

Modélisation structurelle

G. Falquet

References

Grady Booch, James Rumbaugh, Ivar Jacobson
"«The Unified Modeling Language User Guide»", Addison-
Wesley 1999

Pierre-Alain Muller
"«Modélisation objet avec UML»", Eyrolles 1997

Grady Booch, James Rumbaugh, Ivar Jacobson
"«The Unified Modeling Language Reference Manual»",
Addison-Wesley 1999

Modélisation par objets et classes

- ❖ Modélisation du monde en termes
 - ❖ d'objets
 - ❖ de classes d'objets
 - « ce qui est » ≠ « ce qu'on fait »
- ❖ Les cas d'utilisation font référence aux objets du domaine d'application
- ❖ L'architecture du système informatique sera construite autour des classes d'objets

I
S
I

June 2001 - G. Falquet, L. Nerima

Classes

3

Objets et système

- ❖ Le système est fait d'objets
- ❖ Les objets représentent des entités concrètes ou abstraites du monde réel
 - ❖ un livre de la bibliothèque
 - ❖ un emprunteur
 - ❖ un prêt (abstrait)
 - ❖ une fenêtre sur l'écran de l'utilisateur (objet d'interface)
 - ❖ une occurrence de l'évènement « le bouton STOP a été pressé à 16h22.33.01

I
S
I

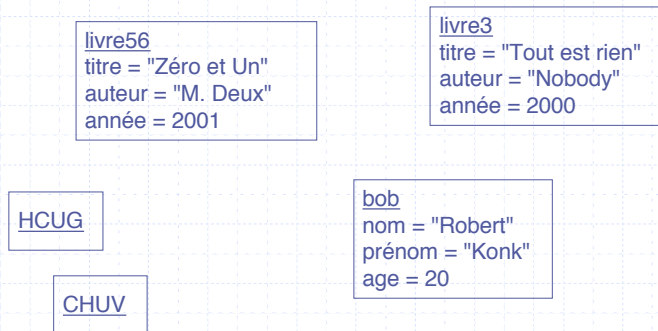
June 2001 - G. Falquet, L. Nerima

Classes

4

A l'intérieur des objets

Un objet est caractérisé par les **valeurs** de ses **attributs**



I
S
I

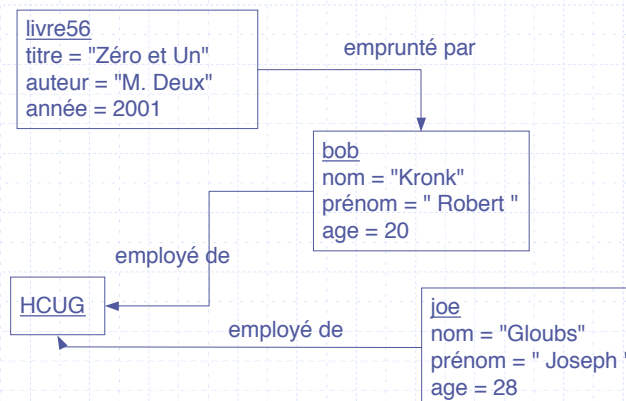
June 2001 - G. Falquet, L. Nerima

Classes

5

Liens entre objets

❖ Il peut exister des liens logiques entre objets



I
S
I

June 2001 - G. Falquet, L. Nerima

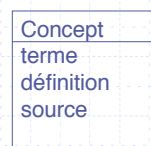
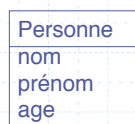
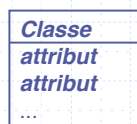
Classes

6

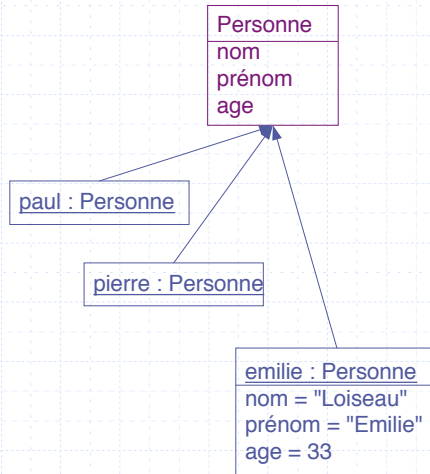
Classes d'objets

- ❖ Regrouper les objets ayant des caractéristiques communes
- ❖ Décrire globalement les propriétés communes à une classe d'objets
- ❖ Organiser le système en modules

Notation et Exemples



Les objets sont des instances de classes



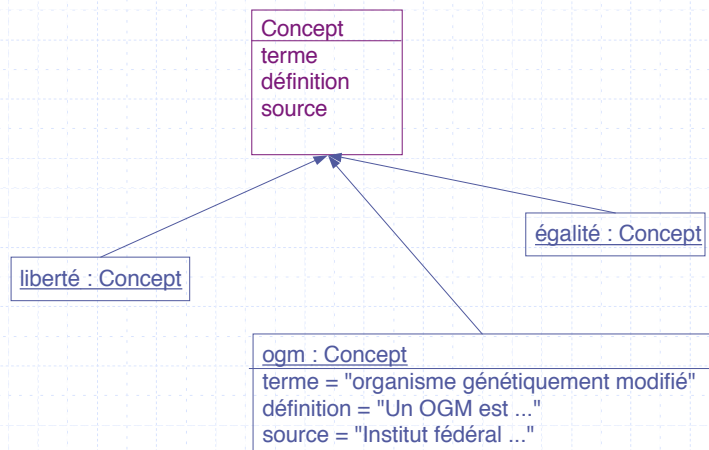
I
S
I

June 2001 - G. Falquet, L. Nerima

Classes

9

Les objets sont des instances de classes



I
S
I

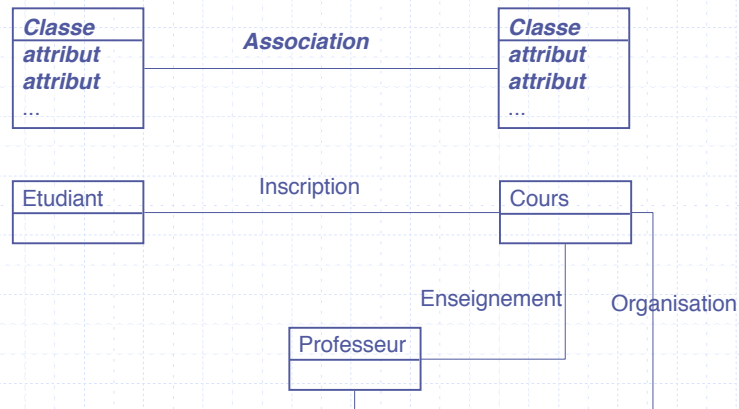
June 2001 - G. Falquet, L. Nerima

Classes

10

Associations entre objets

- ❖ Représente des liens logiques entre objets des classes



I
S
I

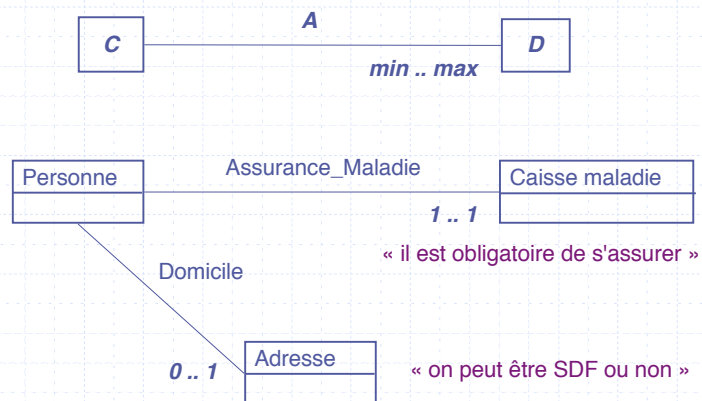
June 2001 - G. Falquet, L. Nerima

Classes

11

Multiplicité des associations

- ❖ Tout objet de C est lié par A à au moins *min* et au plus *max* objets de D.



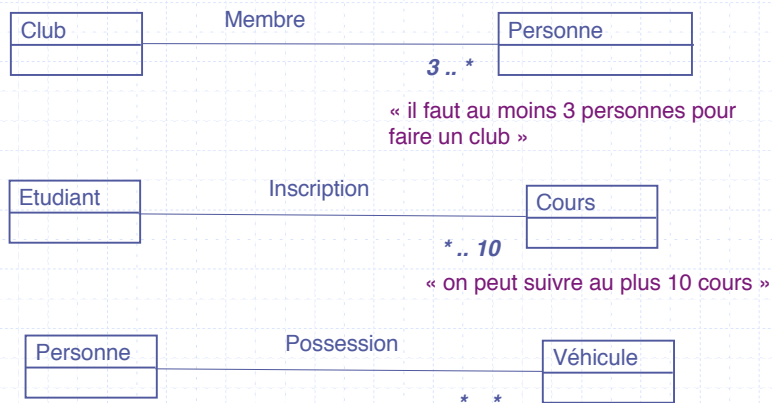
I
S
I

June 2001 - G. Falquet, L. Nerima

Classes

12

Min et max peuvent rester indéterminés



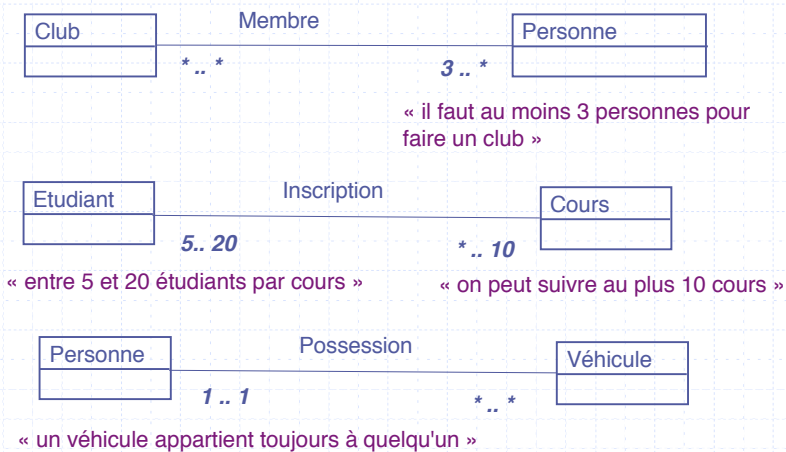
I
S
I

June 2001 - G. Falquet, L. Nerima

Classes

13

Préciser les deux extrémités de l'association



I
S
I

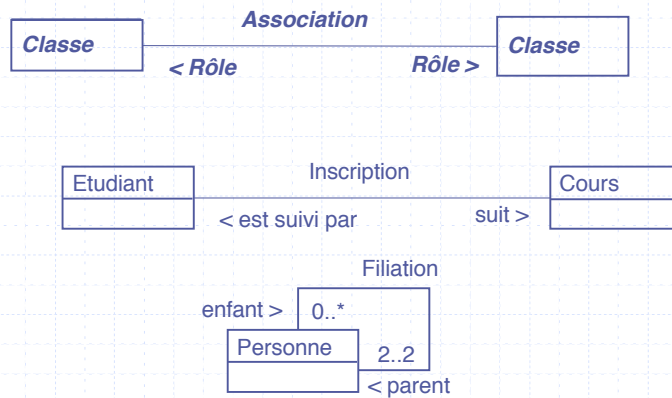
June 2001 - G. Falquet, L. Nerima

Classes

14

Associations : distinguer les rôles

- ❖ Chaque classe joue un rôle dans l'association



I
S
I

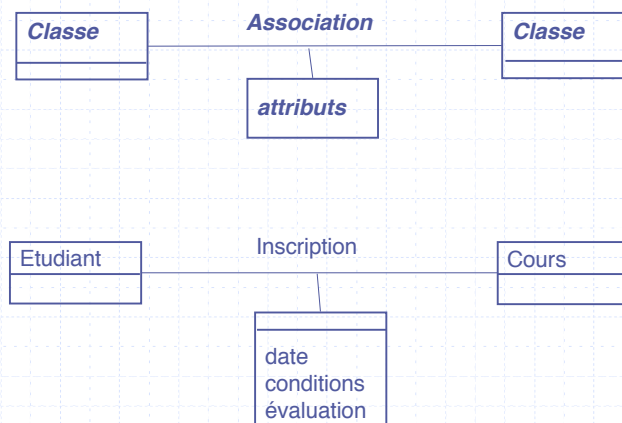
June 2001 - G. Falquet, L. Nerima

Classes

15

Associations : ajouter des attributs

- ❖ Informations propres à l'association



I
S
I

June 2001 - G. Falquet, L. Nerima

Classes

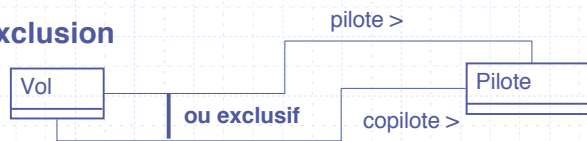
16

Contraintes entre associations

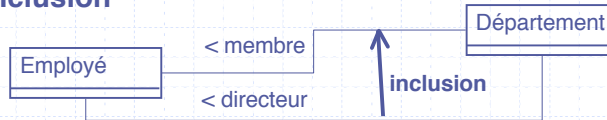
Ordonancement



Exclusion



Inclusion



I
S
I

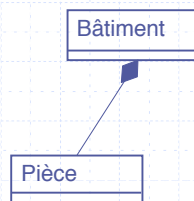
June 2001 - G. Falquet, L. Nerima

Classes

17

Composition

- ❖ Modéliser le lien « fait partie de »
- ❖ A est composé de un ou plusieurs B
- ❖ Un B ne peut exister tout seul (dans le système)
- ❖ Un B n'appartient qu'à un seul A



I
S
I

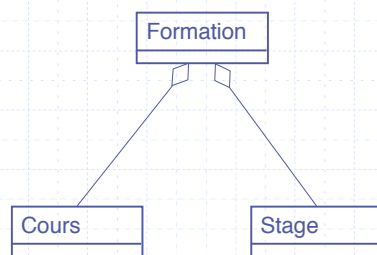
June 2001 - G. Falquet, L. Nerima

Classes

18

Agrégation

- ❖ Modéliser le lien « fait partie de »
- ❖ Moins fort que la composition
- ❖ Un B peut exister indépendamment d'un A
- ❖ Un B peut éventuellement appartenir à plusieurs A



I
S
I

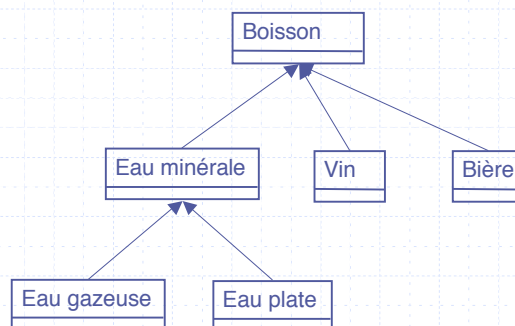
June 2001 - G. Falquet, L. Nerima

Classes

19

Généralisation et spécialisation

- ❖ Organiser le domaine (taxonomie)



I
S
I

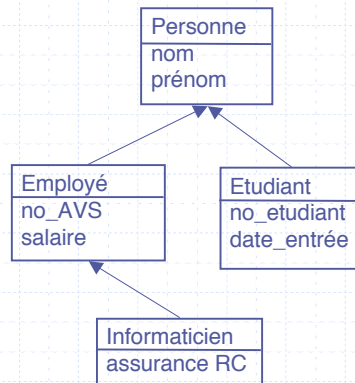
June 2001 - G. Falquet, L. Nerima

Classes

20

Généralisation et spécialisation

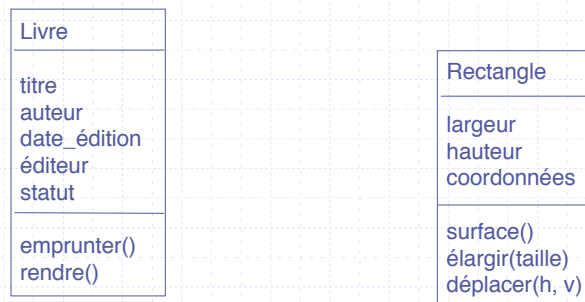
- ❖ Partager les descriptions communes



Opérations avec les objets

- ❖ Chaque objet peut être muni de **méthodes** qui rendent des **services**.
- ❖ Les méthodes seront **programmées** par les développeurs.
- ❖ Une méthode
 - ❖ calcule un résultat à partir de la valeur de l'objet et de **paramètres**
 - ❖ modifie éventuellement la valeur de l'objet

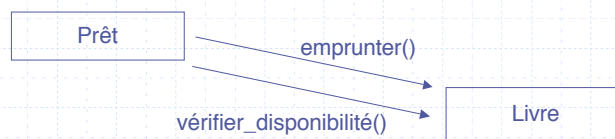
Exemple



Fonctionnement du système

Système = ensemble d'objets qui interagissent

message = demande d'exécution d'une méthode



Cas d'utilisation et objets

- ❖ Deux axes de modélisation distincts
 - ❖ structurel
 - ❖ fonctionnel

- ❖ Lien explicite
 - ❖ les scénarios parlent des objets du système
 - ❖ les classes et objets définissent le vocabulaire des scénarios

- ❖ Les cas d'utilisation déterminent l'étendue de la modélisation des objets
 - ❖ éviter le syndrome de la modélisation infinie

Modéliser quelles entités ?

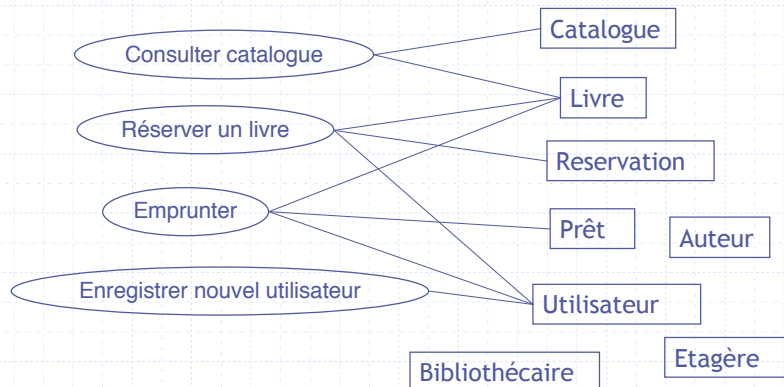
Scenario: " Vérifier la disponibilité du **livre**. Vérifier si le **lecteur** a le droit d'emprunter ce livre. Enregistrer le **prêt** "

=> le système a besoin d'objets pour représenter les **livres, lecteurs et prêts**

mais pas les étagères, portes, chaises, tables, ...

Cas d'utilisation et Objets

Raffinement d'un cas d'utilisation --> implémentation
Connecter les cas d'utilisation aux classes nécessaires



I
S
I

June 2001 - G. Falquet, L. Nerima

Classes

27

Ce n'est pas automatique

Tout terme du scénario ne se traduit pas forcément en classe

" Vérifier la disponibilité du **livre**. Vérifier si le **lecteur** a le droit d'emprunter ce livre. Enregistrer le **prêt** "

« disponibilité » => attribut de **livre**

« droit » => \emptyset

I
S
I

1. Utiliser l'intelligence et l'ingéniosité du concepteur
2. Il n'y a pas de « solution unique » en conception

June 2001 - G. Falquet, L. Nerima

Classes

28