

Modélisation objet

Cours 3 / GL / LP DA2I

Cédric Lhoussaine

2019-2020

Présentation

De quoi s'agit-il ?

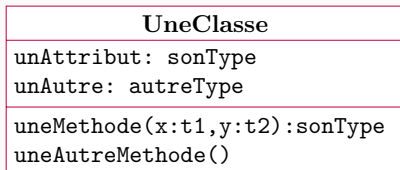
Comment passer de la **spécification** d'un problème à l'ensemble des classes nécessaires.

- liste et structure des classes
- relations entre classes
- aspects dynamiques: envois de messages

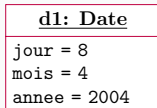
Notations UML

formalisme UML: *Unified Modeling Language*

Classe



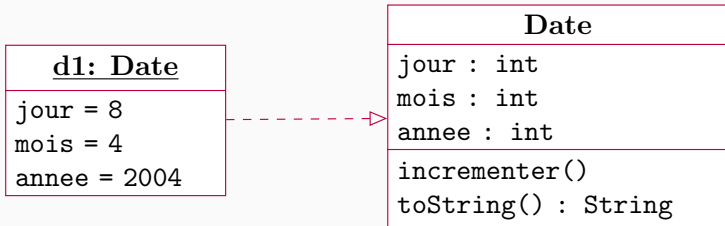
Instance



Relations

Relation "est-un"

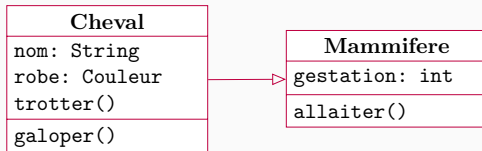
ou **instance-de** (*is-a*), relation d'appartenance à un ensemble: $d1 \in \text{Date}$



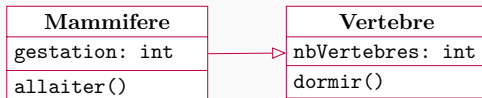
Relation "sorte-de"

Relation "sorte-de" ou "spécialise" (*kind-of*), relation d'inclusion:

Cheval \subset Mammifere



Représentation de niveaux d'abstraction élevés. Transitivité: le cheval est **une sorte de** vertébré.



Associations

Relation **structurelle** entre des objets qui ont une **fonction** l'un par rapport à l'autre.

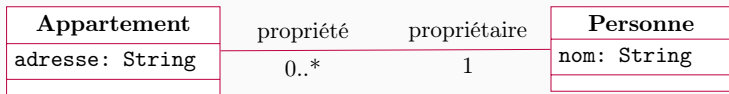
Associations entre classes (exemple)

Une personne est propriétaire d'appartements.

- classe Personne
- classe Appartement

Chaque objet joue un rôle spécifique dans l'association.

- une personne peut être propriétaire de plusieurs appartements;
- un appartement est la propriété d'exactly un propriétaire (pas de copropriété)



à noter...

- nom et sens de l'association important peu
- cardinalités minimales / maximales : 0, 1, n, *
- rôle joué par chaque classe

Une classe peut être associée :

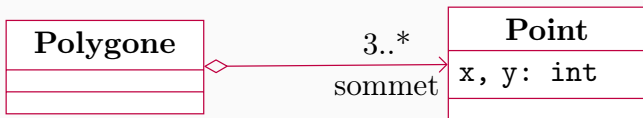
- avec plusieurs autres classes
- plusieurs fois avec une même classe
- avec elle-même

Les associations existant entre classes se traduisent, au moment de l'instanciation, par des associations entre *certaines* instances de ces classes

- Jean *possède* rue de la Paix
- Jean *possède* rue de Vaugirard
- Paul *possède* boulevard Malesherbes

agrégation = association particulière

- relation d'appartenance ou de constitution
- cardinalités 1 ► 1..* implicites
- représentation particulière



- les associations s'appliquant à des entités générales s'appliquent aussi à celles qui les **spécialisent** → placer les associations entre les entités les plus générales
- s'assurer de la **cohérence** de l'ensemble
- veiller aux **cardinalités** de chaque rôle

- relation "sorte-de" \rightarrow plus tard !
- association $(C_1, R_1, 1) \rightarrow (C_2, R_2, *)$:
 - R_1 devient un attribut de type C_1 dans la classe C_2
 - R_2 devient un attribut de type "liste de C_2 " dans C_1 (par ex. $C_2 []$)
 - en général on n'implémente **qu'un seul** des deux rôles

Exemple

```
public class Appartement {  
    // attribut "constitutif" de l'appartement  
    private String adresse ;  
    // attribut résultant de l'association  
    private Personne proprietaire ;  
}  
  
public class Personne {  
    // attribut "constitutif" de la personne  
    private String nom ;  
    // attribut résultant de l'association :  
    // un tableau (p.ex.) car la cardinalité du rôle  
    // 'propriété' est 0..*  
    private Appartement [] proprietes ;  
}
```

