

- Langage graphique
- 13 diagrammes
 - aucun n'est obligatoire
 - utiliser le diagramme le plus adapté pour la modélisation et la communication
- 3 types de diagrammes
 - diagrammes structurels
 - diagrammes comportementaux
 - diagrammes d'interactions

- Diagrammes structurels
 - vues statiques du système
 - diagramme de classes
 - diagramme de composants
 - diagramme de structures composites
 - diagramme de déploiement
 - diagramme de packages
 - diagramme d'objets

- Diagrammes comportementaux
 - vues dynamiques du système
 - diagramme d'activité
 - diagramme de communication
 - diagramme d'interaction d'ensemble
 - diagramme de séquence
 - diagramme d'états-transitions
 - diagramme de chronométrage
 - diagramme des cas d'utilisation

- Diagrammes d'interactions
 - sous-famille définie dans UML 2
 - diagramme de communication
 - diagramme d'interaction d'ensemble
 - diagramme de séquence
 - diagramme de chronométrage

Méta-modèle

Classificateur

- groupe d'objets possédant des propriétés commune
 - une classe par exemple
- notation sous forme de rectangle pouvant être divisé avec des compartiments
 - sous forme d'ellipse dans le diagramme des cas d'utilisation

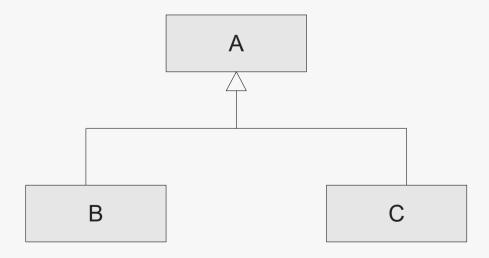
Décoration

- contraintes: {abstract} {ordered,unique}
- stéréotypes: <<interface>> <<entity>>
- garde: [condition] [de 10 à 10] [b==true]

Méta-modèle

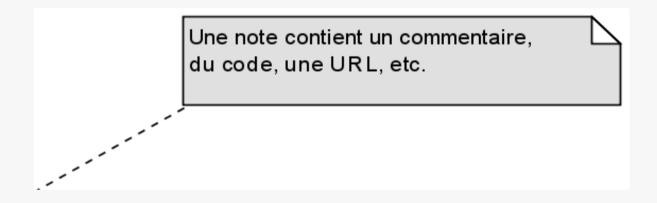
Généralisation

- permet de factoriser des éléments communs à des classificateurs
- le classificateur A factorise des éléments des classificateurs B et C

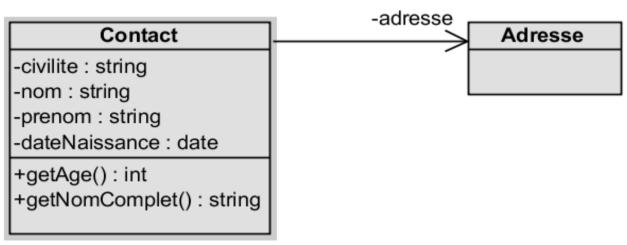


Méta-modèle

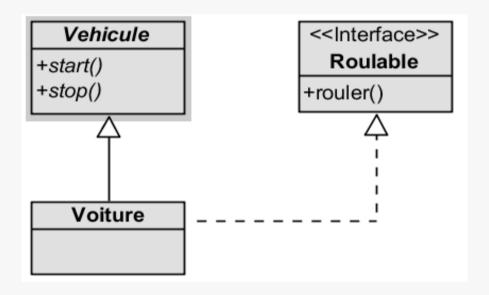
- Note
 - utilisée comme commentaire dans les diagrammes



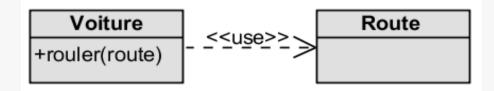
- La classe est représentée dans un rectangle
 - 3 compartiments
 - nom de la classe seul compartiment obligatoire
 - propriétés
 - si une propriété est liée à la classe par une relation, elle n'apparaît pas dans la classe mais dans la relation
 - opérations



Héritage et réalisation

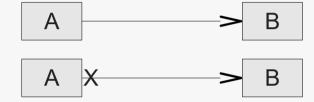


- Relation entre les classes
 - dépendance
 - flèche pointillée
 − − − − − − >
 - relation d'utilisation
 - il n'y a pas de propriété Route dans Voiture
 - une instance de Route est passée à la méthode rouler (...) de Voiture

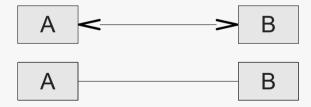


- Relation entre les classes
 - association ———>
 - "... a un ..." mais pas de notion de propriété
 - agrégation
 - "... est propriétaire de ..."
 - mais l'objet n'est pas détruit avec son propriétaire
 - composition ◆
 - "... fait partie de ..."
 - possession totale, les cycles de vie sont liés

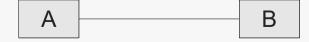
- Navigabilité des associations
 - unidirectionnelle



- bidirectionnelle



- inconnue



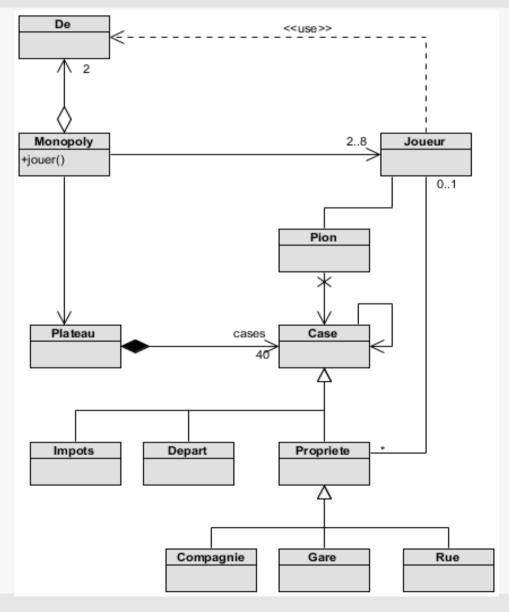
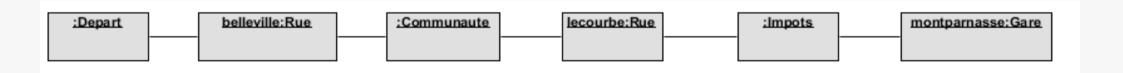


Diagramme d'objets

- Instantané d'une partie des objets présents à un moment donné
- Représente une configuration particulière d'objets
- Aide à comprendre certains diagrammes de classes

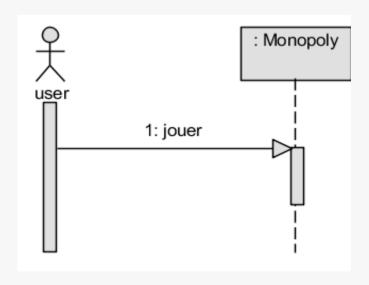


- Acteurs d'une interaction
 - instance de l'objet
 - ligne de vie de l'objet
 - nom de l'acteur sous la forme

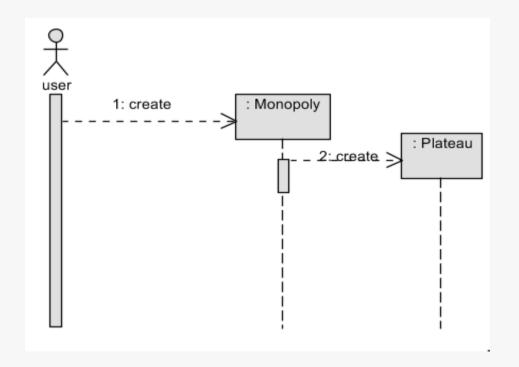
```
nom [sélecteur] : classe
```

- nom : nom de l'instance
 - si absent : instance anonyme
- sélecteur : si nécessaire, représente l'indice de l'instance dans une collection
- classe nom de la classe obligatoire, toujours préfixé par :

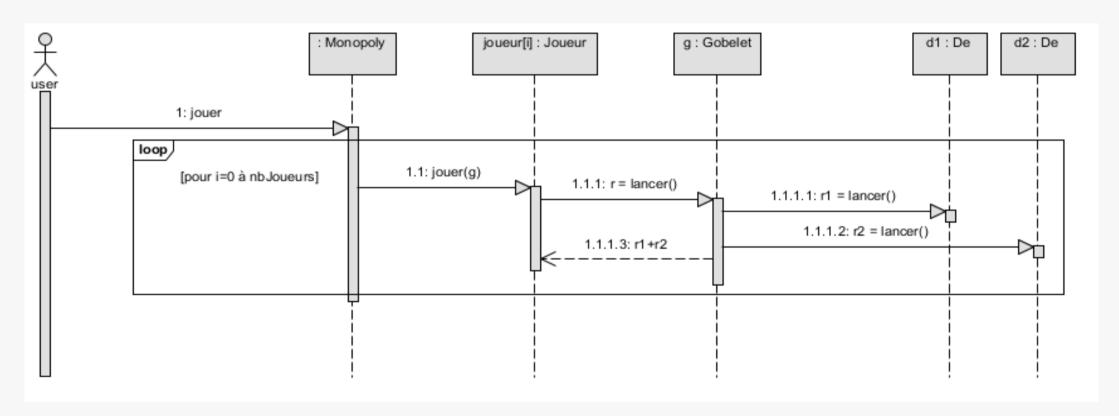
- L'acteur qui initie le premier diagramme de séquence est souvent représenté par un bonhomme en fil de fer
 - correspond au début d'un cas d'utilisation



 Le diagramme peut comporter les créations d'instances



- Les fragments combinés représentent des blocs
 - boucles, tests, choix, etc.



- Un message est envoyé vers une instance
 - le message correspond à une méthode de la classe de destination
 - le retour de valeur peut être spécifié par une flèche pointillée
- Les diagrammes peuvent se référencer les uns les autres pour une meilleure lisibilité

Diagramme états-transitions

- Deux types de diagramme
 - pour les machines à état
 - pour les protocoles
- Un état
 - modélise un moment spécifique, une situation
 - caractérisé par
 - son nom
 - ses activités internes (facultatif)
 - en entrée dans l'état, pendant l'état, en sortie de l'état

Diagramme d'états-transitions

- Une transition représente une relation entre deux états
 - syntaxe

déclencheur[garde]/action

- déclencheur : événement qui fait passer d'un état à un autre
- garde : contrainte évaluée lorsque l'événement se produit
- action : activité exécutée si la transition est effective
- déclencheur, garde et action sont facultatifs



Diagramme d'états-transitions

- Il existe des pseudo-états pour décrire
 - le départ, la fin, les choix, des points d'entrée et sortie vers des sous-états, les historiques, ...
 - seul le point de départ est obligatoire
 - point de départ
 - point de fin (●)



Diagramme d'états-transitions

• États d'une propriété du jeu de Monopoly

