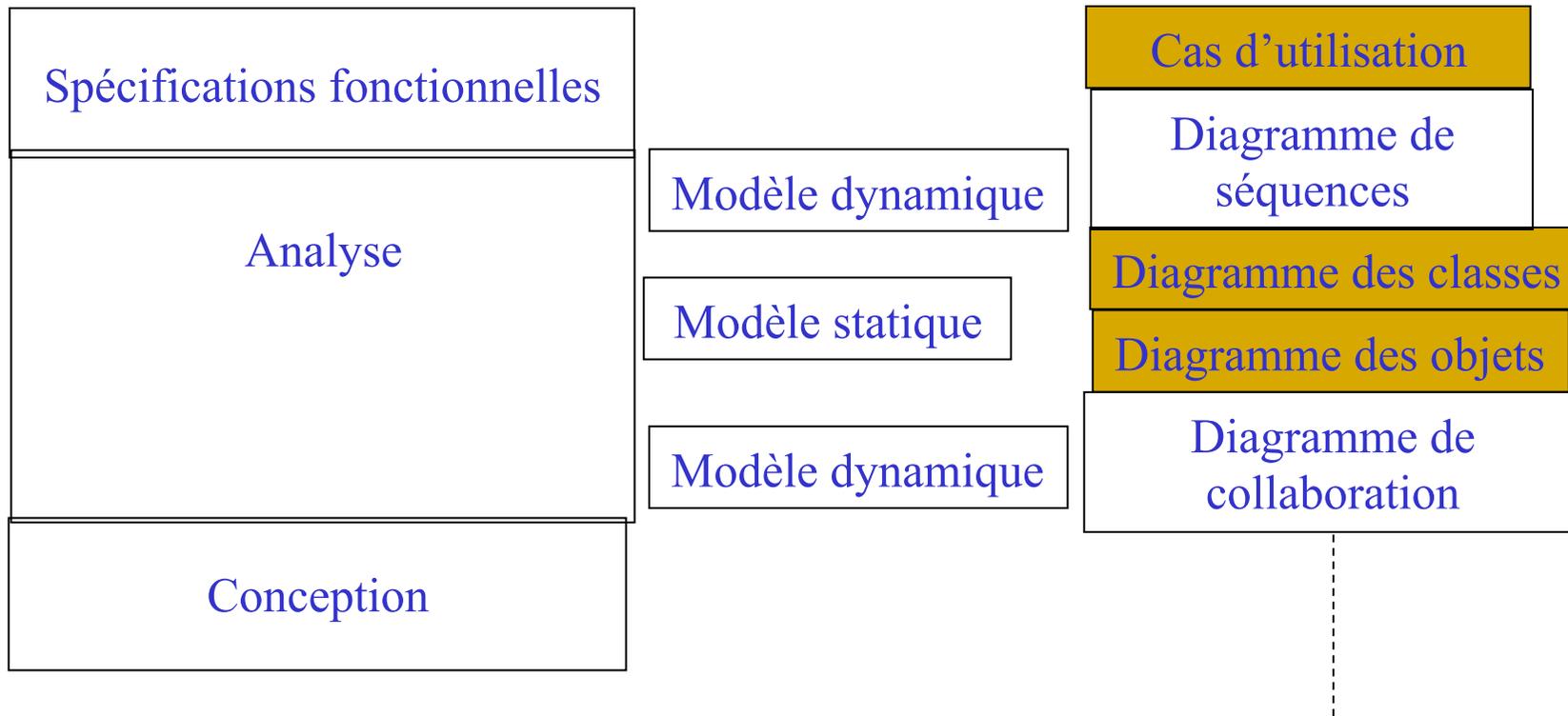


Le diagramme des objets

A.Osmani

Site officiel UML : www.uml.org

Le diagramme des classes en UML



Sommaire

1. Diagramme des objets
2. Définition d'un objet
3. Description objet-classe
4. Notations avancées

Diagramme des objets

Le diagramme des objets est représenté par le graphe des instances, contenant les objets, les valeurs des attributs ainsi que les relations entre objets.

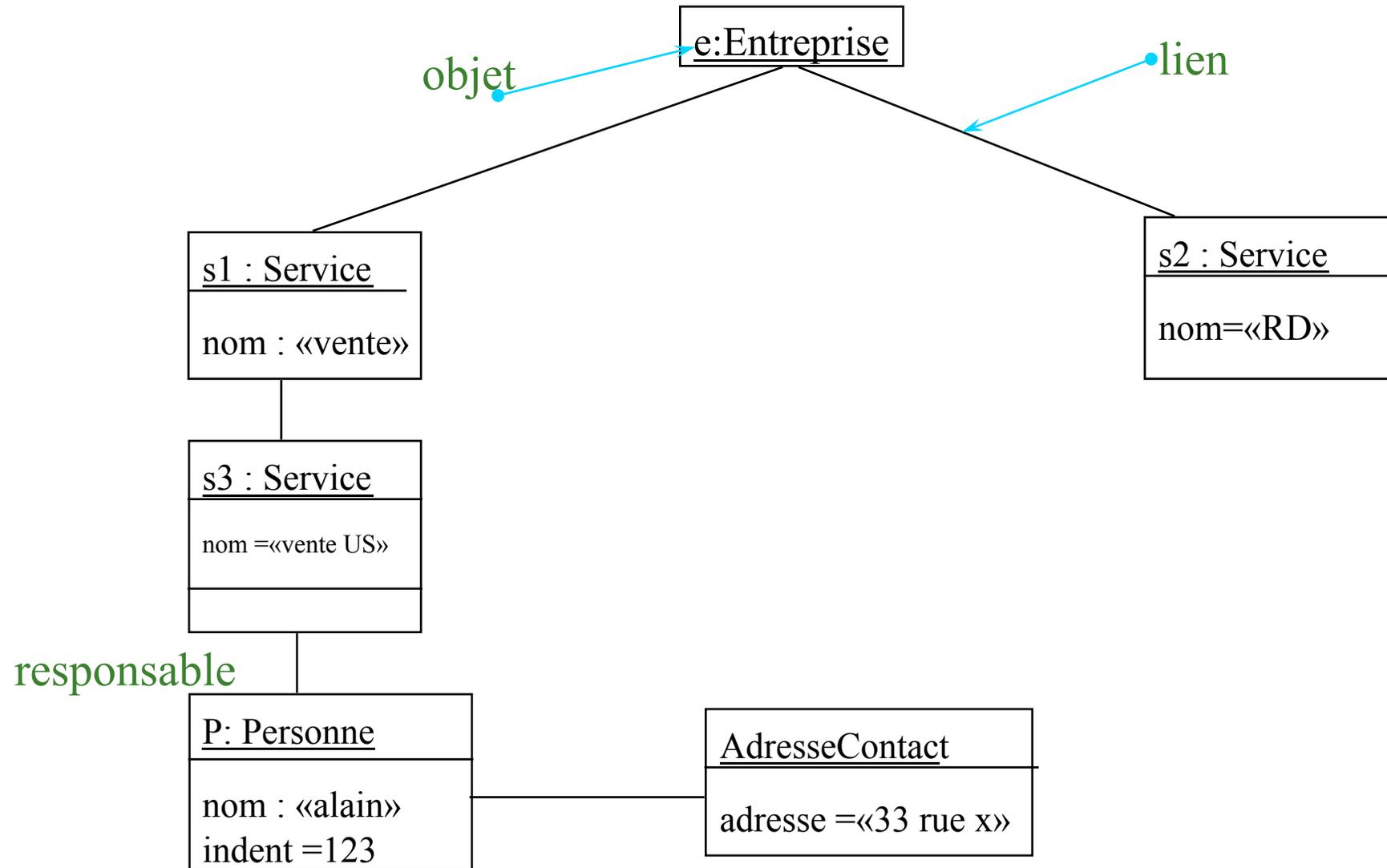
Le diagramme des objets est une instance du diagramme des classes; il décrit l'état détaillé des classes à un instant donné.

Le diagramme des objets est limité à la présentation d'exemples illustrant les classes et leurs structures et faisant apparaître les relations spécifiques entre des instances particulières.

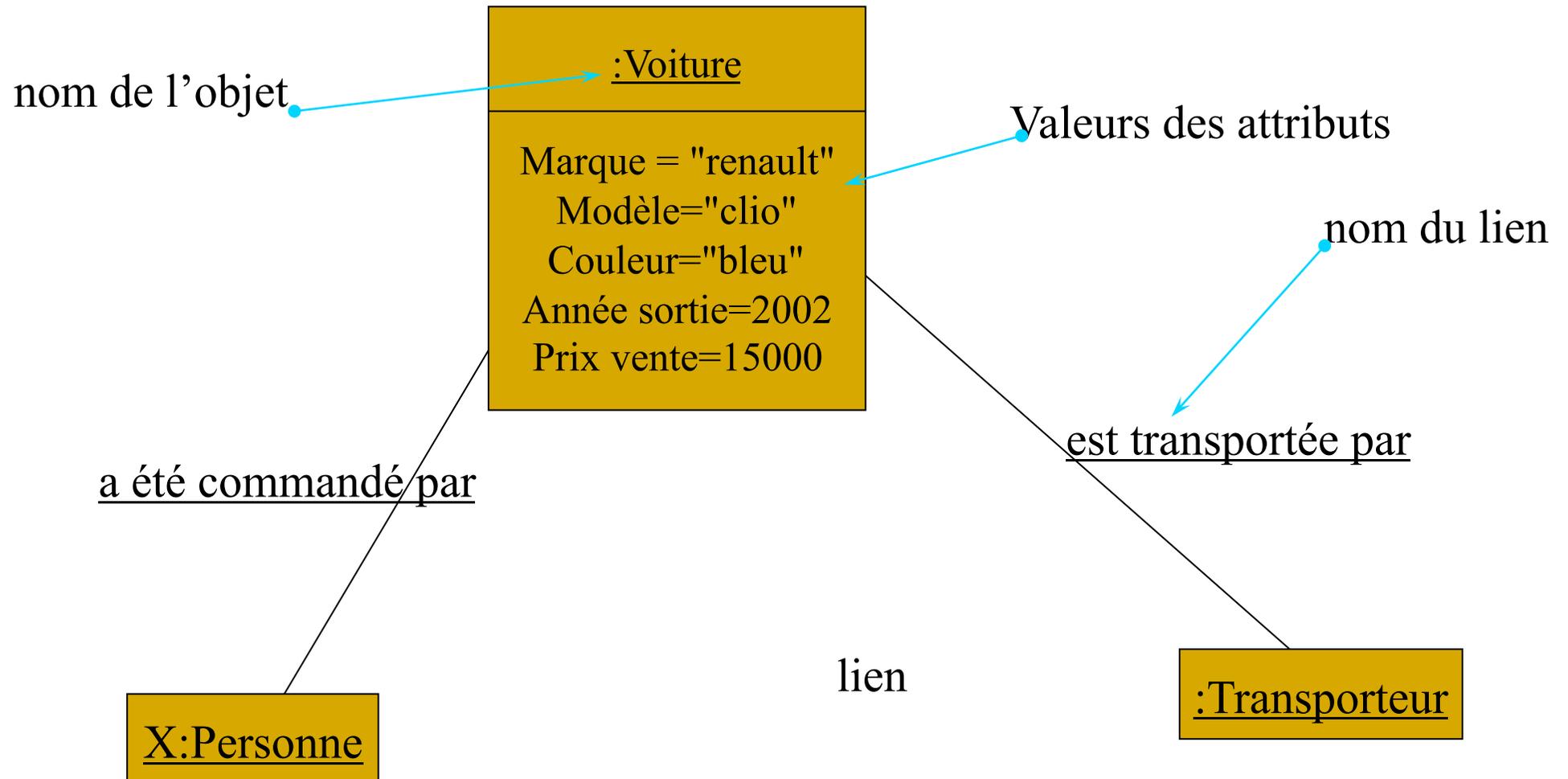
Le diagramme des classes peut contenir des objets, quand il ne contient que des objets alors il devient diagramme des objets.

Le diagramme des objets peut donc être considéré comme un cas spécial du diagramme des classes ou bien du diagramme de collaboration (que nous verrons plus tard).

Exemple de diagramme d'objets



Exemple de diagramme d'objets



Définition d'un objet

Un objet est une représentation abstraite d'une entité qui existe dans le monde modélisé.

Un objet est une abstraction pour une entité

- ❑ Matérielle. Exemple : une table
- ❑ Immatérielle. Exemple : une commande
- ❑ Logique. Exemple : le plan d'une activité
- ❑ D'une interface. Exemple : le bouton ok
- ❑ Informatique. Exemple : un fichier
- ❑ Active. Exemple : une tâche

L'exécution d'un système repose sur les objets et les lien.

(lien = connexion entre plusieurs objets. Il permet l'envoi de messages entre objets, il peut être statique ou transitoire (chemin temporaire établi pour l'envoi de messages à un instant donné))

L'objet et la classe

- Une classe est un descripteur d'objets.
- Un objet est une manifestation d'une classe.
- Tous les objets représentés par une classe possèdent la même description
- Une classe définit
 - Un ensemble d'attributs
 - Un ensemble d'opérations
 - Un ensemble de relations
 - Un ensemble de responsabilité
- Chaque objet
 - Possède ses propres valeurs pour ses attributs
 - Possède ses propres liens
 - Réalise telle ou telle opération définie dans la classe à un certain moment
 - Est soumis aux contrats définis par les responsabilités de la classe

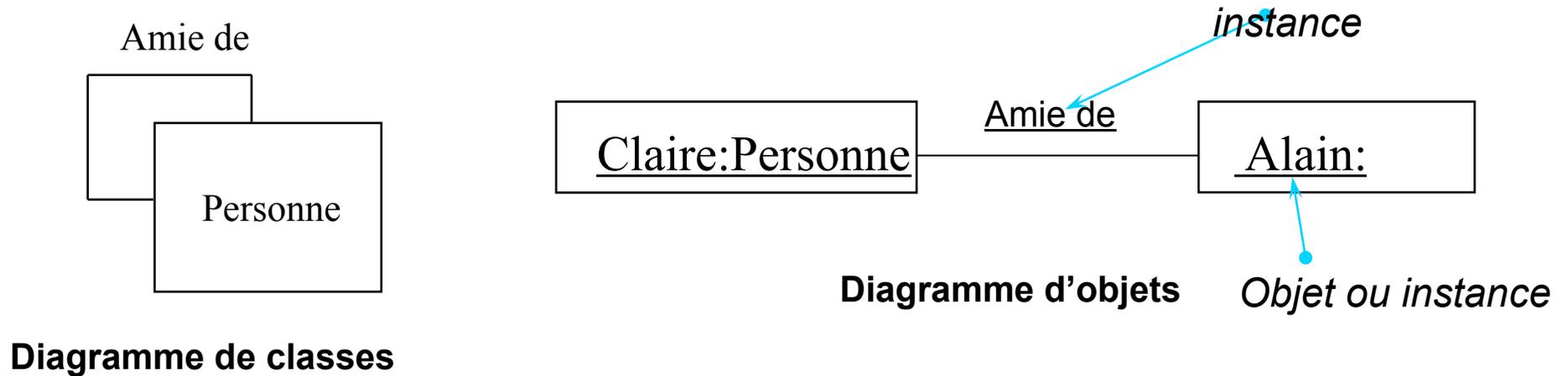
L'instance et l'objet

Les objets sont les instances de classes.

Cependant, une instance n'est pas forcément une manifestation d'une classe. Par exemple, l'instance d'une association n'est pas un objet.

Une instance (d'une classe ou d'une quelconque entité) est représentée en soulignant le nom.

Dans un système les noms des instances doivent tous être différents.



Quelle est la différence entre : une école pouvant accueillir 200 étudiants et l'école où est inscrit mon ami peut accueillir 200 étudiants.

Exemples d'objet et représentation du nom:

Instances nommées

Claire : Adhérent

Claire

Instance orpheline

agent:

Instances anonymes

:MultiMédia::FluxAudio

:Adhérent

Instance multiple

:CodeClé

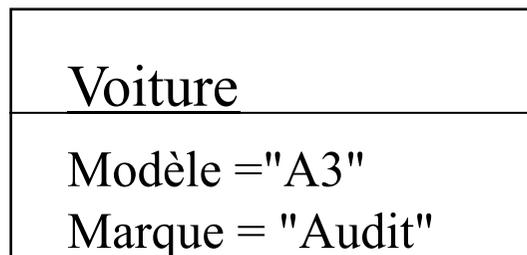
Opérations et état des objets

Les opérations définies dans les classes peuvent s'appliquer sur les objets.

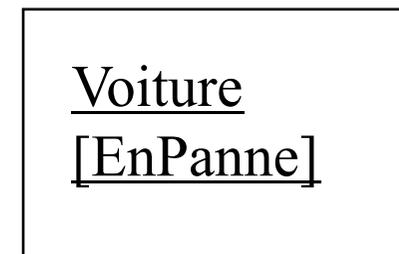
Par exemple, si l'opération `achat()` est définie dans la classe `Transaction` alors cette opération peut être utilisée par l'objet `maTransaction` comme suit : `maTransaction.achat()`.

L'état d'un objet comprends les valeurs des attributs de la classe ainsi que celles de ses agrégats. L'état d'un objet est dynamique. Quand celui-ci est visualisé, il correspond aux valeurs de l'objet à un instant et à un endroit donnés.

Instance avec les valeurs des attributs



Instances avec état explicite



Notations avancées

Objets actifs : instances des classes actives

T: leThreadPrincipal

Objet stéréotypé

« exception »
e: IOException

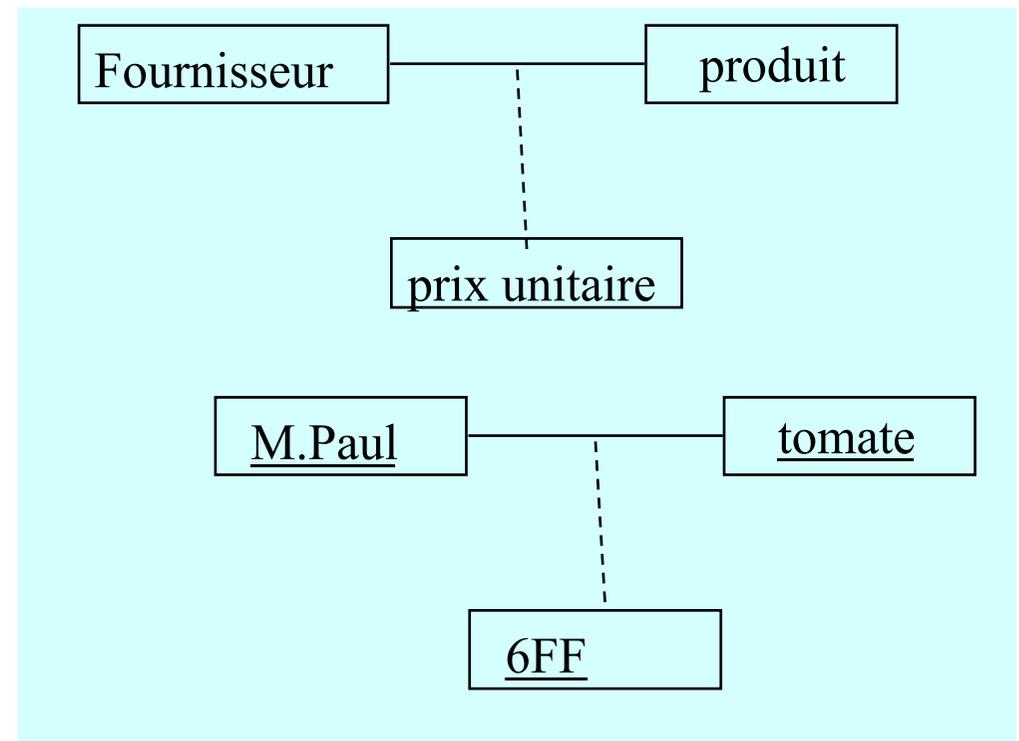
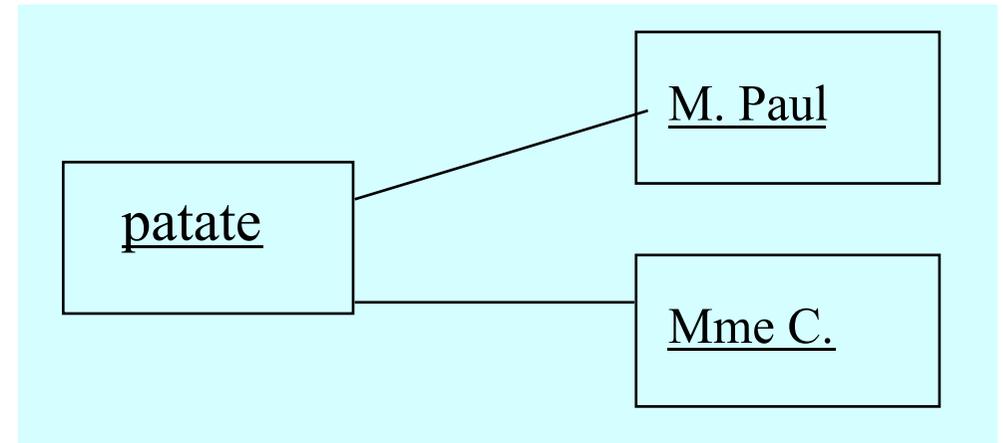
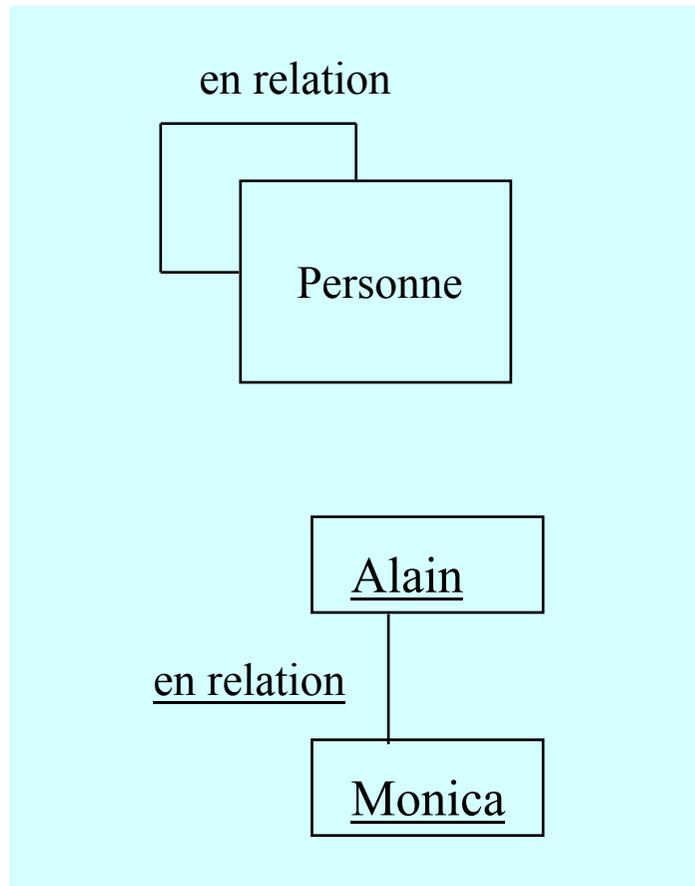
Objet stéréotypé : deux types de stéréotypes sont définis entre les classes et les objets :

- InstanceOf : objet client est une instance de la classe
- Instantiate : la classe cliente peut créer des instances de la classe fournisseur

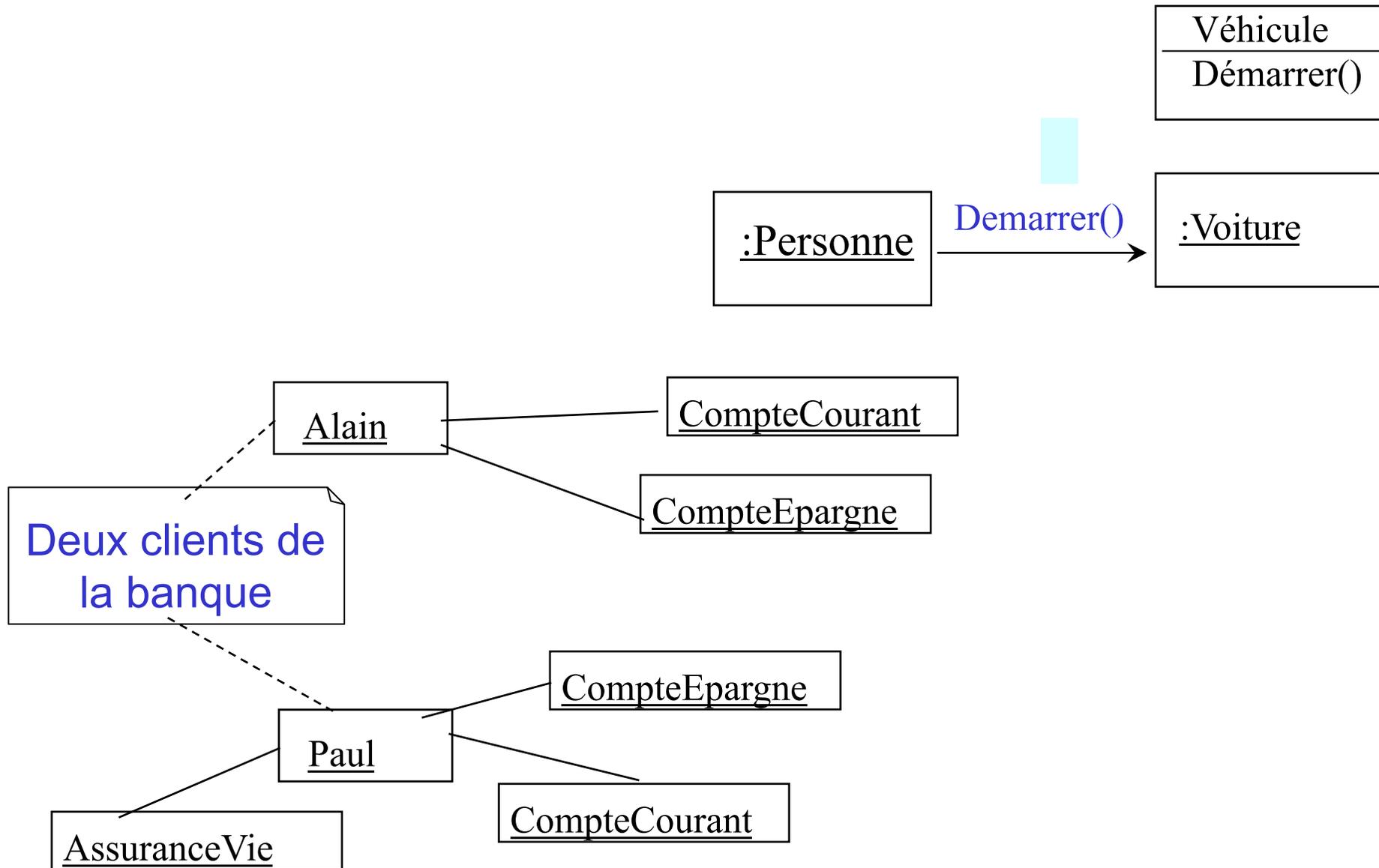
D'autres stéréotypes sont utilisés entre les objets

- become : indique que les deux instances représentent le même objet à des instants différents
- copy : indique que l'objet cible est une copie de l'objet source

Exemples de diagrammes de classes et d'objets



Exemples d'objets



Exemple de diagramme des classes

L'application consiste à concevoir des robots qui se déplacent dans un environnement composé de zones, de murs et de portes.

Donner le diagramme des objets décrit par la situation suivante :

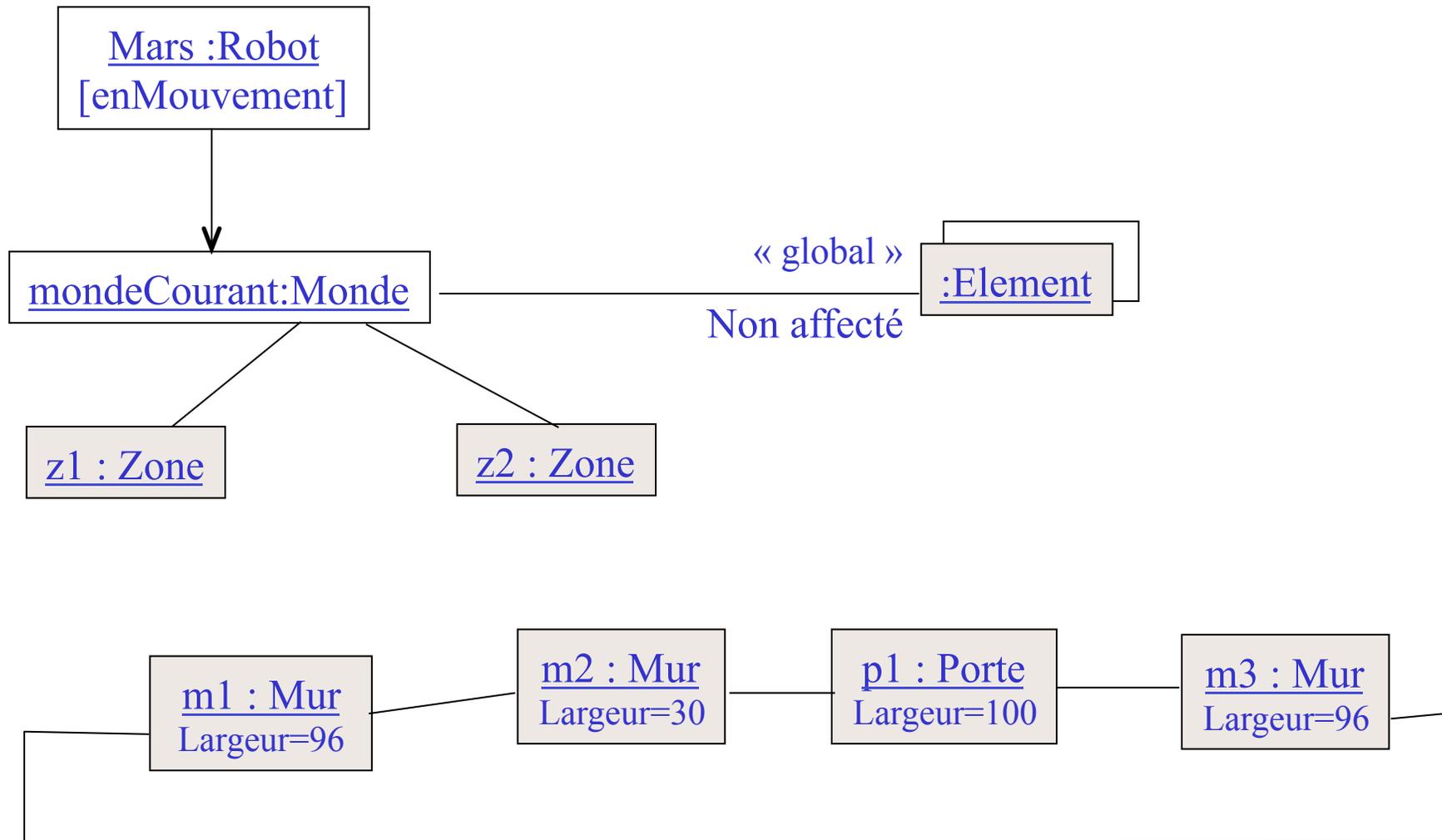
Le robot Mars est dans l'état « enMouvement ». Il est lié à une instance mondeCourant de la classe Monde décrivant les mondes possibles ou peut évoluer le robot. Le robot peut manipuler des éléments qui ne sont pas encore connus. Ces éléments sont évidemment liés au monde. Les éléments sont marqués comme étant faisant partie de l'état global du robot.

A l'instant qui nous intéresse, Mars est en mouvement et mondeCourant est lié à deux zones : la zone z1 non détaillée et la zone z2.

z2 est composée de trois murs m1, m2 et m3 et d'une porte p1. Chacun des murs indique ses liens avec les portes et les autres murs. m1 est lié à m2 et m3, m2 est lié à p1 et m3 est lié à p1. Les largeurs des murs et des portes sont connues.

Donner le diagramme des objets associé à cette situation

Solution :

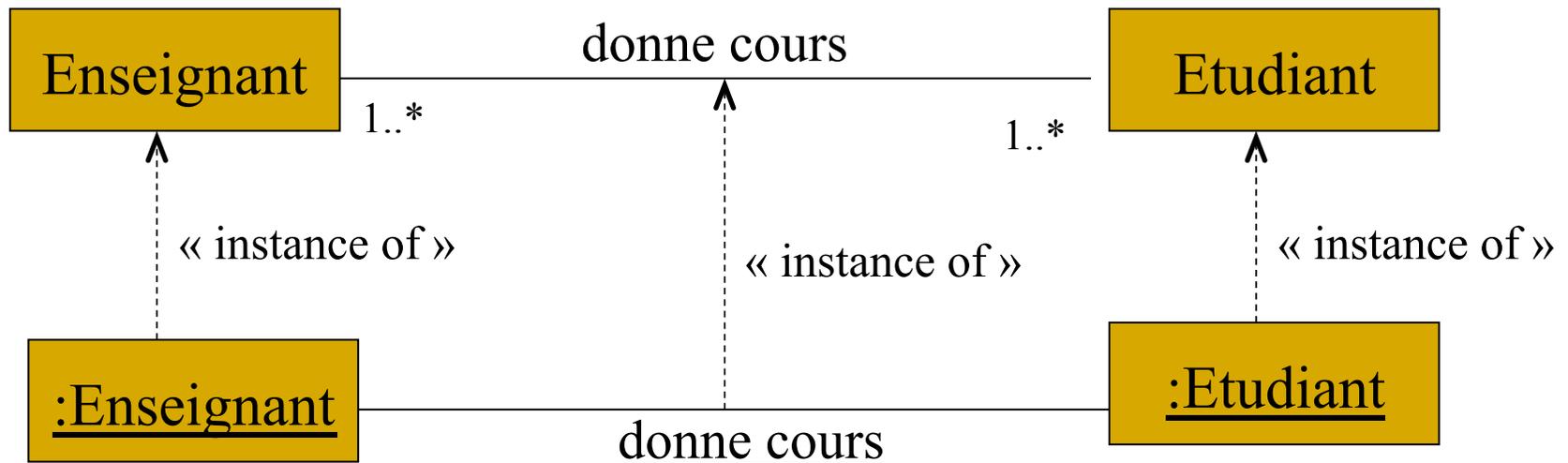


Relation de dépendance d'instanciation

Elle décrit la relation de dépendance entre

- une classe et un de ses objet
- Une association entre classes et une association entre objets

Exemple

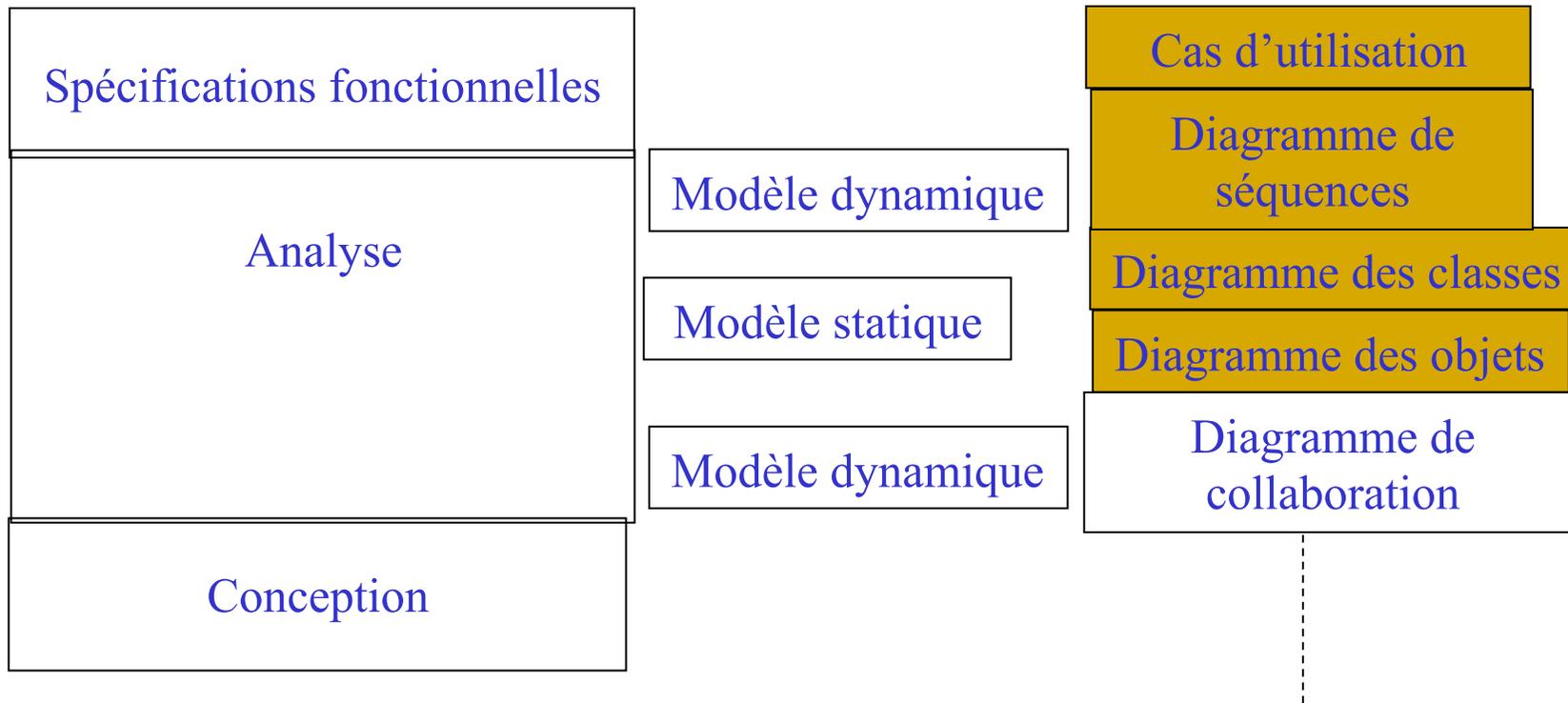


Le diagramme de séquences

A.Osmani

Site officiel UML : www.uml.org

Le diagramme des classes en UML



Sommaire

1. Les diagrammes d'interactions
2. Exemple de diagramme de séquences
3. Ligne de vie et période d'activité
4. Type de messages (simple, minuté, dérobant, synchrone, asynchrone)
5. Branchements et récursivité
6. Création et activation d'un objet
7. Diagramme de séquences

Les diagrammes d'interaction

Ensemble d'objets et leurs relations, en incluant les messages échangés entre eux.

Les diagrammes de séquences et les diagrammes de collaboration constituent les diagrammes d'interaction.

Ils concernent la partie dynamique du système.

Le diagramme de séquence fait apparaître l'ordre temporel et le diagramme de collaboration l'organisation des objets qui envoient et reçoivent des messages.

Les deux diagrammes sont isomorphes.

Diagramme de séquences

Le diagramme de séquences (comme les diagrammes de collaboration, d'états-transition, d'interaction et les cas d'utilisation) décrit les aspects dynamiques du système.

Il décrit un ensemble d'objets et les messages qui peuvent circuler entre eux par ordre chronologique.

Le diagramme de séquences a deux dimensions :

1. L'axe verticale représente le temps
2. L'axe horizontale représente les différentes instances échangeant des messages (appel de méthodes) . La disposition des objets sur l'axe horizontal n'a pas de conséquence pour la sémantique du diagramme.

Les dimensions peuvent éventuellement être inversées.

Exemple de diagramme de séquences

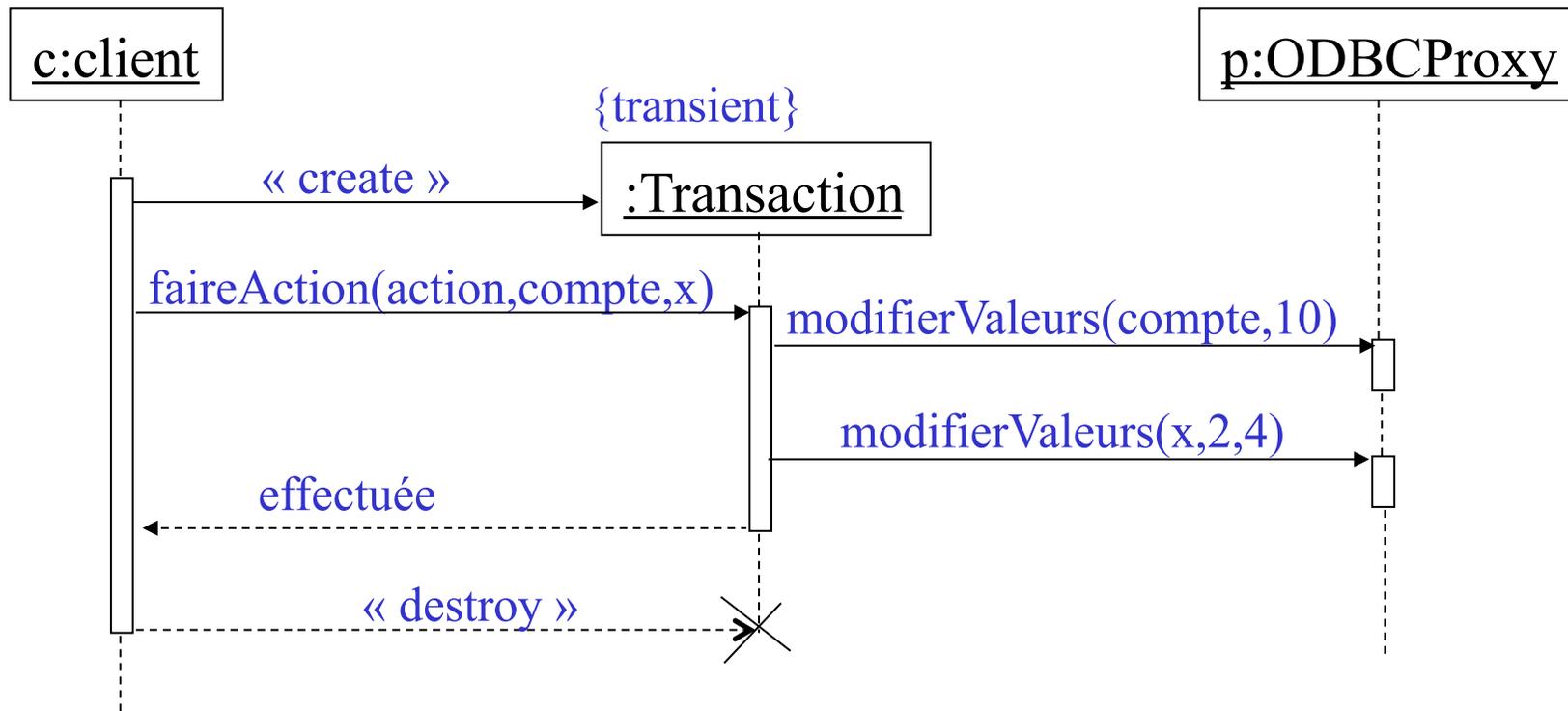


Diagramme de séquences

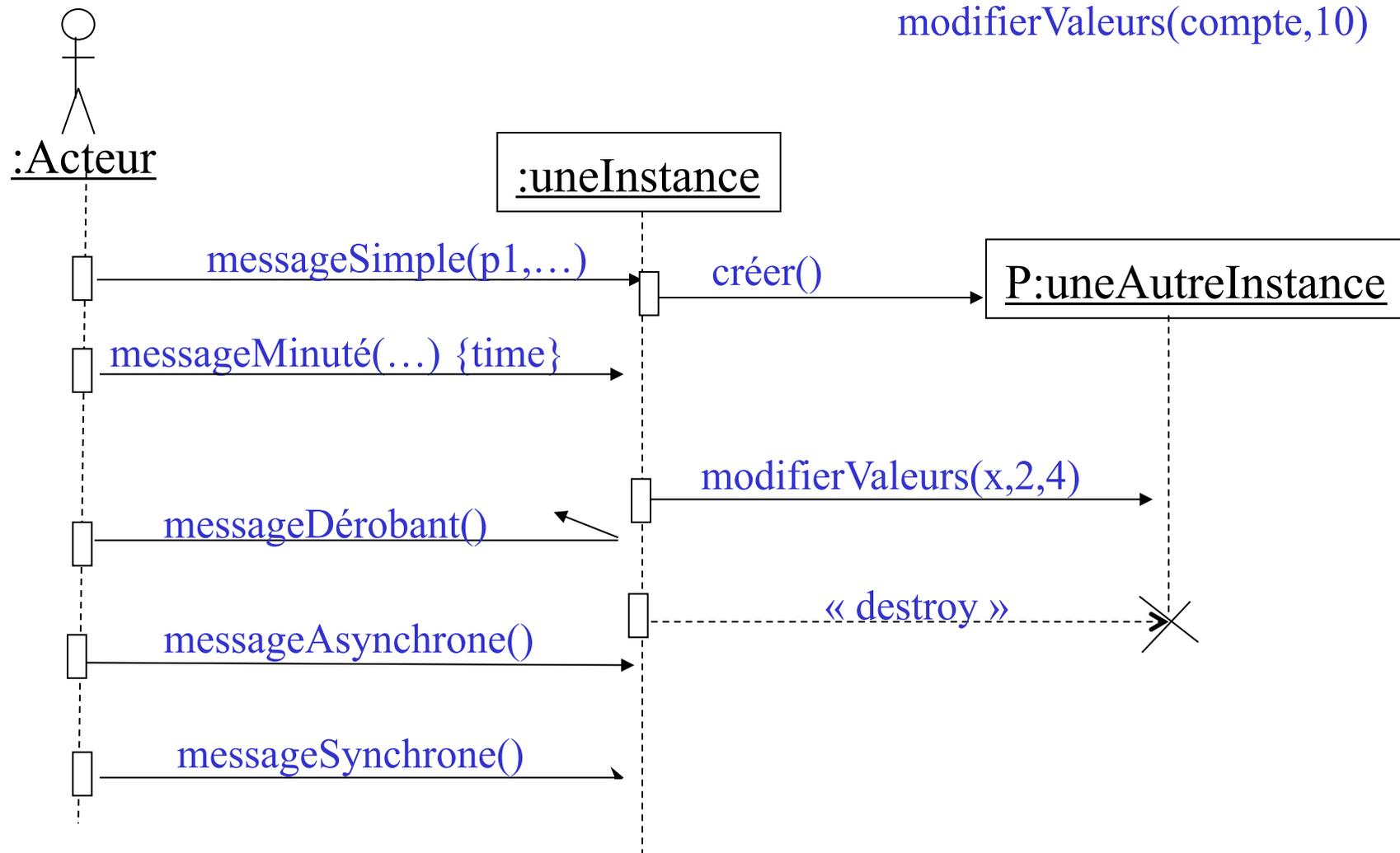
Les diagrammes de séquences permettent de représenter des collaborations entre objets selon un point de vue temporel, on y met l'accent sur la chronologie des envois de messages.

Contrairement au diagramme de collaboration, on n'y décrit pas le contexte ou l'état des objets, la représentation se concentre sur l'expression des interactions.

Les diagrammes de séquences peuvent servir à illustrer un cas d'utilisation.

L'ordre d'envoi d'un message est déterminé par sa position sur l'axe vertical du diagramme ; le temps s'écoule "de haut en bas" de cet axe.

Types de symboles utilisés



Types de symboles utilisés

UML propose un certain nombre de stéréotypes graphiques pour décrire la nature du message

Message simple

Aucune spécification des caractéristiques d'envoi ou de réception particulière n'est précisée.

Message minuté (timeout)

Il bloque l'expéditeur pendant un temps donné (qui peut être spécifié dans une contrainte), en attendant la prise en compte du message par le récepteur. L'expéditeur est libéré si la prise en compte n'a pas eu lieu pendant le délai spécifié.

Types de symboles utilisés

Message synchrone

Bloque l'expéditeur jusqu'à prise en compte du message par le destinataire. Le flot de contrôle passe de l'émetteur au récepteur (l'émetteur devient passif et le récepteur actif) à la prise en compte du message.

Message asynchrone

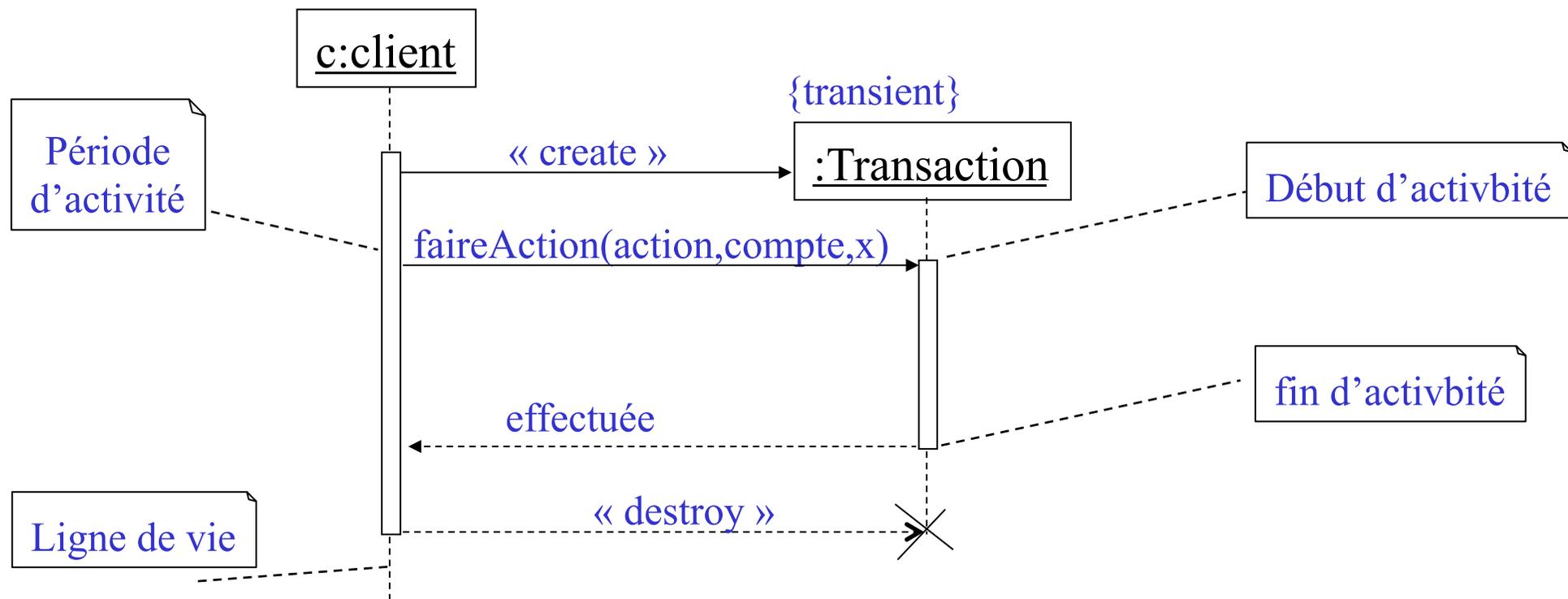
N'interrompt pas l'exécution de l'expéditeur. Le message envoyé peut être pris en compte par le récepteur à tout moment ou ignoré (jamais traité).

Message déroband

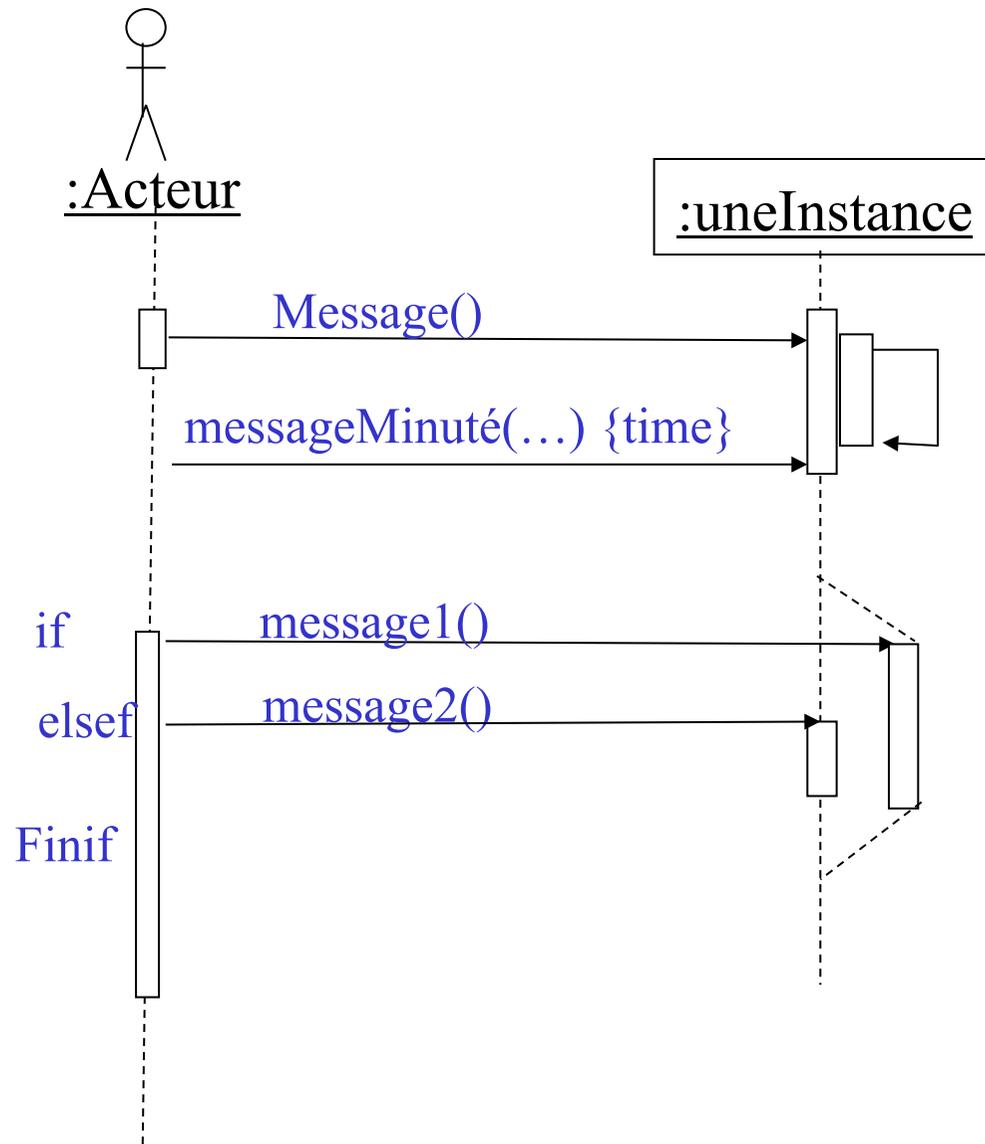
N'interrompt pas l'exécution de l'expéditeur et ne déclenche une opération chez le récepteur que s'il s'est préalablement mis en attente de ce message.

Activation d'un objet

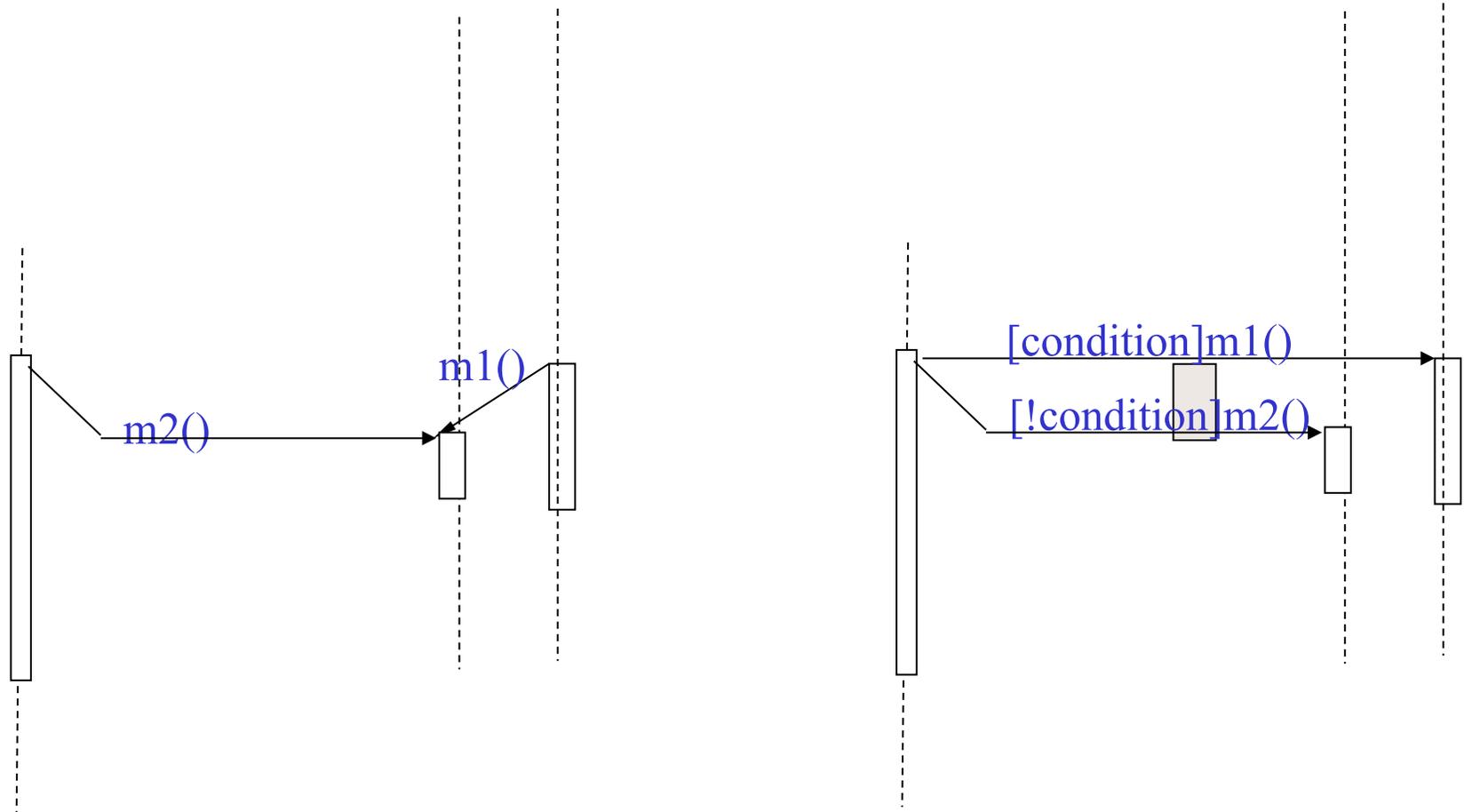
La ligne discontinue représente la ligne de vie de l'objet. La bande rectangulaire correspond à la période d'activité.



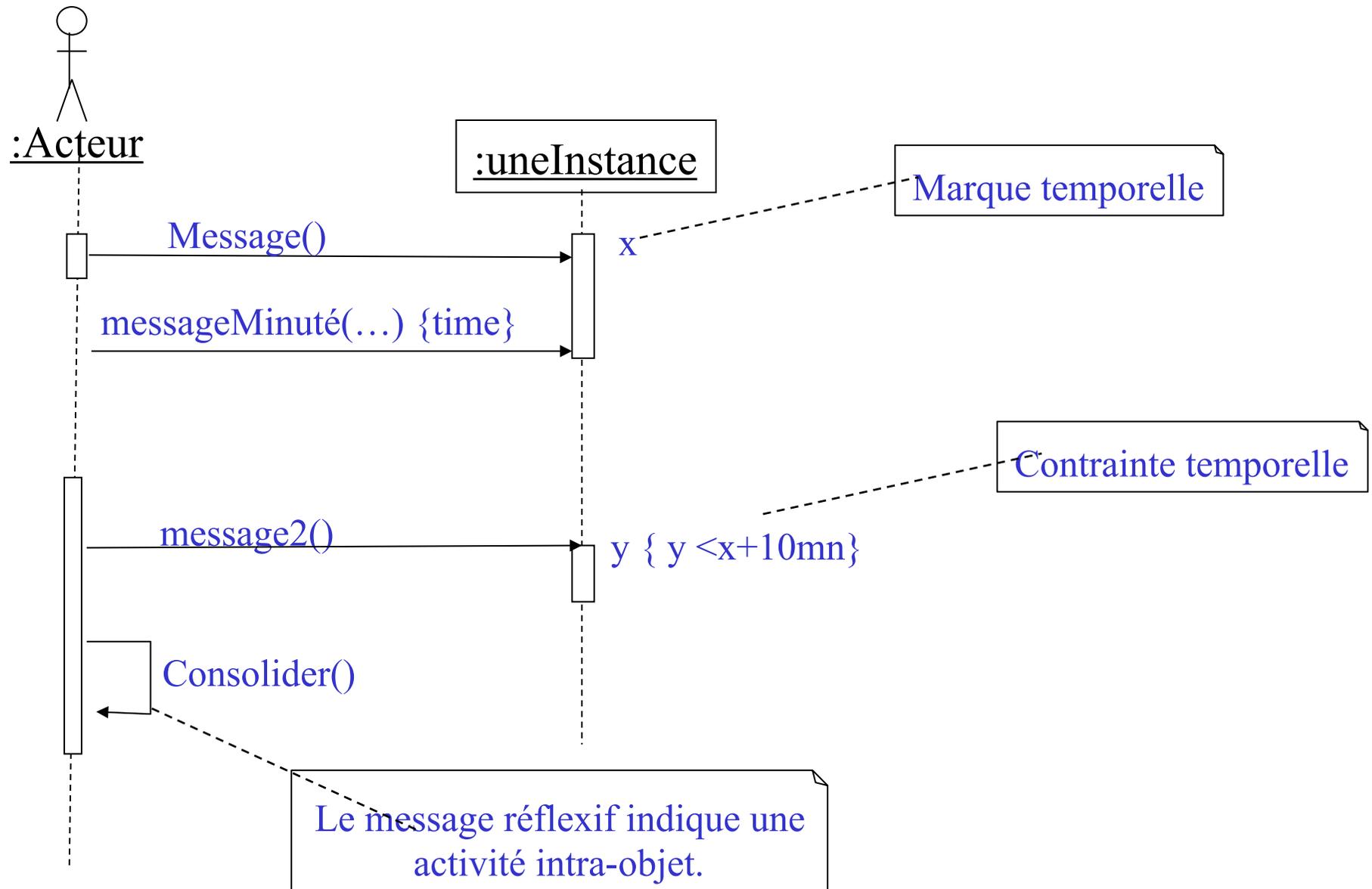
Représentation d'une récursion et d'un branchement



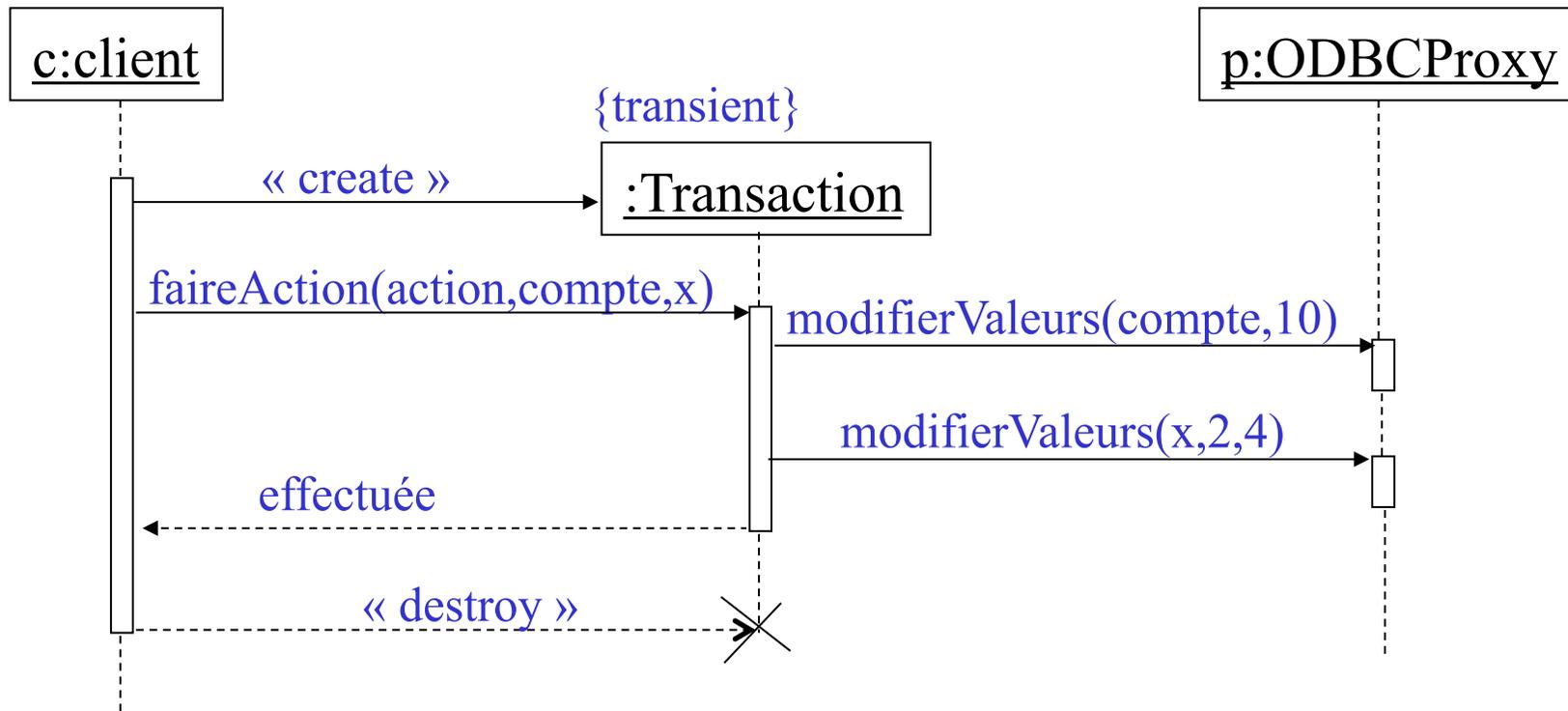
Représentation d'un branchement ou alternative concurrente



Messages réflexifs, marques temporelles



Exemple de diagramme de séquences



Exemple de diagramme de séquences

