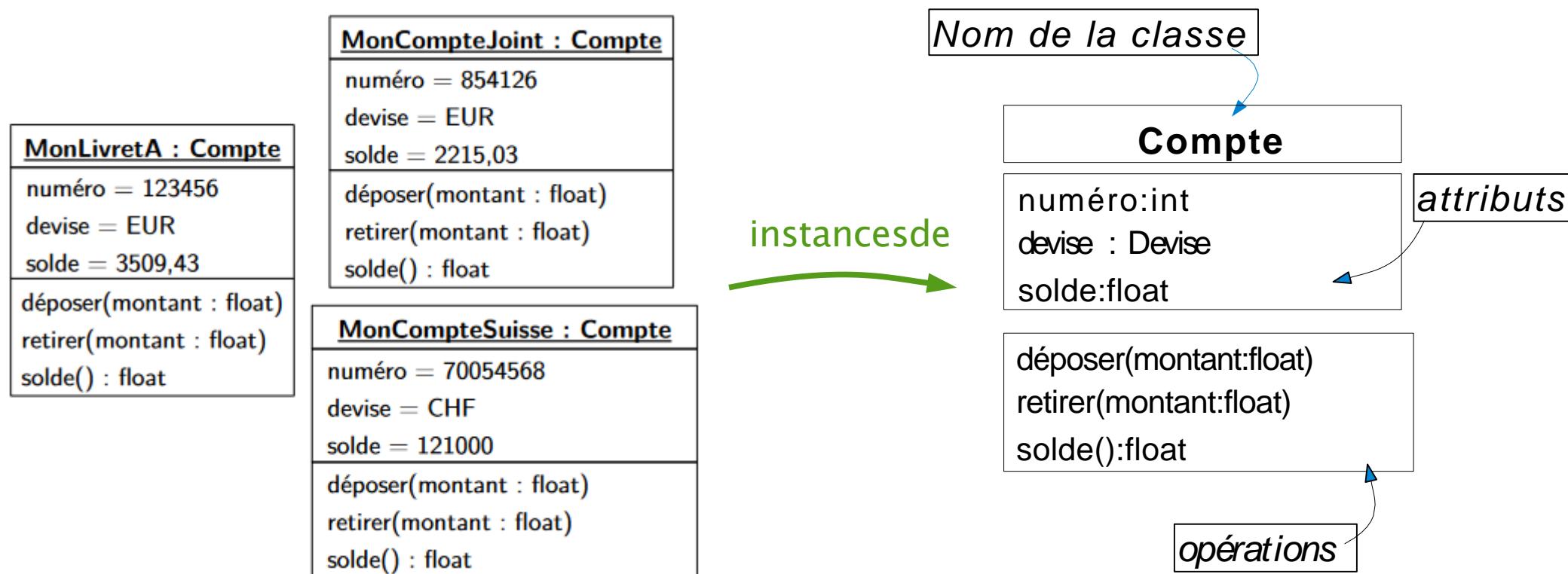
- Entité concrète ou abstraite du domaine d'application
- Décrit par:**identité** (adresse mémoire)
 - + état (attributs)
 - + comportement (opérations)



Objets et classes

Classe: Regroupement d'objets de même nature (mêmes attributs + mêmes opérations)

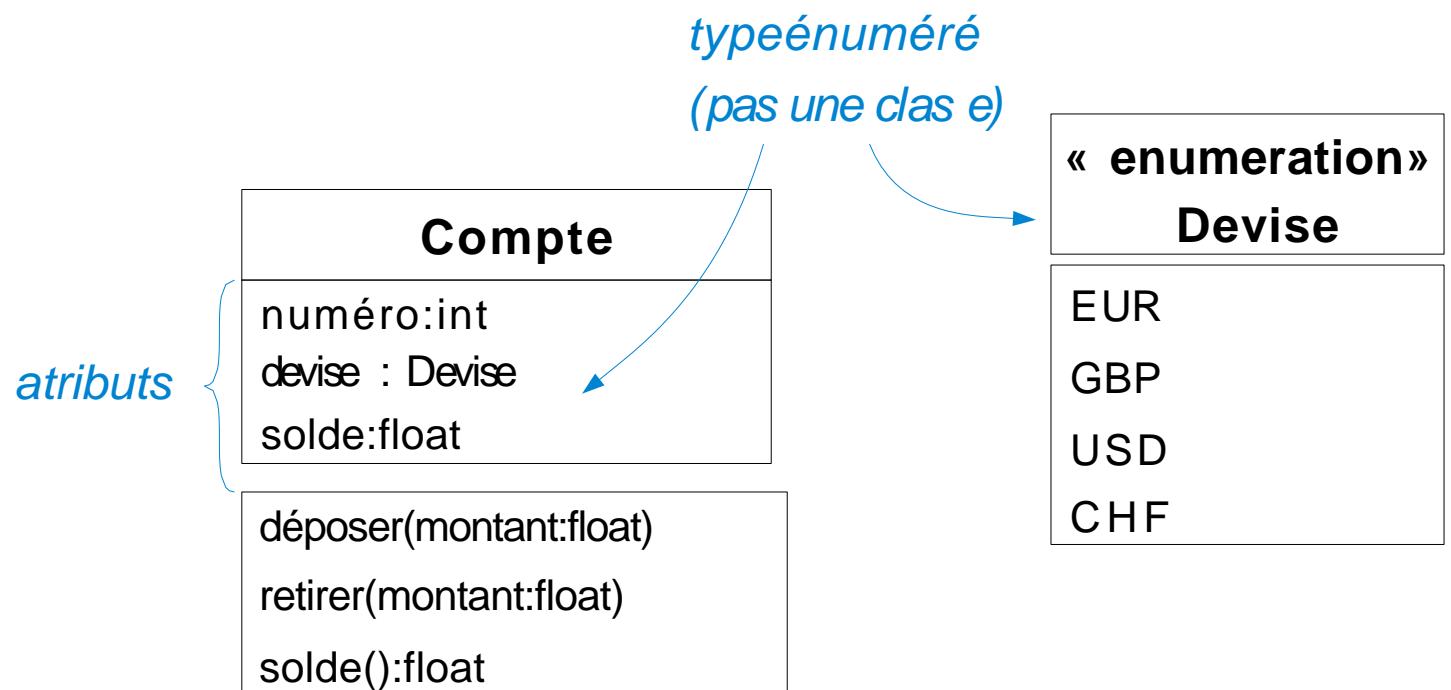
Objet = instance d'une classe



Classes

Atributs

- Caractéristique partagée par tous les objets de la classe
- Associe à chaque objet une valeur
- Type associé simple(int,bol...), primitif (Date) ou énuméré



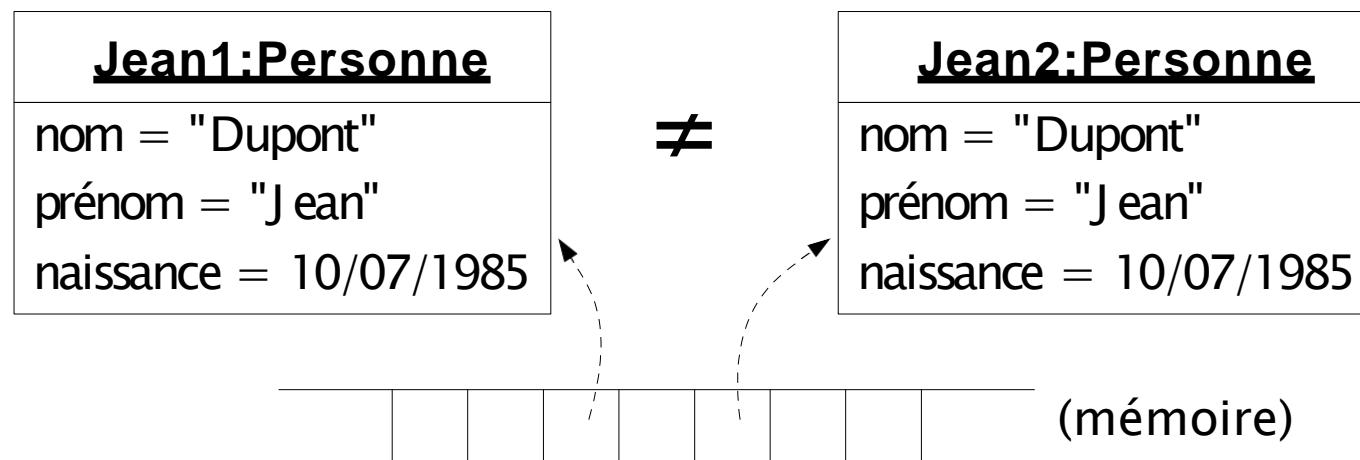
Classes

Attributs

- Caractéristique partagée par tous les objets de la classe
- Associe à chaque objet une valeur
- Type associé simple(int,bol...),primitif(Date)ou énuméré

Valeur des attributs: État de l'objet

- Objets différents (identités différentes) peuvent avoir mêmes attributs



Classes

Opérations

- Service qui peut être demandé à tout objet de la classe
- Comportement commun à tous les objets de la classe



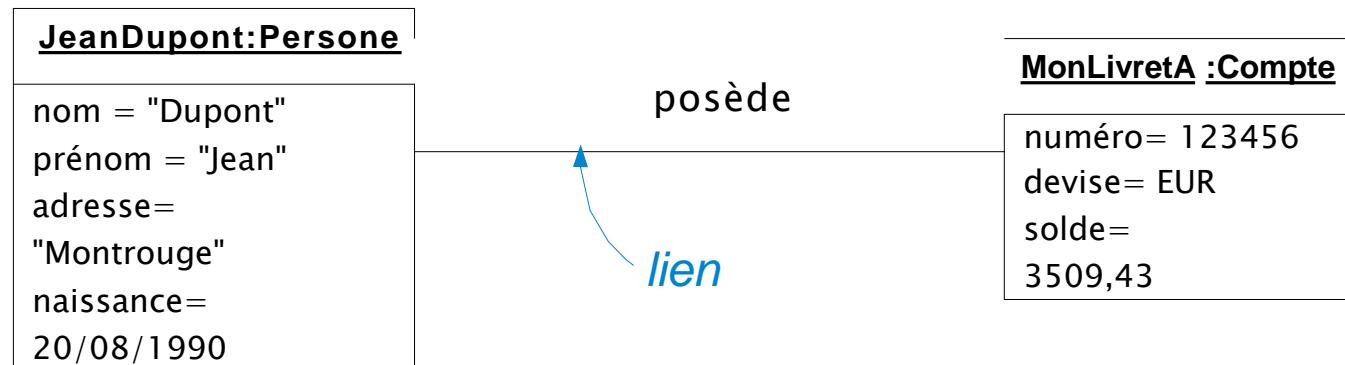
Ne pas confondre avec une méthode=implantation de l'opération

| Compte | |
|------------------------|--|
| numéro:int | |
| devise:Devise | |
| solde:float | |
| <i>opérations</i> | |
| déposer(montant:float) | |
| retirer(montant:float) | |
| solde():float | |

Relations entre objets

Lien entre objets

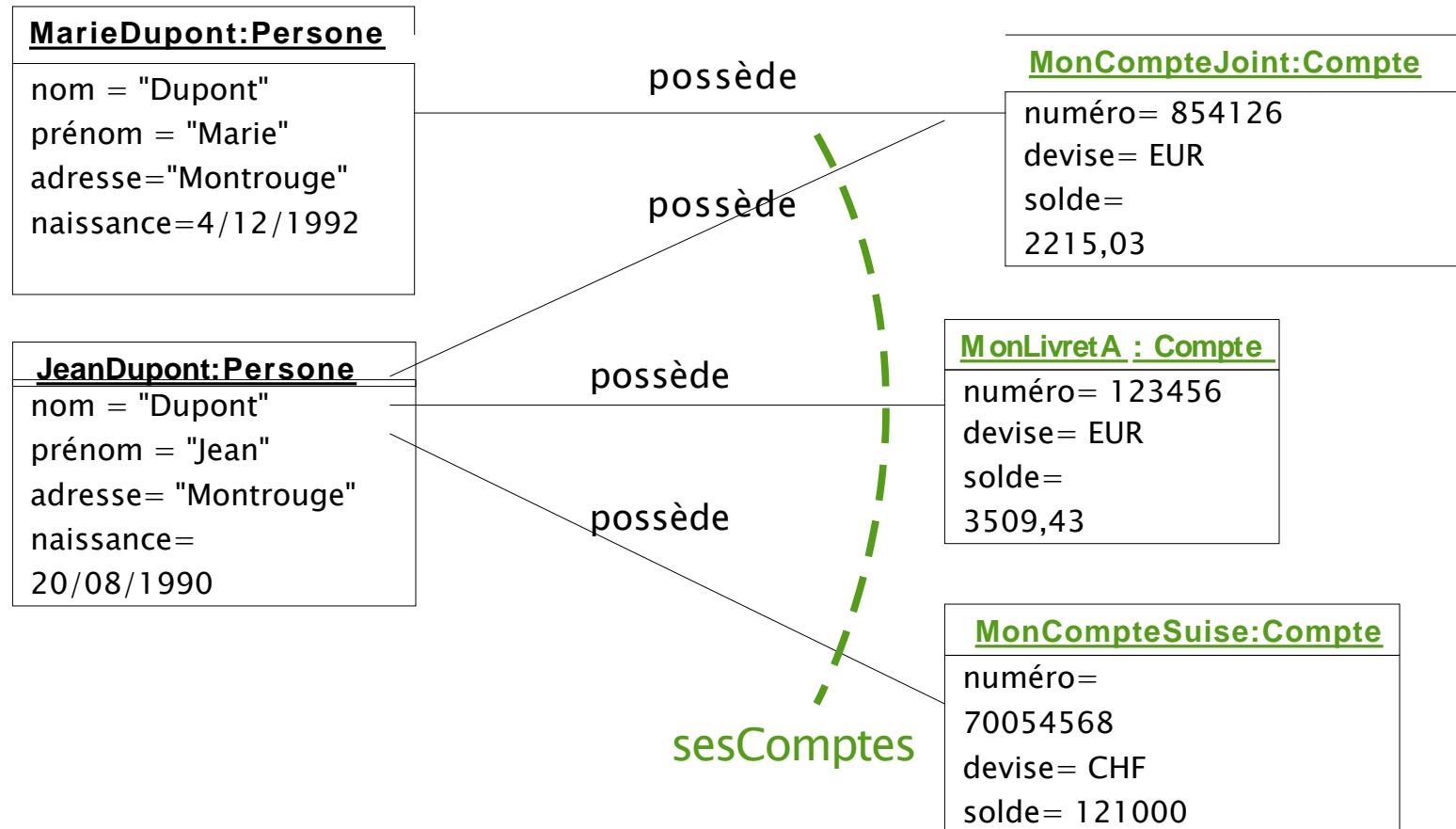
- Relation binaire (en général)
- Au plus un lien entre deux objets (pour une association)



Relations entre objets

Lien entre objets

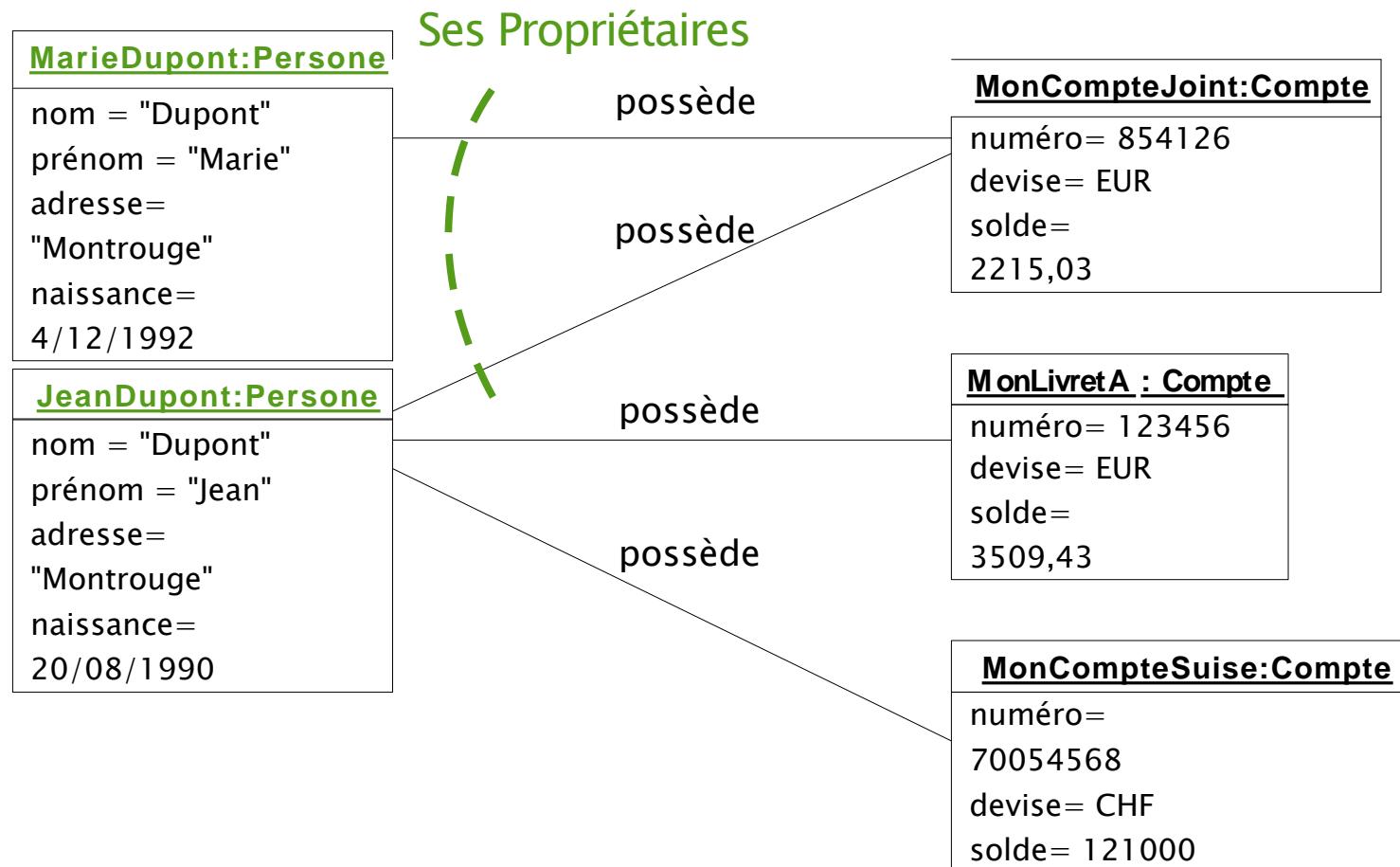
- relation binaire (en général)
- Au plus un lien entre deux objets (pour une association)



Relations entre objets

Lien entre objets

- Relation binaire (en général)
- Au plus un lien entre deux objets (pour une association)

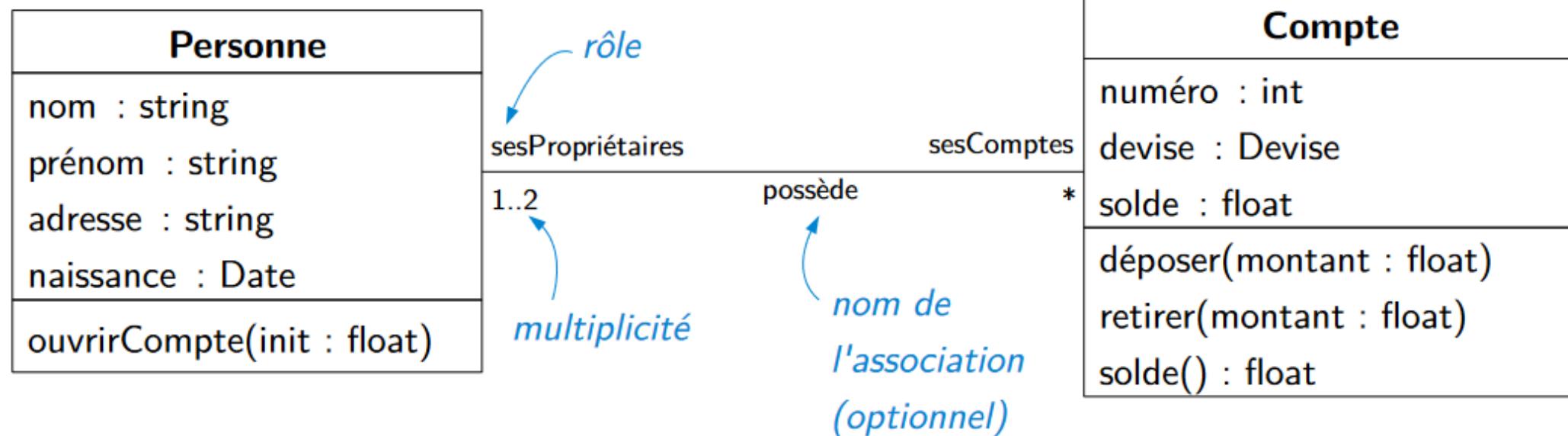


Relations entre classes

Association entre classes: Relation binaire (en général)

Rôle: Nomme l'extrémité d'une association, permet d'accéder aux objets liés par l'association à un objet donné

Multiplicité: Contraint le **nombre d'objets liés** par l'association



Lien = instance d'association

Atribut et association

Rappel: Types des « attributs simple, primitif ou énuméré »

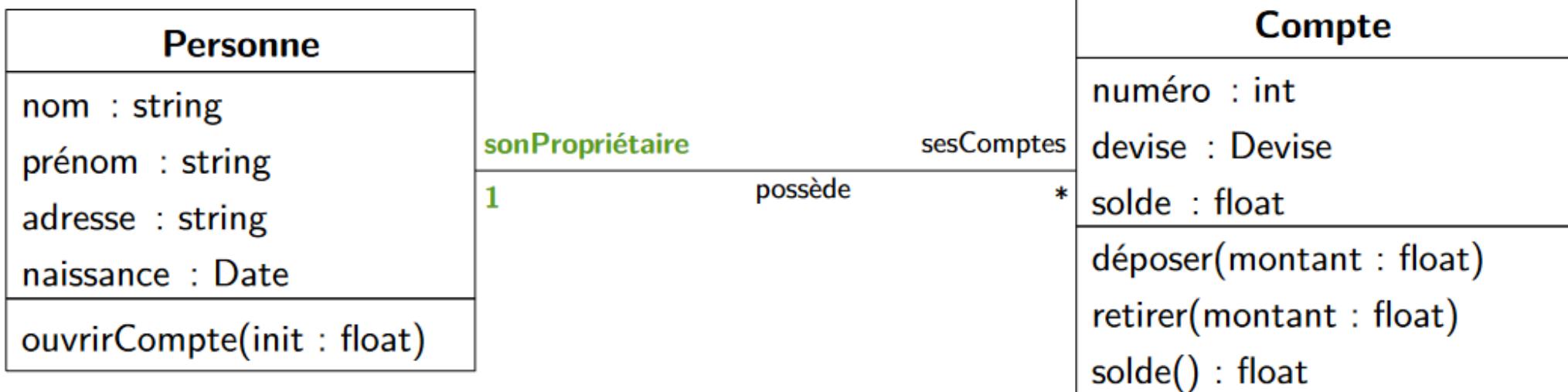
En particulier, pas d'attribut dont le type est une classe du diagramme

| Compte |
|--|
| numéro:int |
| devise : Devise |
| solde:float |
| propriétaire:  Personne |
| déposer(montant:float) |
| retirer(montant:float) |
| solde():float |

Atribut et association

Rappel: Types des attributs « simple, primitif ou énuméré»

En particulier, pas d'attribut dont le type est une classe du diagramme, mais association vers cette classe



Multiplicités

Nombre d'objets de la classe B associés à un objet de la classe A

Exactement n



En pratique

Exactement n ou m ou p



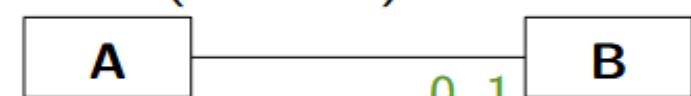
Exactement 1



Entre n et m



Au plus 1 (0 ou 1)



Au moins n



Au moins 1 (jamais 0)



Plusieurs (0 ou plus)



0 ou plus



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 |

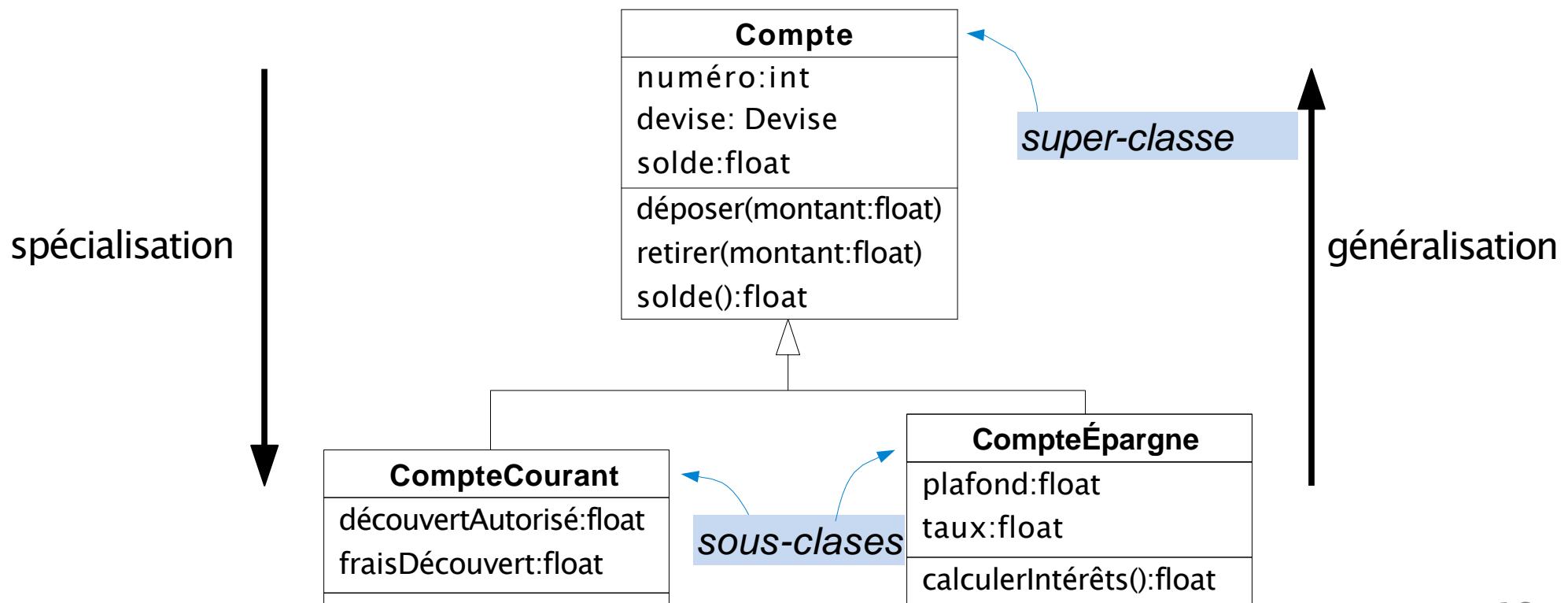
| CompteÉpargne |
|--------------------------|
| numéro:int |
| devise:Devise |
| solde:float |
| plafond:float |
| taux:float |
| déposer(montant:float) |
| retirer(montant:float) |
| solde():float |
| calculerIntérêts():float |

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

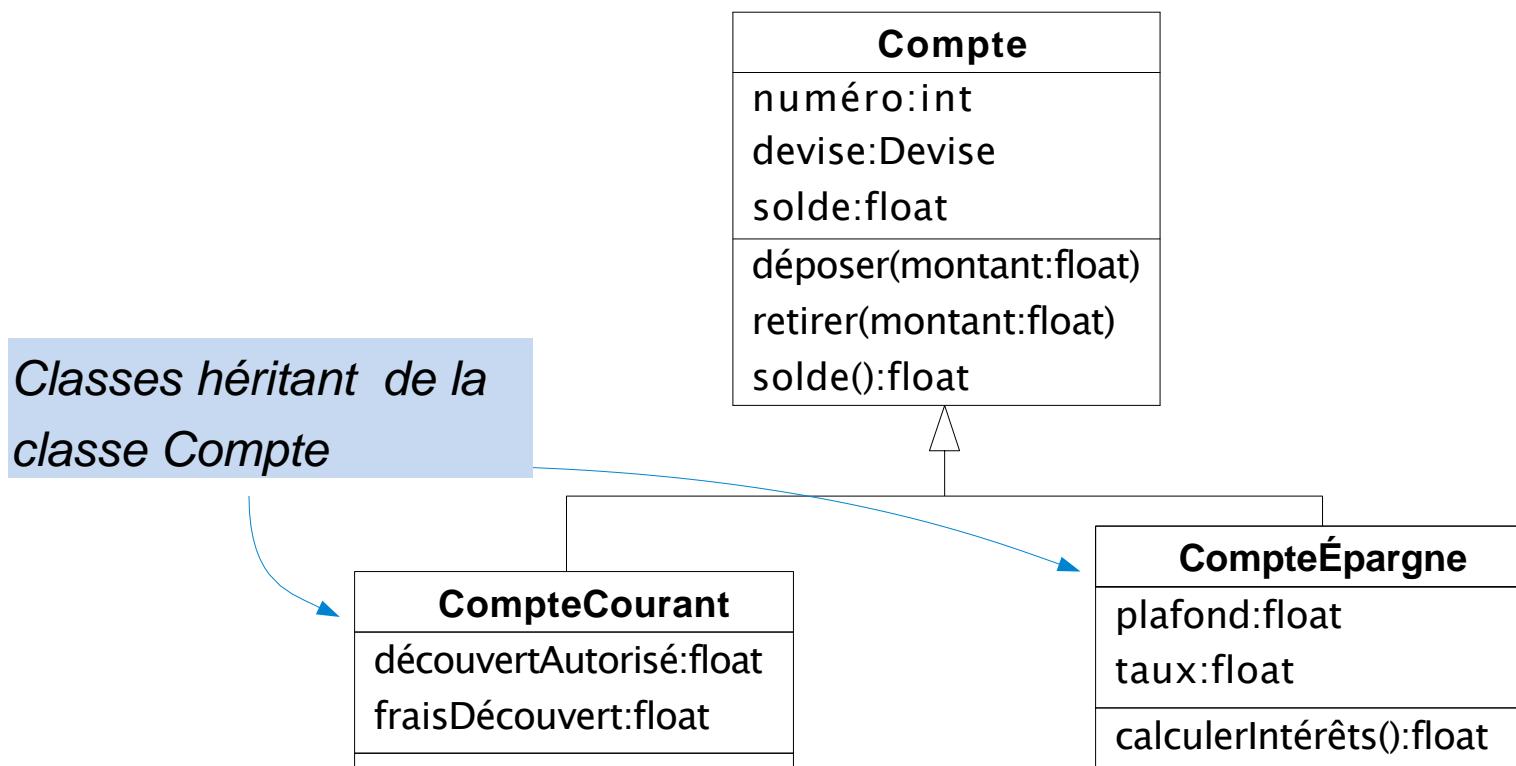
Généralisation: abstraction d'un ensemble de classes en super-classe



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...)



Hiérarchie de classes

Diagramme de classes

| 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 |

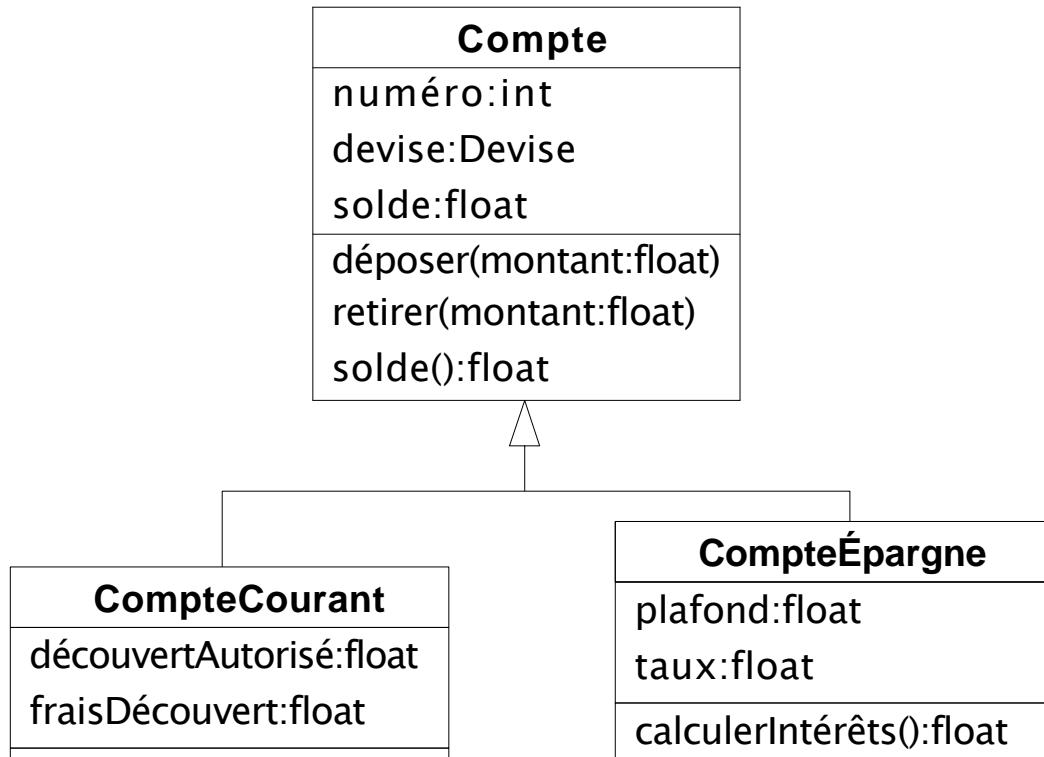
Exemples d'objets

| CC:CompteCourant |
|--------------------------|
| numéro= 875421 |
| devise= EUR solde= |
| 1290,30 |
| découvertAutorisé= - |
| 200,0 fraisDécouvert= |
| 2,30 |

| LivA:CompteÉpargne |
|--------------------|
| numéro=094435 |
| devise= EUR |
| solde= 10542,0 |
| plafond= 2950,0 |
| taux= 0,75 |

Hiérarchie de classes

Diagramme de classes



Exemples d'objets

| <u>C1:Compte</u> |
|------------------|
| numéro = 463527 |
| devise = EUR |
| solde = 213,50 |

| <u>CC:CompteCourant</u> |
|----------------------------|
| numéro = 875421 |
| devise = EUR |
| solde = 1290,30 |
| découvertAutorisé = -200,0 |
| fraisDécouvert = 2,30 |

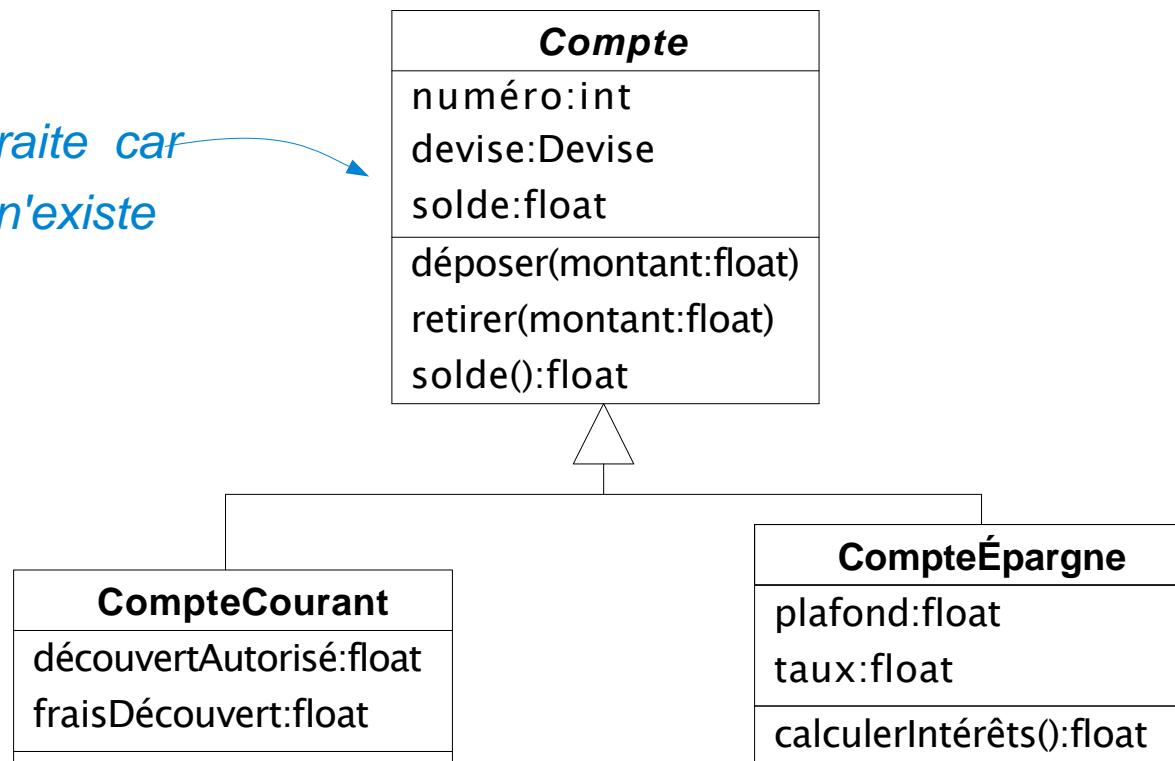
| <u>LivA:CompteEpargne</u> |
|---------------------------|
| numéro=094435 |
| devise = EUR |
| solde = 10542,0 |
| plafond = 2 950,0 |
| taux = 0,75 |

Classe abstraite

Clases sans instance, seulement une base pour classes héritées

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

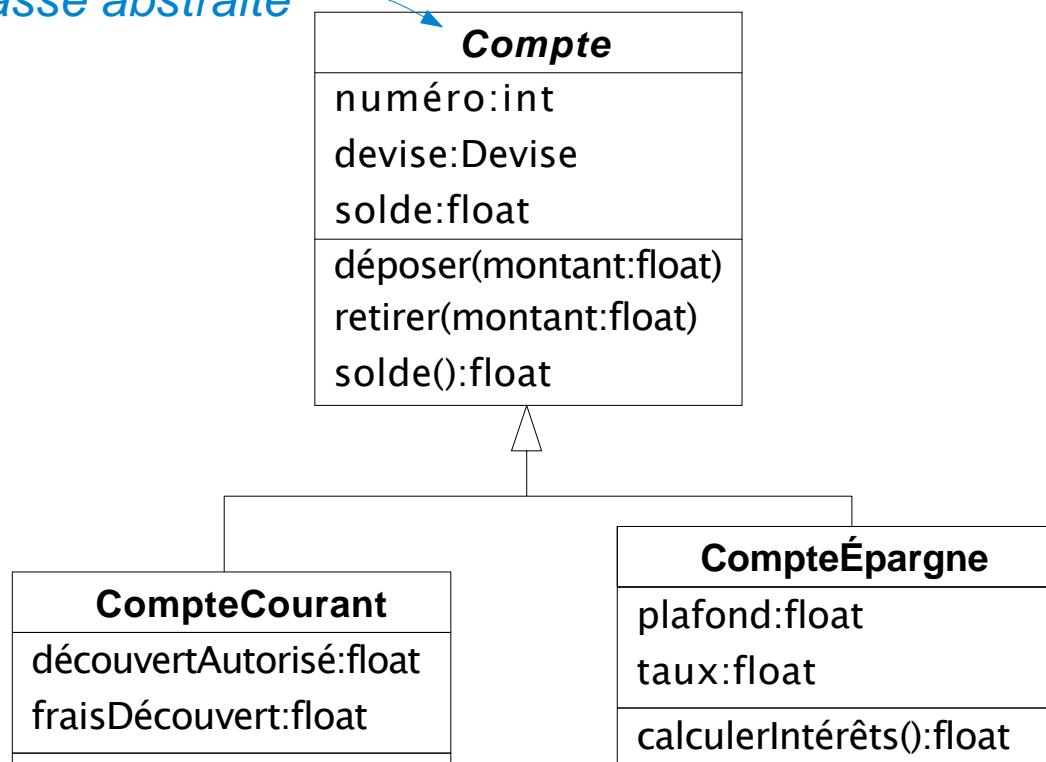
Classe abstraite car
un compte n'existe
pas en soi



Hiérarchie de classes

Diagramme de classes

Classe abstraite

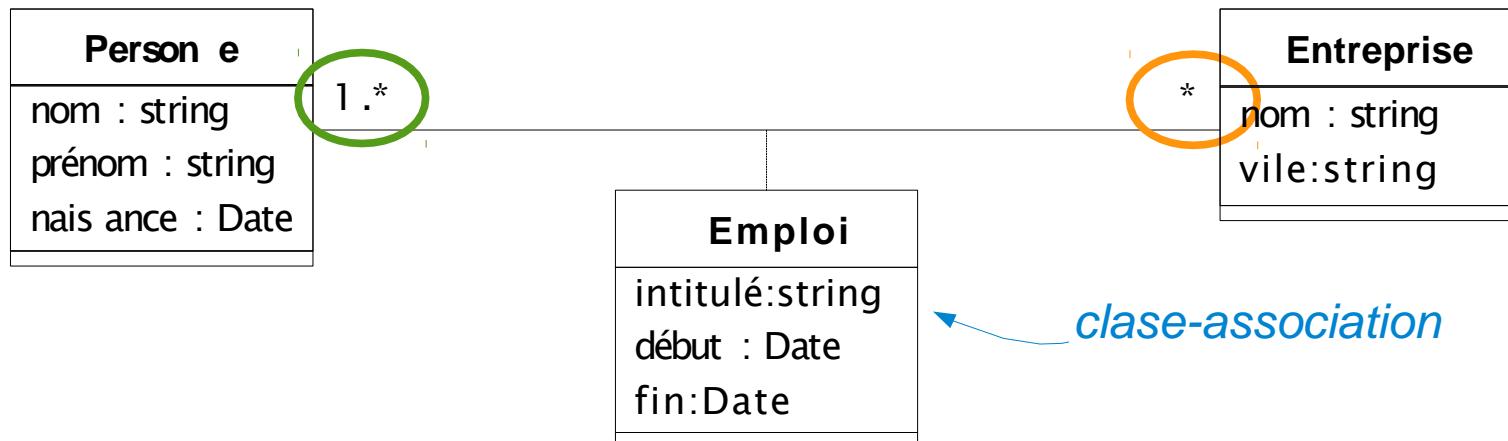


Exemples d'objets

| |
|----------------------------|
| C1:Compte |
| numéro= 463527 |
| devise= EUR |
| solde= |
| 213,50 |
| CC:CompteCourant |
| numéro= 875421 |
| devise= EUR solde= |
| 1290,30 |
| découvertAutorisé= - |
| 200,0 fraisDécouvert= |
| 2,30 |
| LivA:CompteEpargne |
| numéro=094435 |
| devise= EUR |
| solde= 10542,0 |
| plafond= 2950,0 |
| taux= 0,75 |

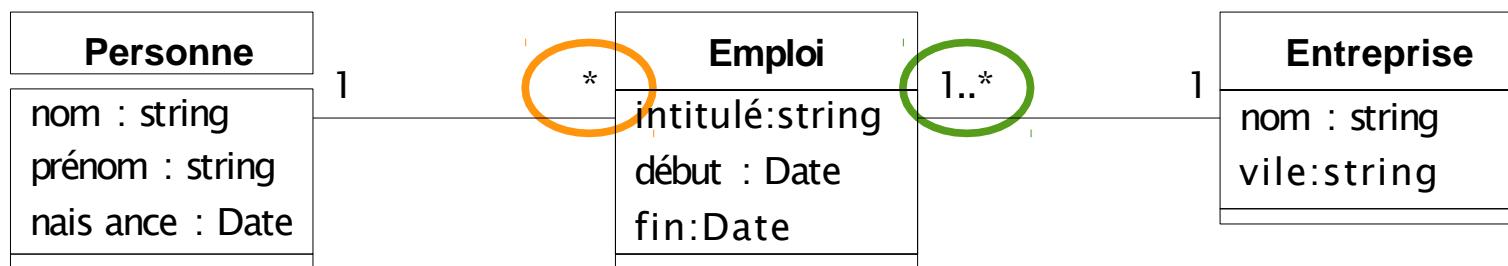
Classe-association

Permet de paramétrer une association entre deux classes par une classe



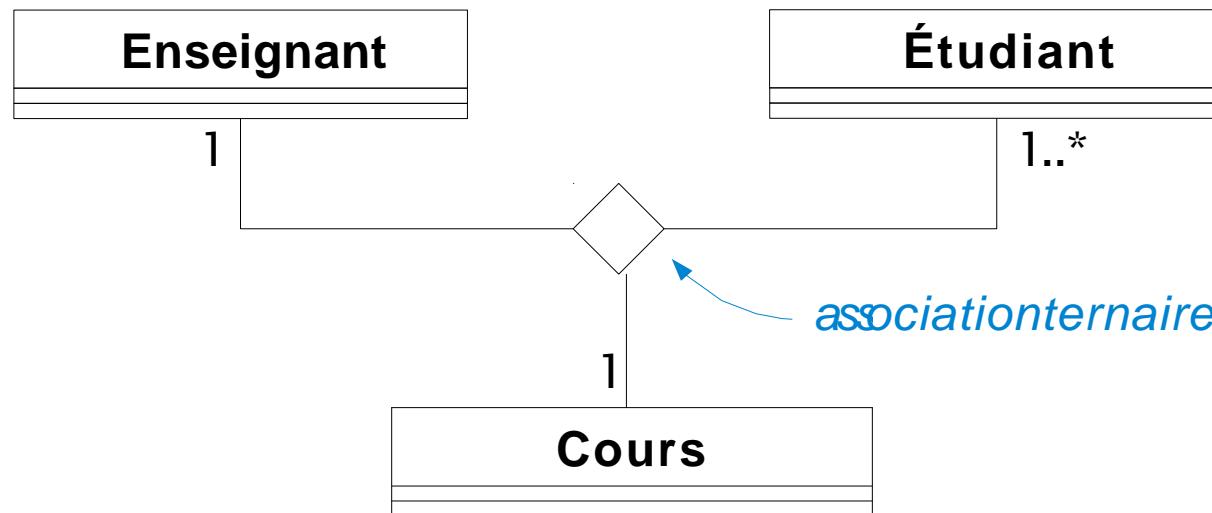
Instance unique de la classe-association pour chaque lien entre objets

Équivalence en termes de classes et associations:



Association n -aire

Association reliant plus de deux classes



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



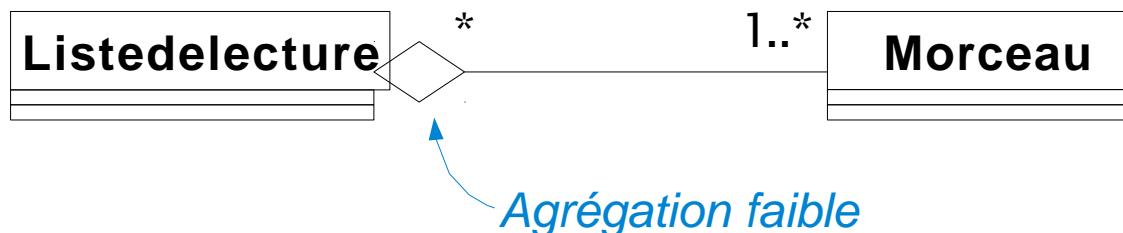
Agrégation faible

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

