



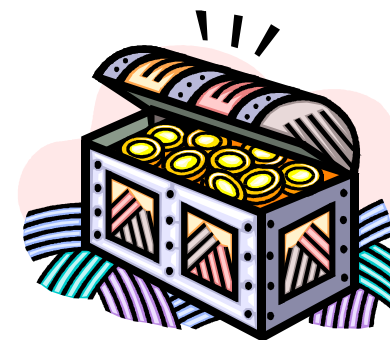
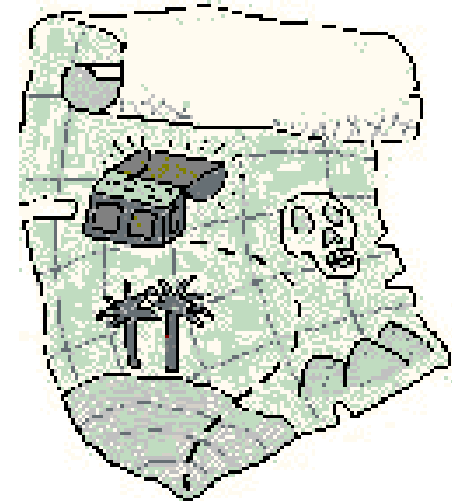
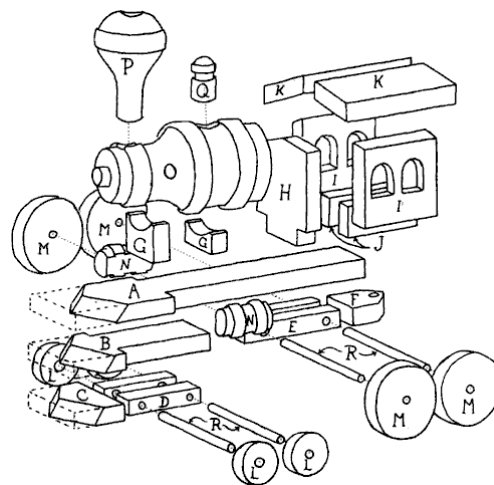
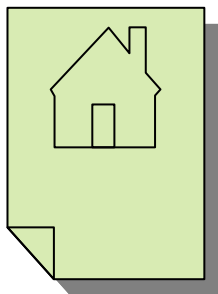
Le langage UML

Lydie du Bousquet

Lydie.du-bousquet@imag.fr

En collaboration avec J.-M. Favre, I. Parissis, Ph. Lalande

Des représentations pour discuter, organiser, construire, documenter...



Des modèles en informatique

- Comme point de départ
 - Pour abstraire, pour comprendre
 - Comme support à la discussion
 - Pour organiser, planifier
- Pour concevoir, détailler
- Pour finir
 - Pour tester
 - Pour documenter
 - Pour maintenir





UML = Unified Modeling Language

- Un langage, une notation
- Pour l'analyse et la conception orientées objets
- Unifiée
 - Pour couvrir le maximum de domaines
 - Pour couvrir le maximum de notions
- But: que les différents analystes aient
 - Un langage de discussion commun
 - des outils communs





Avant UML

- Des terminologies différentes
 - Classes, objets, types
 - Relation, connexion d'instance, utilise, association
 - Héritage, généralisation, sous-type, spécialisation
 - Agrégation, composition, contient, tout/partie
- De nombreux autres langages (notations)
 - OMT, Merise, Booch, +50 méthodes
- Des outils non compatibles
 - Rational Rose, Objecteering, TogetherJ, ArgoUML,



UML = standard

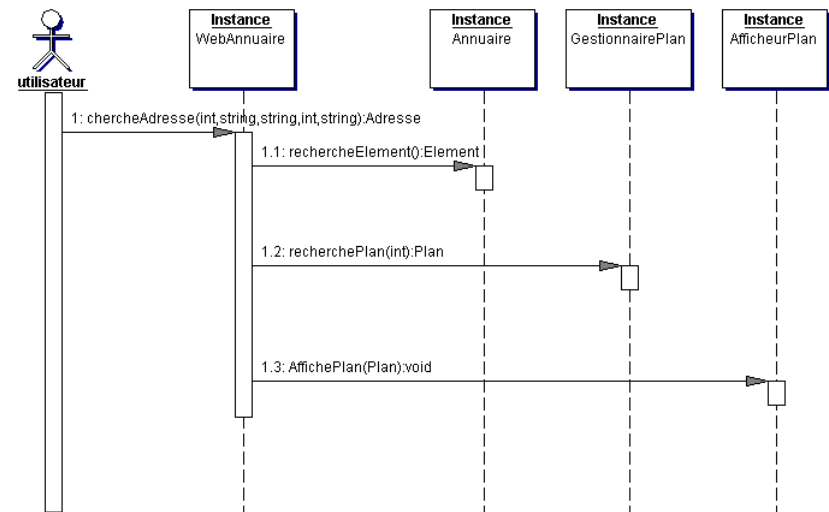
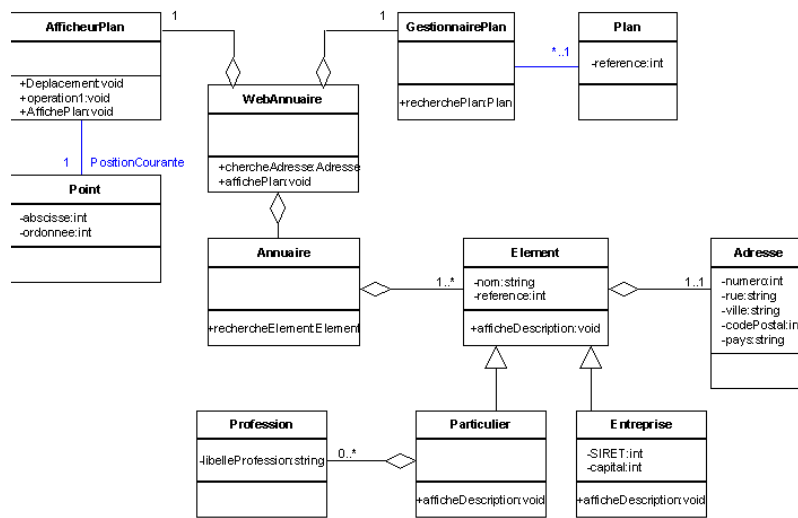
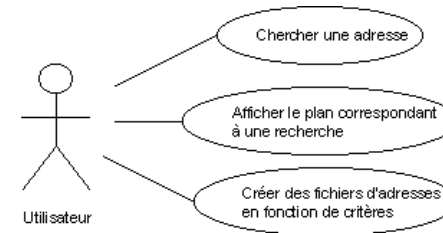
- Standard international
 - Pour contenter le plus grand nombre
 - Assez large (beaucoup de notions)
- Utilisé de facto dans l'industrie
 - De plus en plus
- Associés à plusieurs méthodes
- Plusieurs outils
 - Vers une plus grande compatibilité



UML : un langage, des vues

- Besoins différents
 - aspects statiques ou dynamiques
 - analyse, spécification, conception, ...
- Notions de vues
 - Pour séparer les préoccupations

UML: un langage, des vues





Les briques de base

- Éléments constituant des modèles
 - Classes, interfaces, composants, cas d'utilisations...
- Relations
 - Associations, généralisations, dépendances,...
- Diagrammes
 - De classe, de cas d'utilisation, de séquence, ...



Règles de mises en forme

- Les briques s'assemblent selon des règles
- Règles sémantiques pour
 - Les **noms** : manière de nommer les éléments, diagrammes
 - Le **contexte** : environnement qui donne une signification précise à un nom
 - La **visibilité** : façon dont les noms peuvent être vus et utilisés par d'autres
 - L'**intégrité** : manière dont les objets établissent des relations correctes et cohérentes
 - L'**exécution** : conséquences de l'exécution d'un modèle dynamique



Certains modèles peuvent être

- **Partiels**
des éléments sont cachés
- **Incomplets**
des éléments manquent
- **Incohérents**
l'intégrité n'est pas garantie

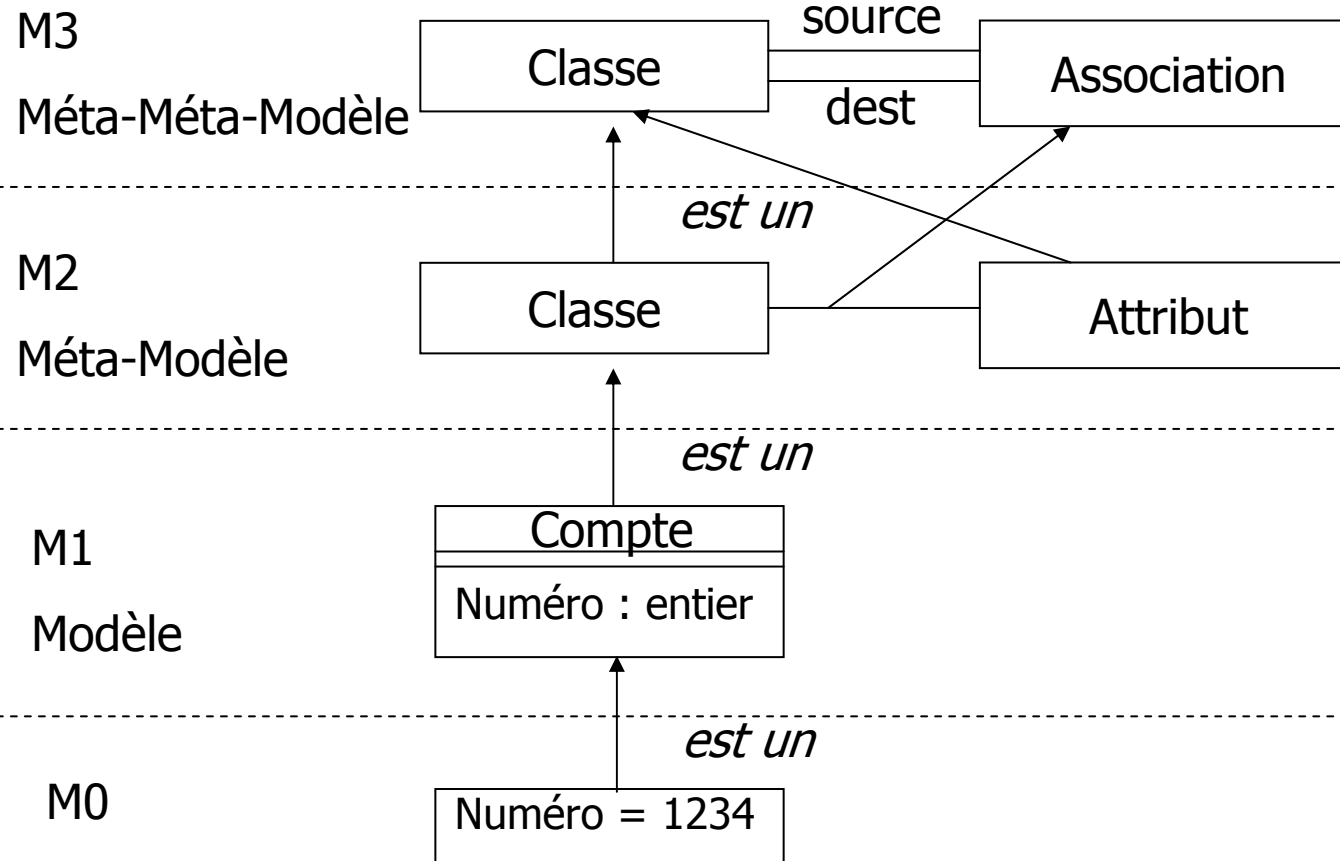


Exemple de règles

| Compte |
|----------------|
| numéro |
| solde |
| découvertMax |
| ConsulterSolde |
| Créditer |
| Débiter |

- Si une classe est concrète, toutes les opérations doivent avoir un code associé
- Une classe est dessinée par un rectangle composé de 3 compartiments séparés par des lignes horizontales
- Les attributs et opérations peuvent ne pas apparaître dans une présentation de classe
- Le nom d'une classe doit commencer par une majuscule
- ...

Notion de méta-modèle





Les 13 diagrammes d'UML 2.0

- de cas d'utilisation
- de classes
- d'objets
- de séquences
- de collaborations
- d'états
- d'activités
- de composants
- de déploiement
- de modules (package diagram)
- de structures composites
- global d'interaction
- de temps (timing diagram)



Capturer la structure avec

un diagramme de classes
un diagramme d'objets



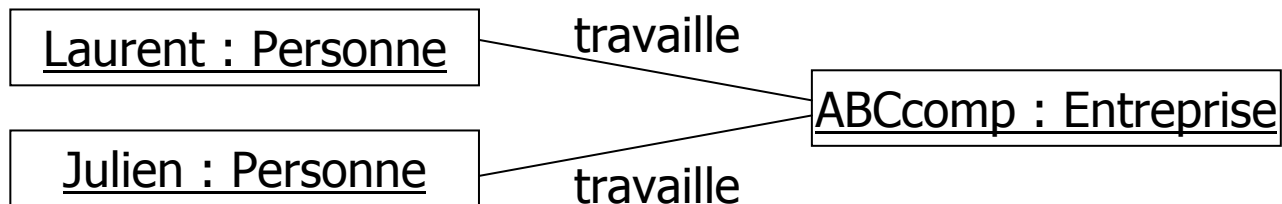
Un diagramme de Classe

- Montre les classes et leurs relations
 - Attributs et opérations des classes
 - Héritage, agrégation, association
- Fait abstraction des aspects dynamiques et temporels
- Peut être instancié en un diagramme d'objets
- Est utilisé
 - Pour l'analyse et la conception globale



Association entre classes

- Exprime une connexion sémantique entre classes
- Définit un ensemble de lien
- Pour le diagrammes de classe ou d'objets

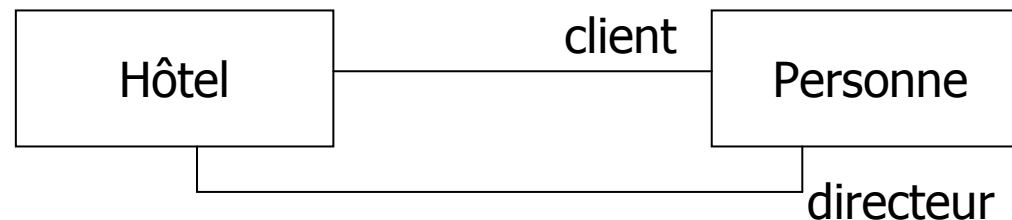


Documentation d'une association

- **Sens de lecture** principal d'une association

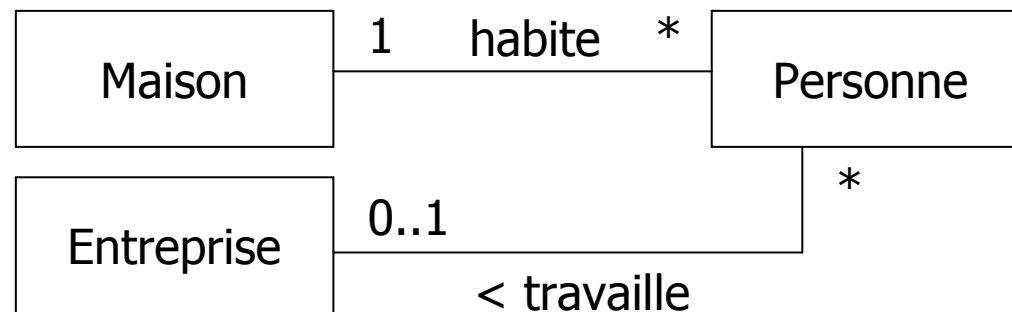


- **Rôles** pour spécifier la fonction d'une classe pour une association



Documentation d'une association

- **Cardinalité** : précise le nombre d'instances



Une personne travaille pour au plus une entreprise
Plusieurs personnes travaillent pour une entreprise donnée



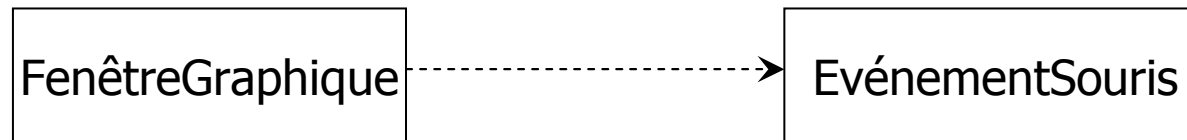
Documentation d'une association

- **Cardinalité** : précise le nombre d'instances
 - **n** : exactement « n » (entier naturel) (3 ou 12 ...)
 - **n .. m** : de « n » à « m » : 0..1, 3..n, 1..31
 - ***** : équivalent à « 0..n » et « 0..* »
 - **n .. *** : « n » ou plus



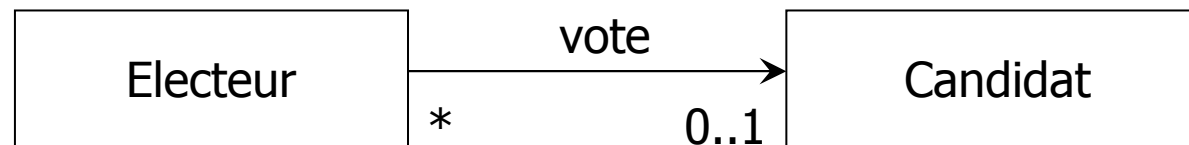
Dépendance

- Pour exprimer une relation d'utilisation unidirectionnelle
- Une modification de l'élément dont on dépend peut nécessiter une mise à jour de l'élément dépendant



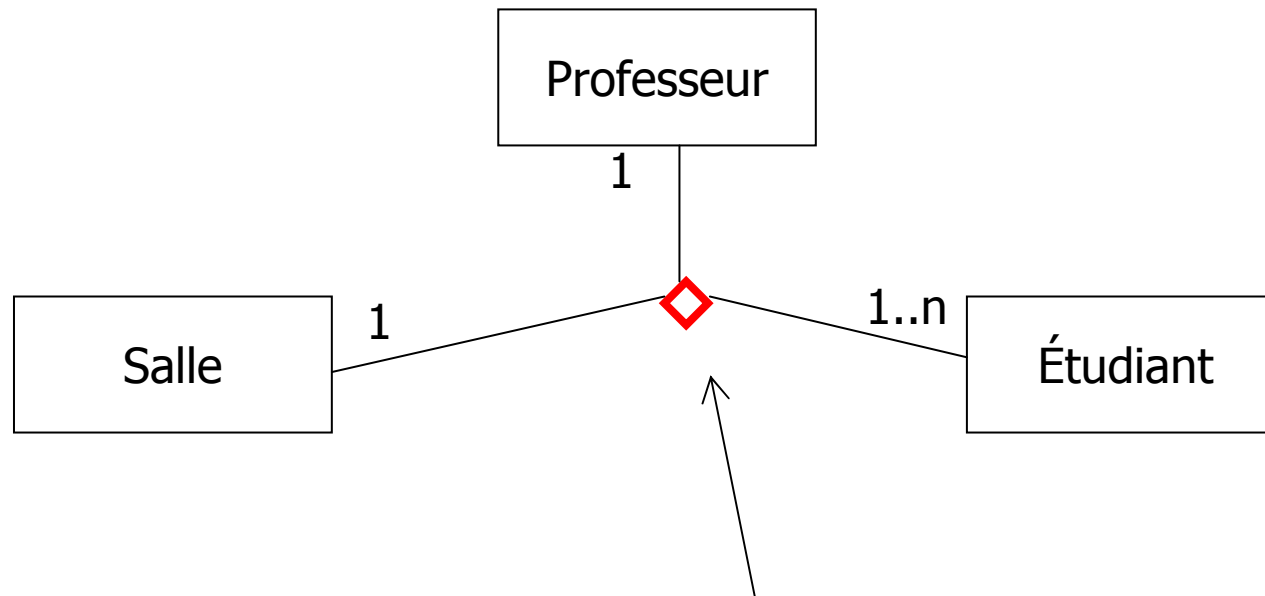
Association à navigabilité restreinte

- Pour exprimer qu'une instance d'une classe ne connaît pas les instances de l'autre
- Par défaut, une association est navigable dans les 2 sens



Association n-aire à ne pas utiliser...

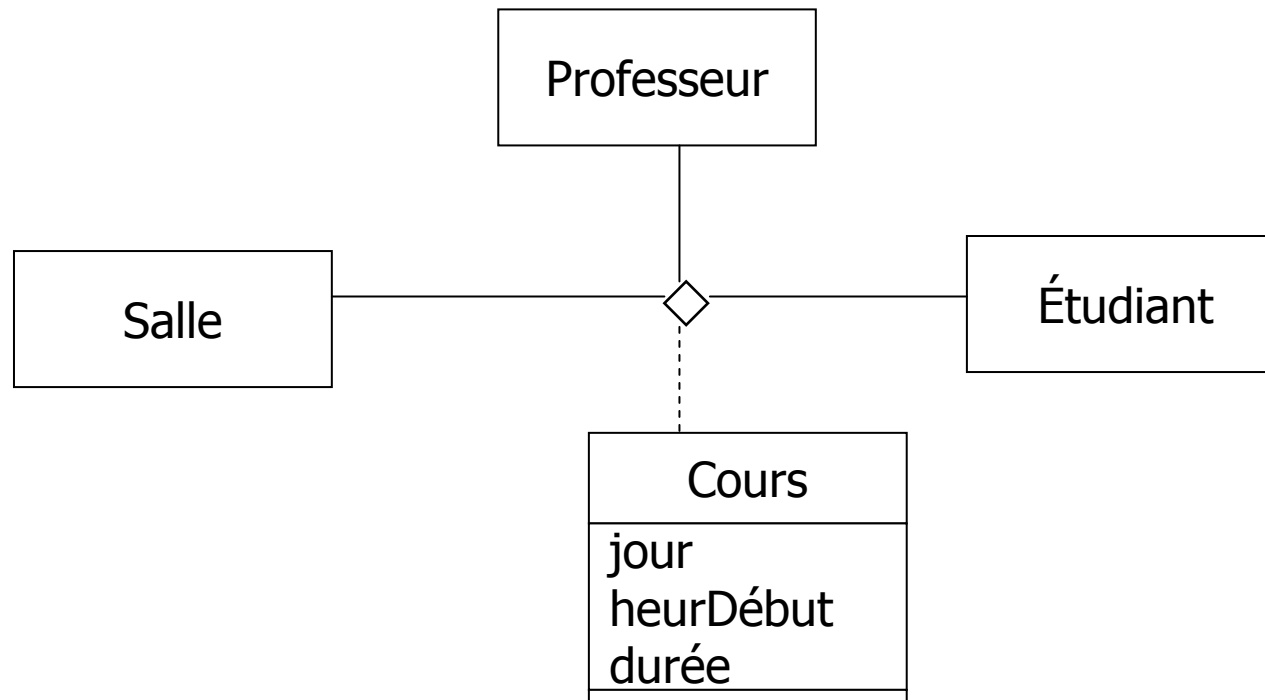
- Pour relier plusieurs classes
- Difficile à comprendre...



Utiliser une classe pour remplacer le symbole d'association

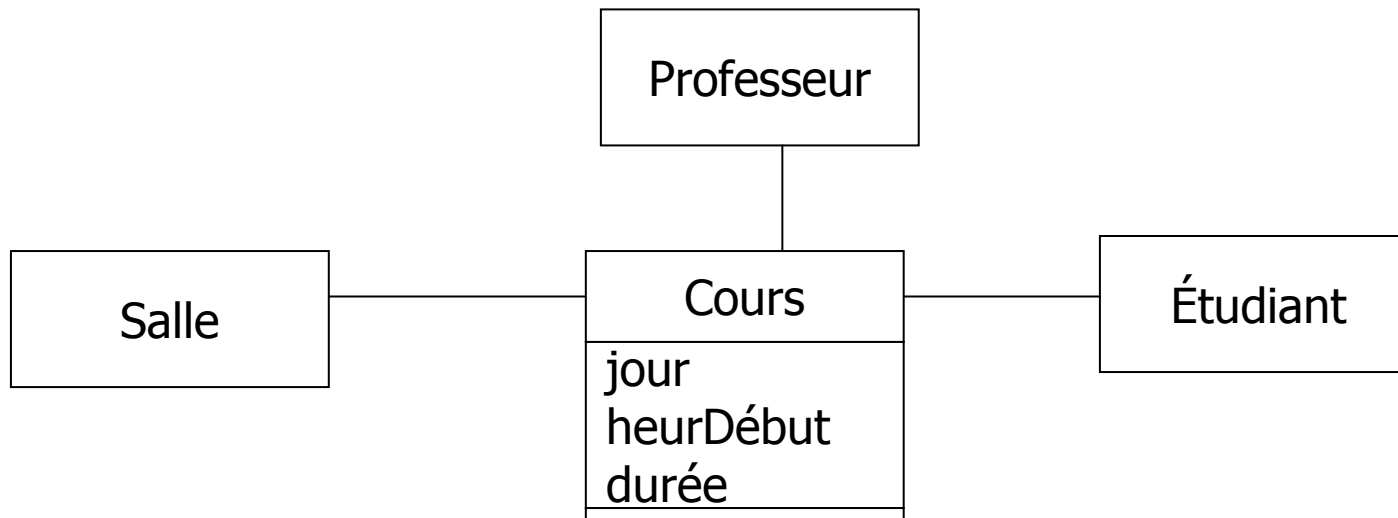
Classe d'association

- Pour réaliser la navigation entre les instances d'autres classes

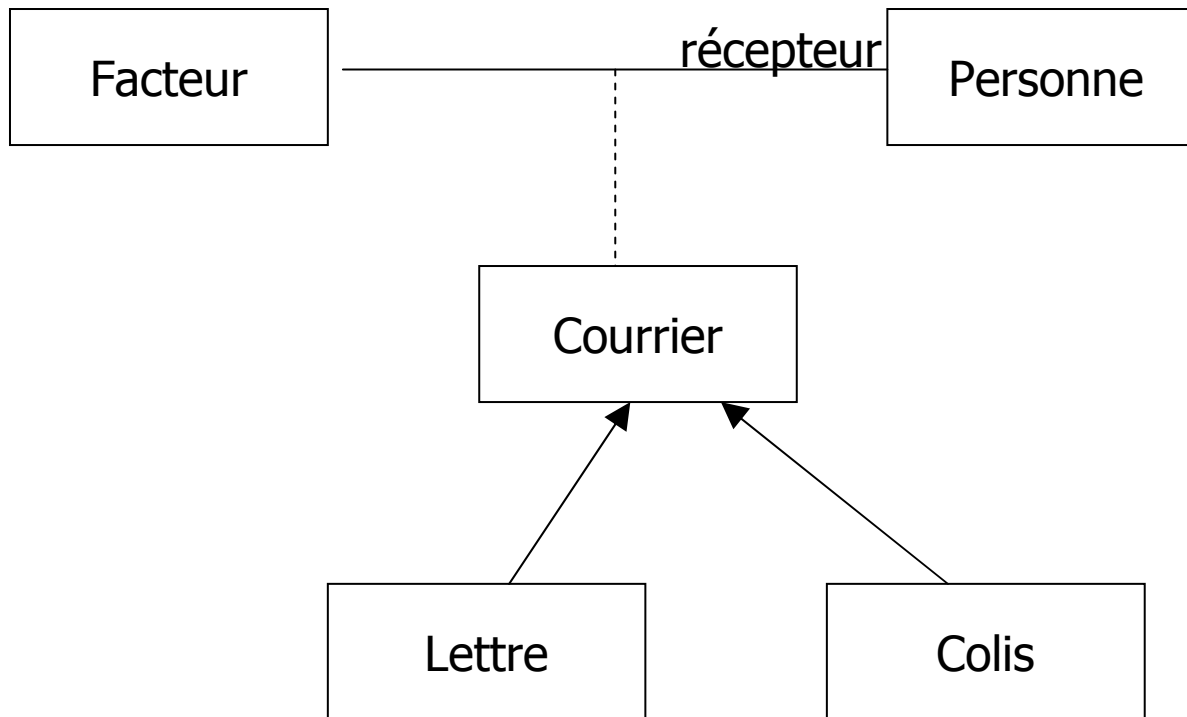


Classe d'association

- Pour réaliser la navigation entre les instances d'autres classes



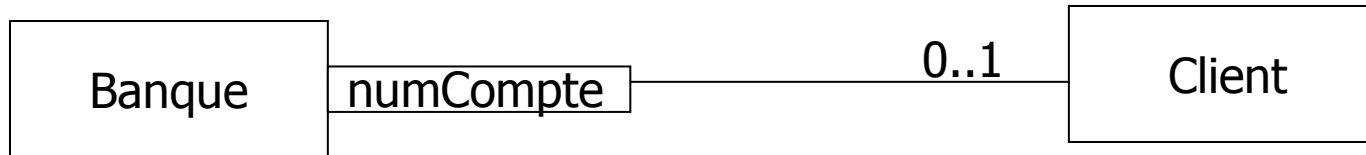
Classe d'association (2)





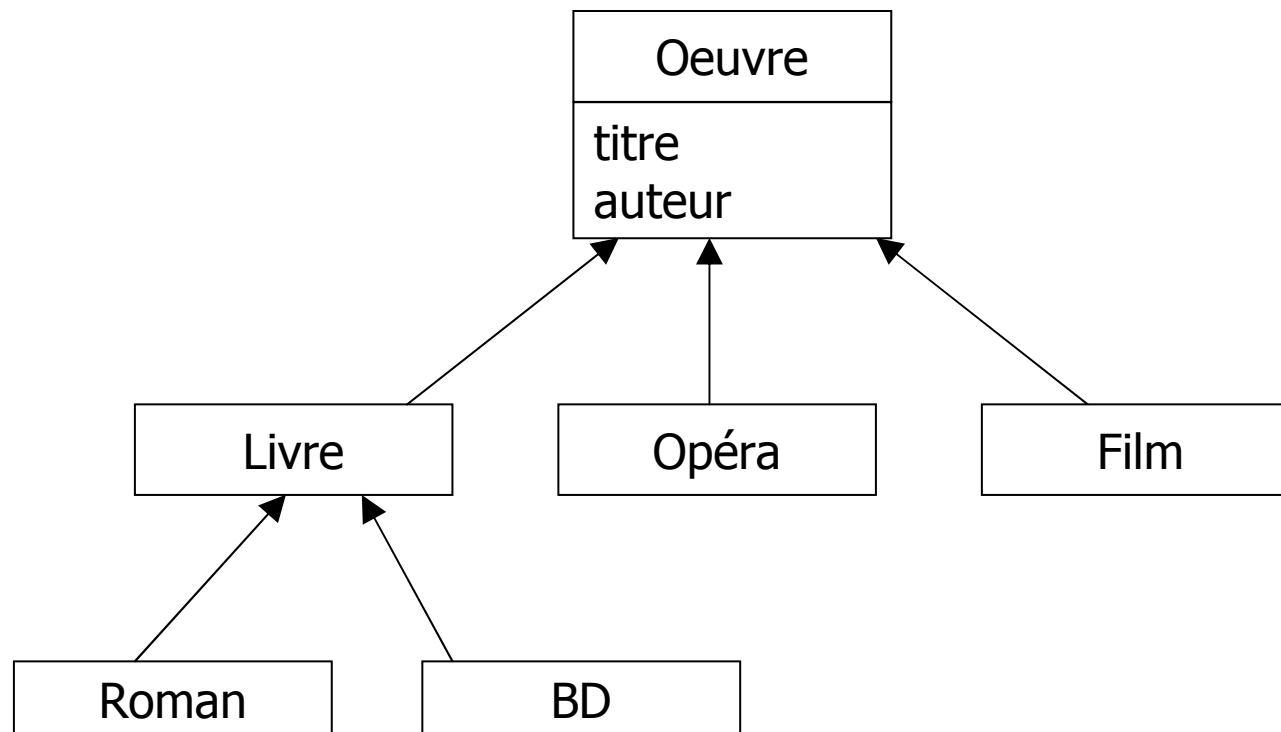
Qualification

- Pour sélectionner un sous-ensemble d'objet (instances)



Héritage

- Pour réutiliser et gérer la complexité
- Classification des objets



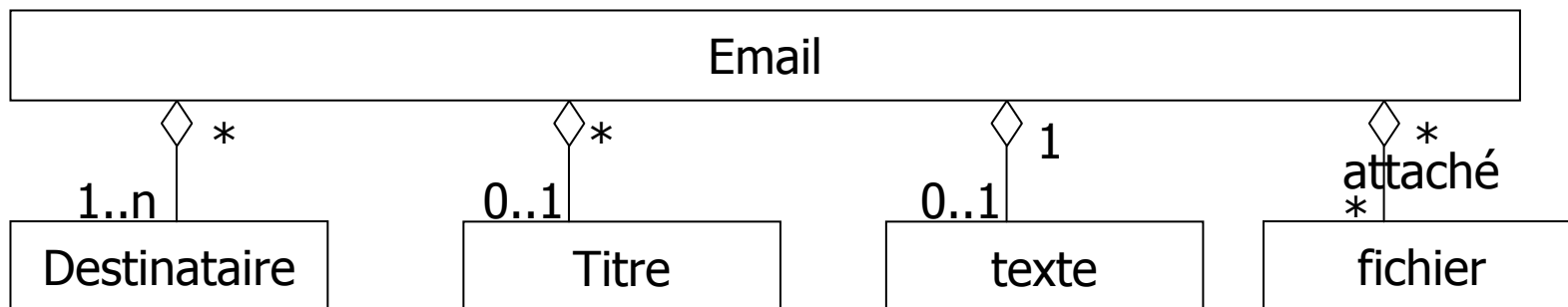
spécialisation



généralisation

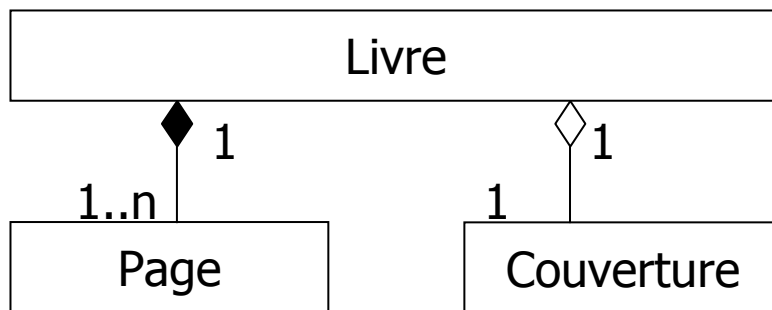
Agrégation

- Association non symétrique
- Relation de subordination
- « Ensemble / élément »
- Ensemble et élément peuvent vivre l'un sans l'autre



Composition

- Agrégation forte
- Les composants et l'agrégat sont liés : si l'agrégat est détruit, les composants aussi



Les pages sont physiquement contenues dans le livre

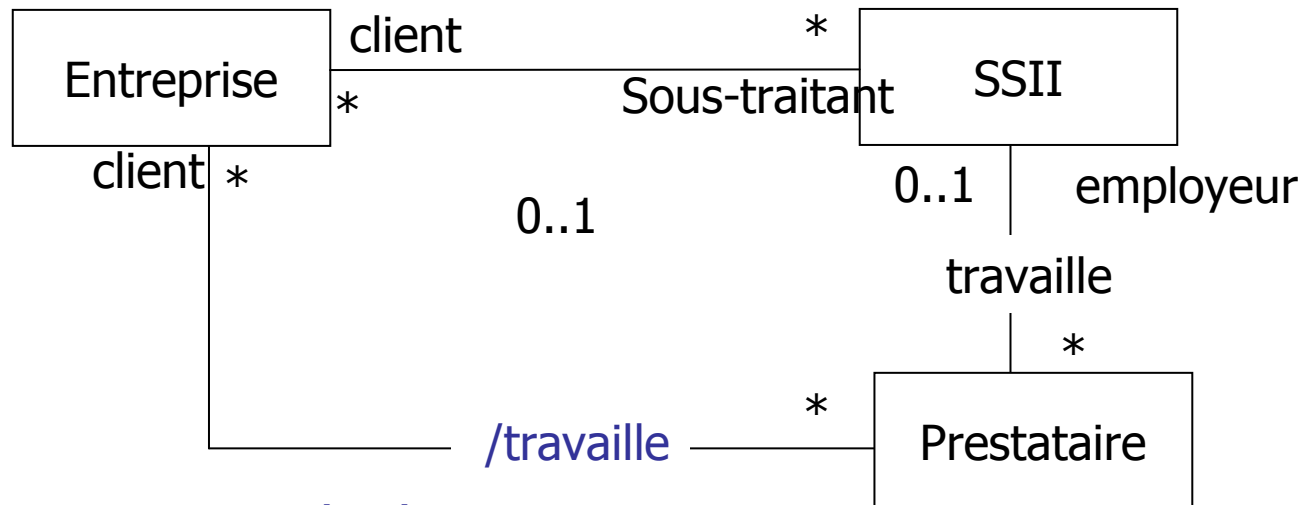


Composition et agrégation

- Vues subjectives
- A utiliser pour ajouter une sémantique aux modèles, lorsque c'est pertinent

Association dérivée

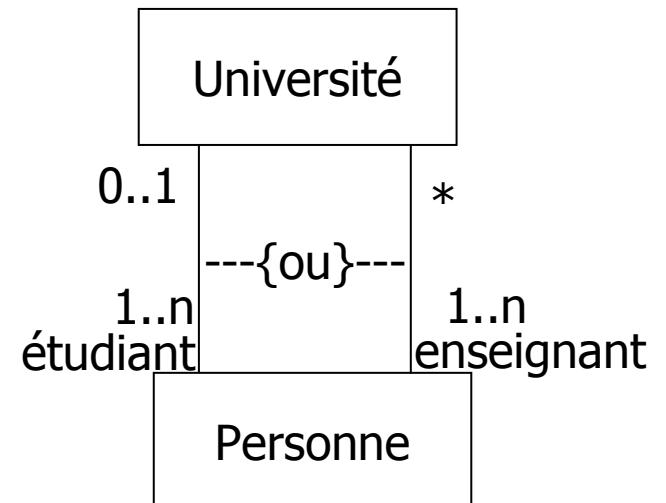
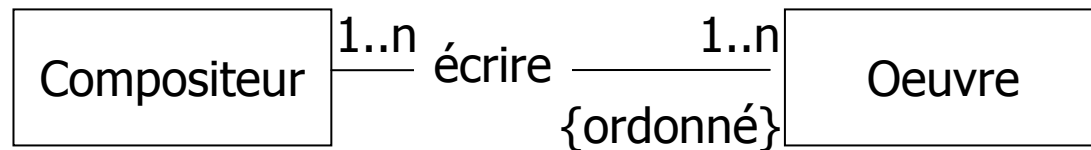
- Association redondantes
- Que l'on peut déduire d'une autre/d'autres association(s)
- Pour indiquer les chemins de navigation calculés
- Pour la compréhension de la navigation



Association dérivée

Contrainte sur une association

- Expression pour préciser le rôle ou la portée d'un élément de modélisation
- Peut-être exprimé en langage naturel ou en OCL





Exercice...
