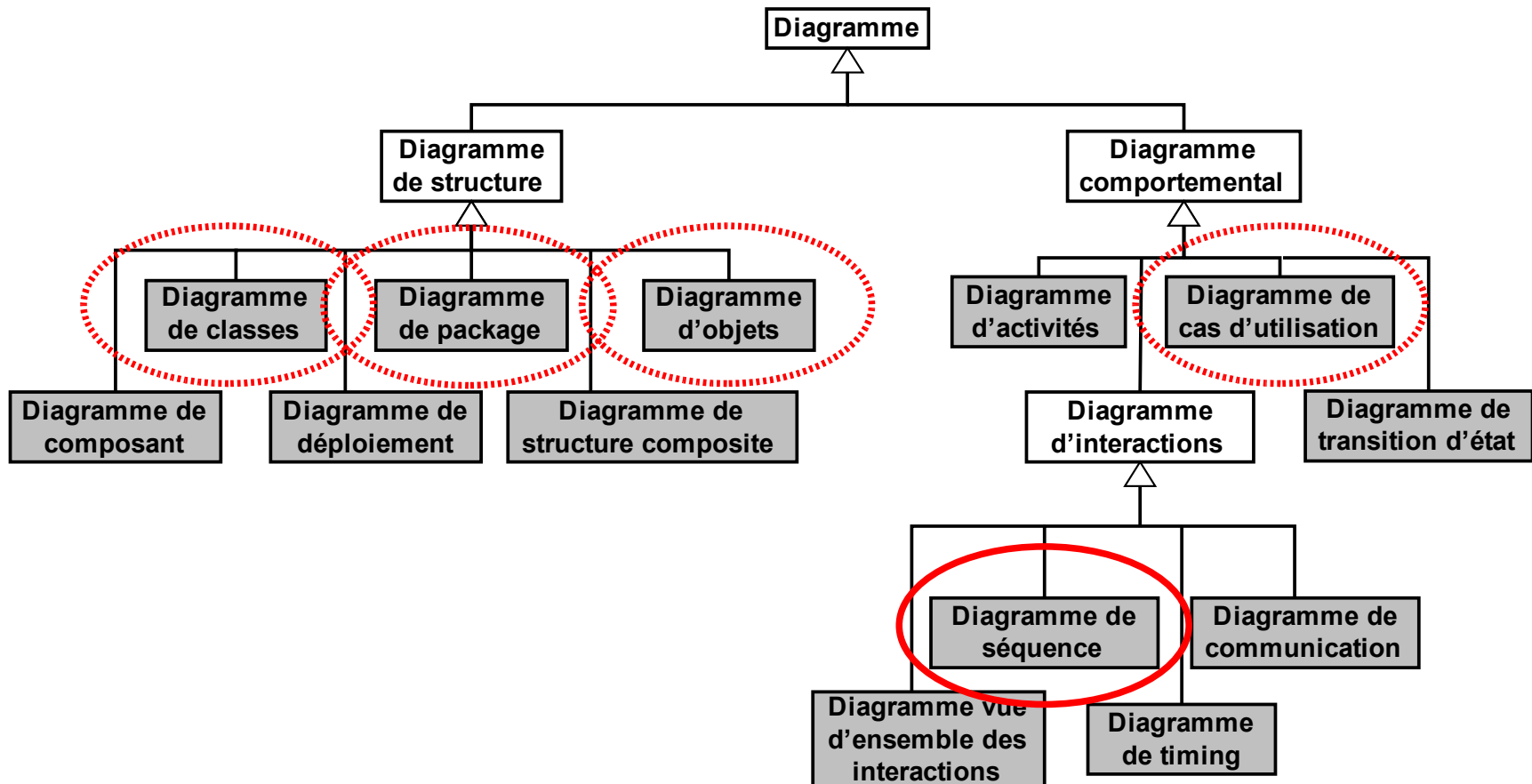




## II.6. Diagrammes d'interaction

# Introduction

## ■ Les diagrammes d'UML



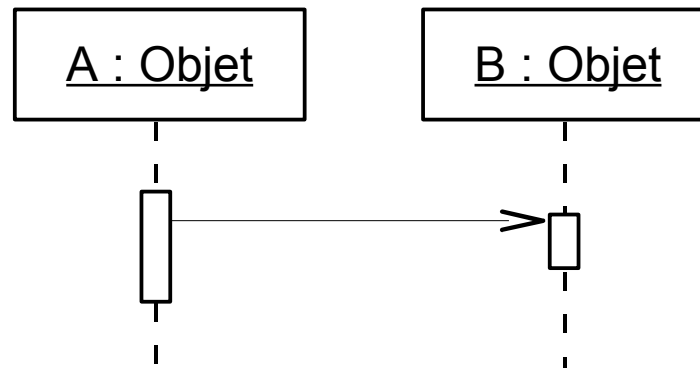
# Diagrammes d'interaction

## ■ Introduction

Modéliser la façon dont les groupes d'objets collaborent pour réaliser un comportement donné.

### Diagrammes de séquence

- Montrent les interactions entre objets selon un point de vue temporel
- Le contexte des objets n'est pas représenté

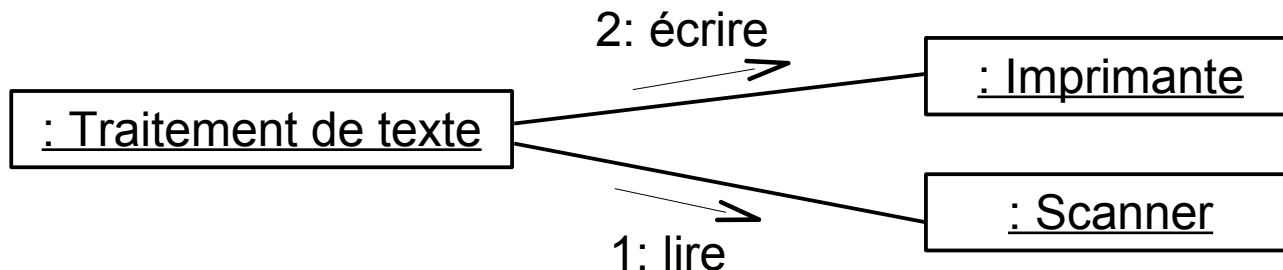


# Diagrammes d'interaction

## ■ Introduction

### Diagrammes de collaboration / communication

- Montre les interactions entre objets en insistant sur la structure spatiale statique
- Exprime à la fois le contexte de groupe d'objets et l'interaction entre ces objets
- Extension du diagramme d'objets





## II.6. Diagrammes d'interaction

### 1. Diagrammes de séquence

# Diagrammes de séquence

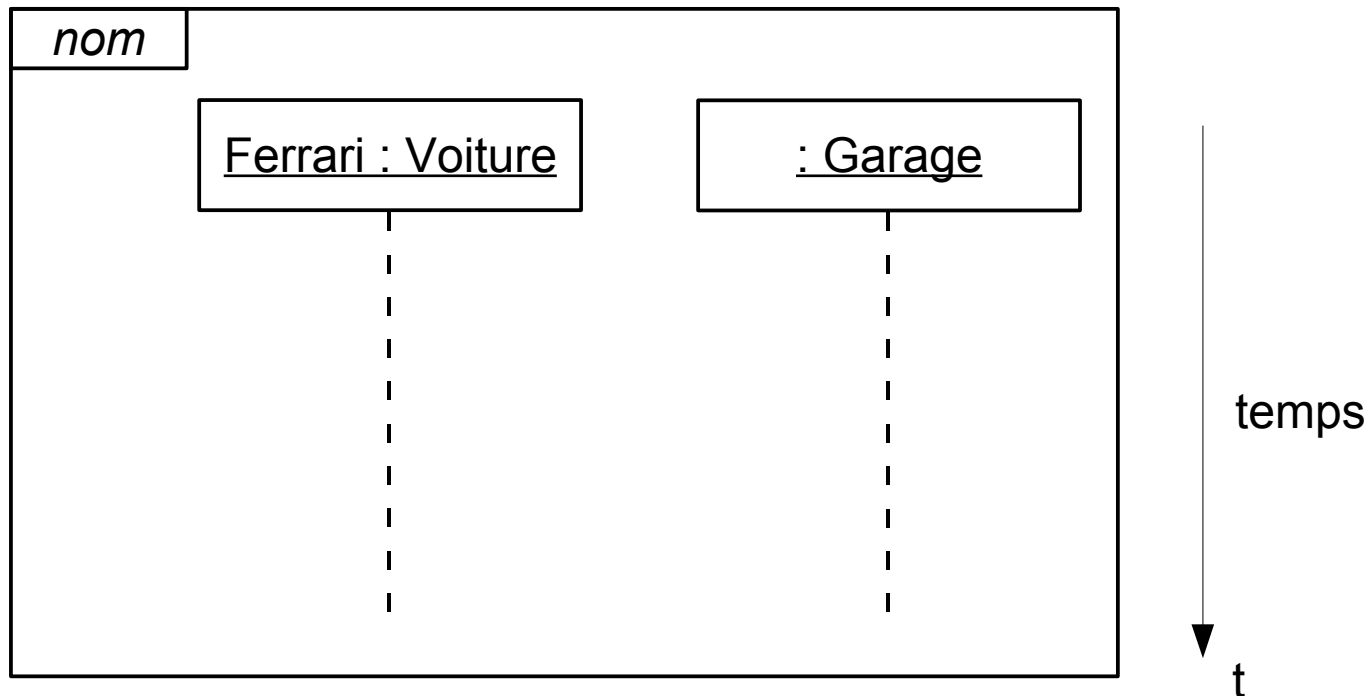
## ■ Concepts principaux :

- 1 Les participants (le plus souvent des objets)
  - Une ligne de vie
  - Des zones d'activation
- 2 Les messages
  - L'opération et éventuellement ses paramètres
  - Éventuellement son résultat
- 3 Des structures de contrôle
  - Alt : conditionnelle
  - Loop : boucle
  - Réf : référence à un autre diagramme de séquence (=appel de fonction)
  - Etc.

# Diagrammes de séquence

## ■ Représentation des acteurs

- Rectangle + Ligne de vie
- `Nom_objet : nom_classe`

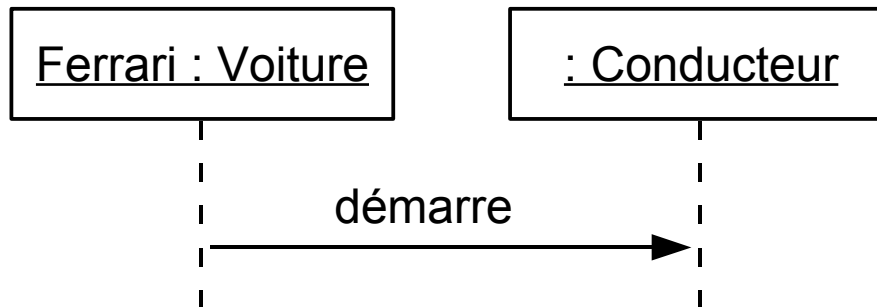


# Diagrammes de séquence

## ■ Messages

Communication entre les lignes de vie

- appel de méthode
- envoi d'un signal
- création/destruction d'une instance



attribut = nom\_signal\_ou\_op (arguments) : valeur\_retournée



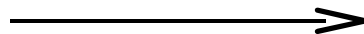
# Diagrammes de séquence

## ■ Types de messages

- 1 Appels de méthodes ou autres méthodes d'appels synchrones



- 2 Communication asynchrone : l'émetteur déclenche le stimulus et passe immédiatement à la suite de l'exécution

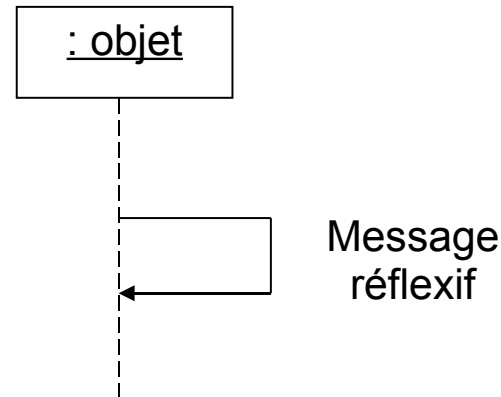


- 3 Retours de procédures (ou de méthodes) : optionnel

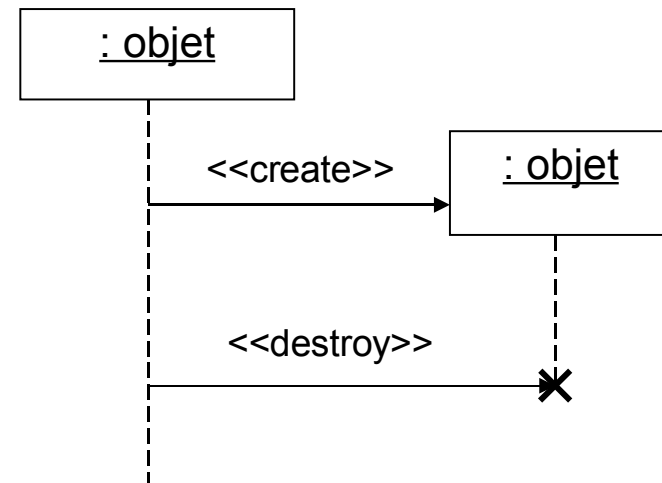


# Diagrammes de séquence

- Un objet peut s'envoyer des messages



- Un message peut entraîner la création ou la destruction d'objets

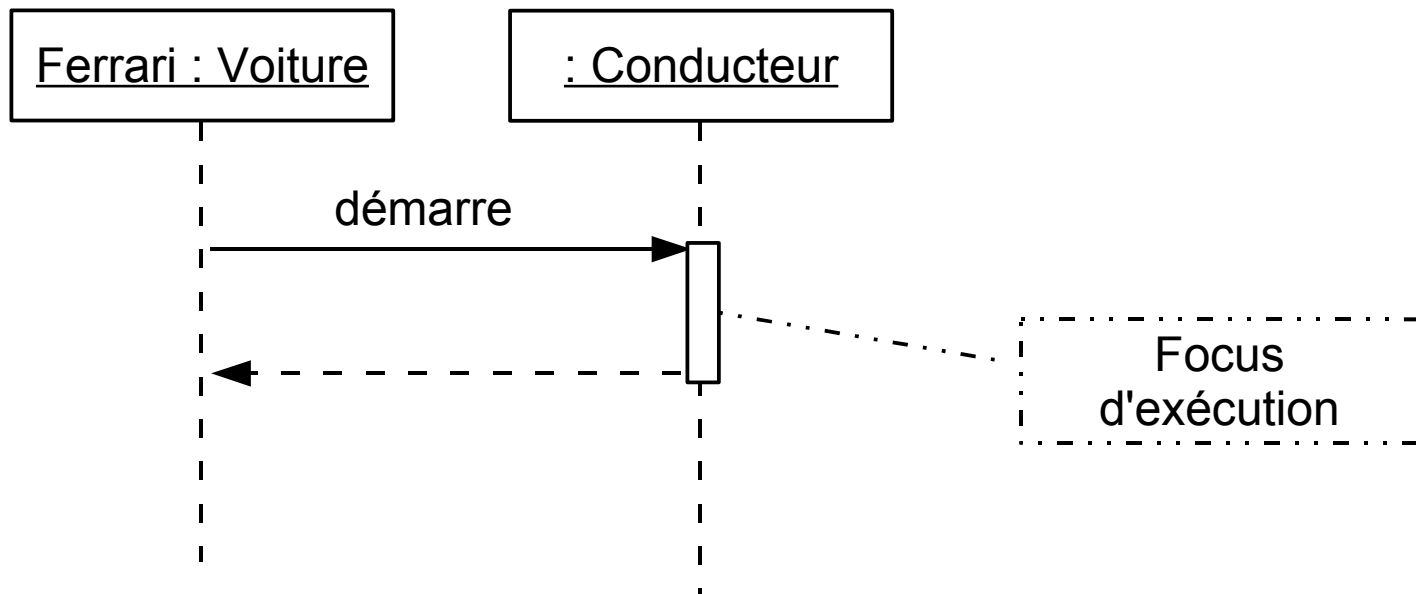


# Diagrammes de séquence

## ■ Occurrence d'exécution

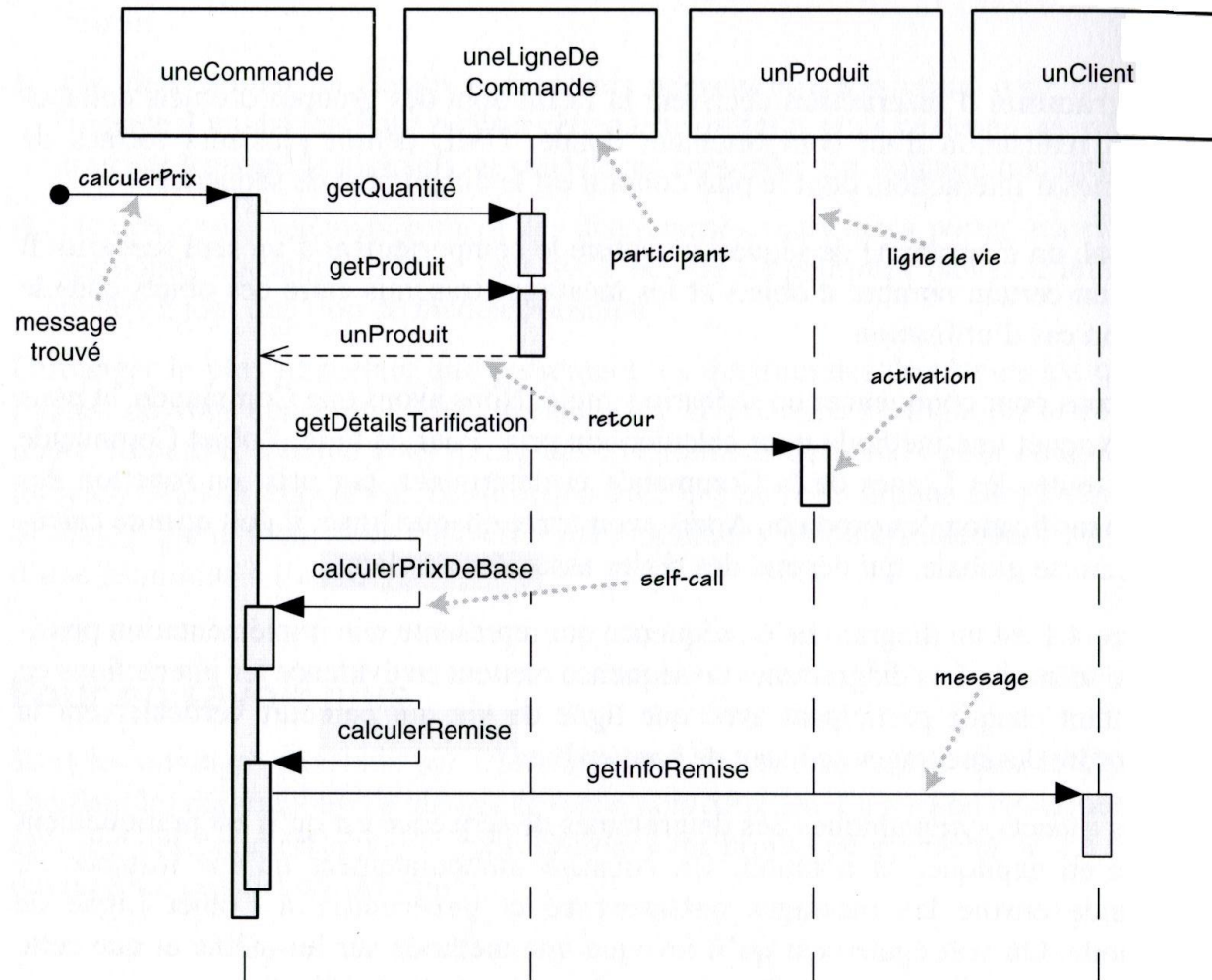
Aussi appelé « focus de contrôle »

Correspond à la durée d'activité de l'objet  $\Leftrightarrow$  temps durant lequel il est actif



# Diagrammes de séquence

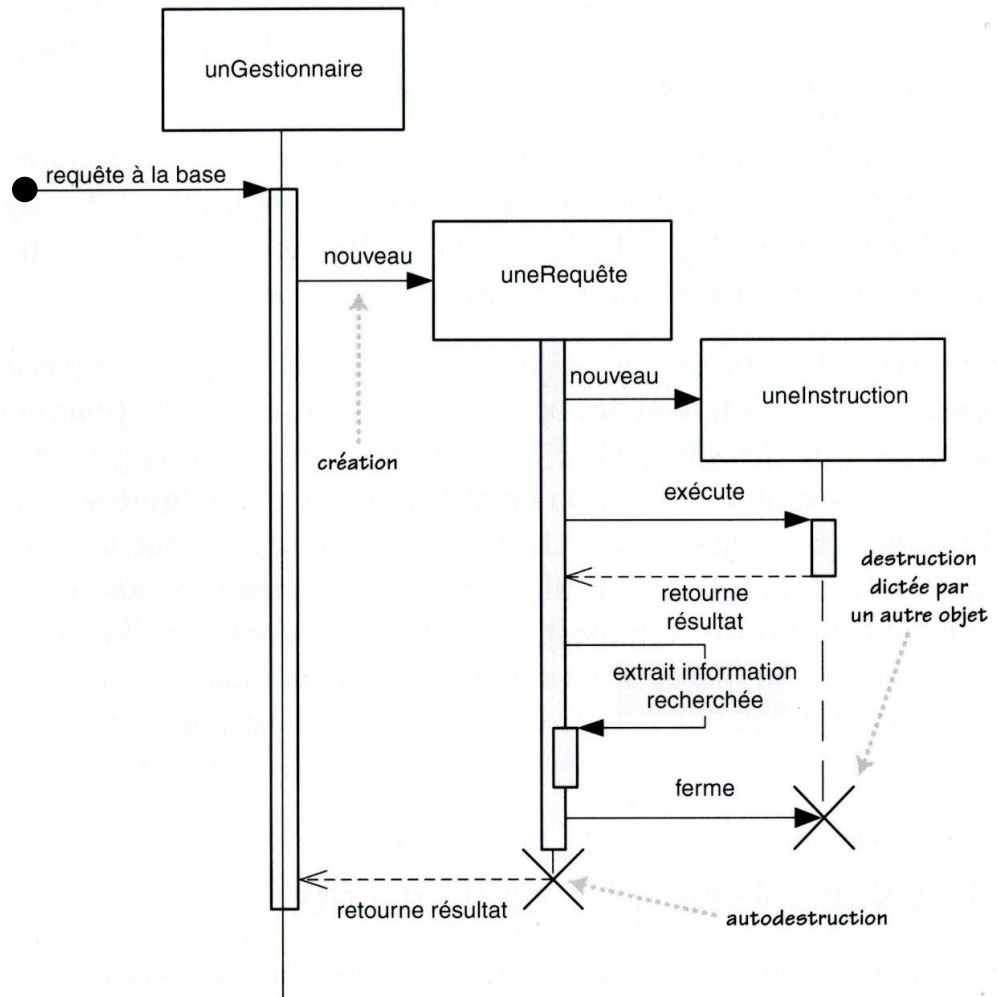
- Message trouvé = émetteur inconnu.



# Diagrammes de séquence

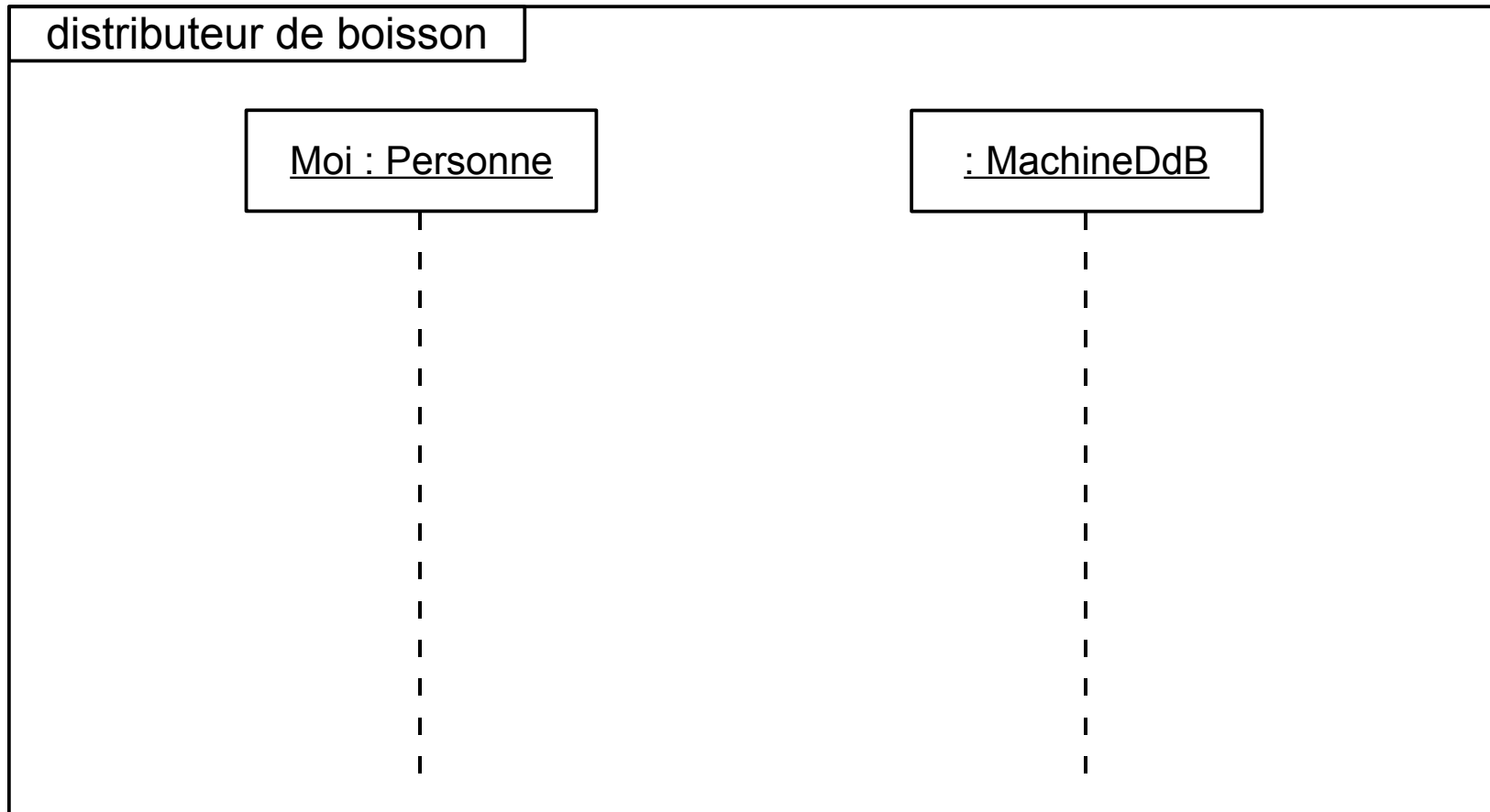
## ■ Exemple

Créations  
Destructions



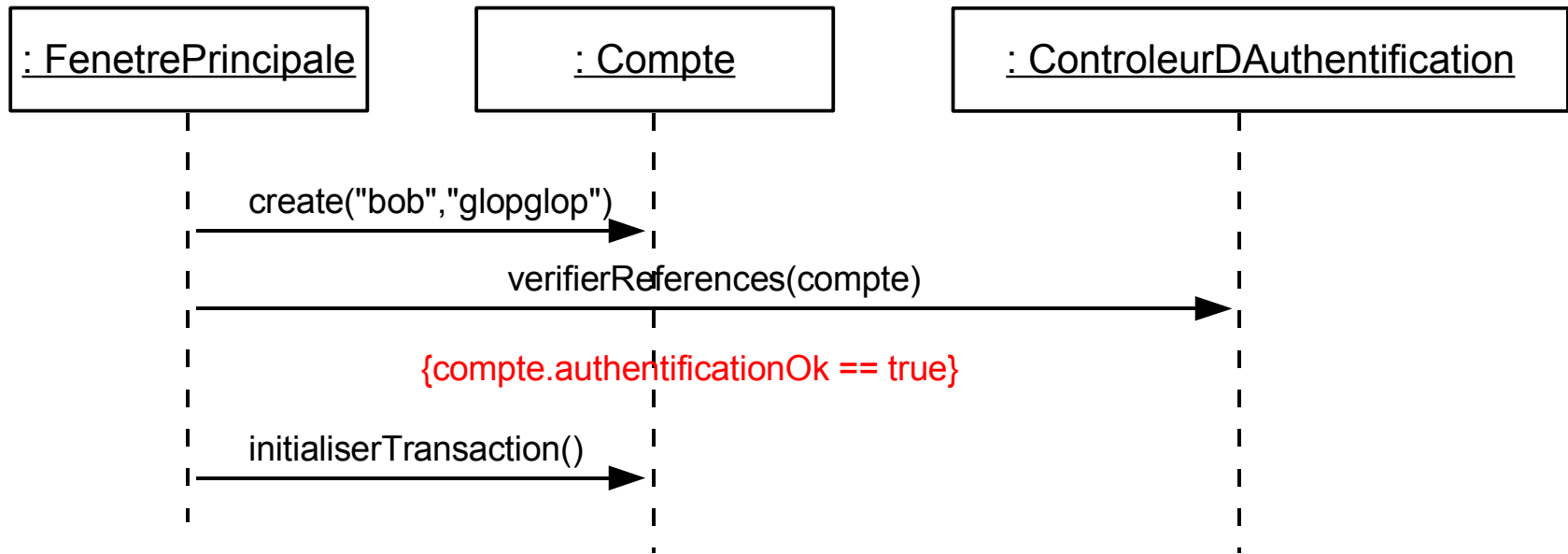
# Diagrammes de séquence

## ■ Exemple : distributeur de boisson



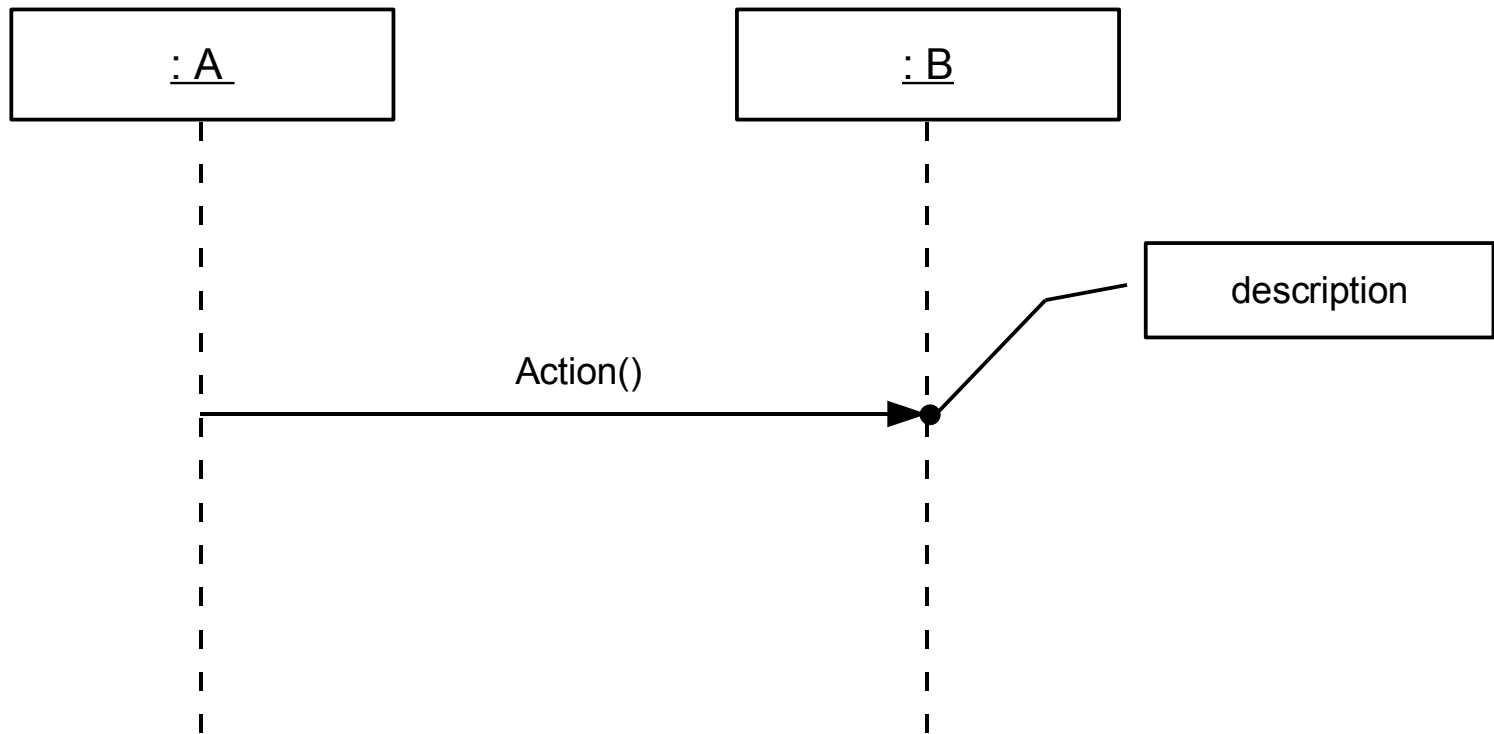
# Diagrammes de séquence

## ■ Invariants d'état



# Diagrammes de séquence

## ■ Occurrences événementielles



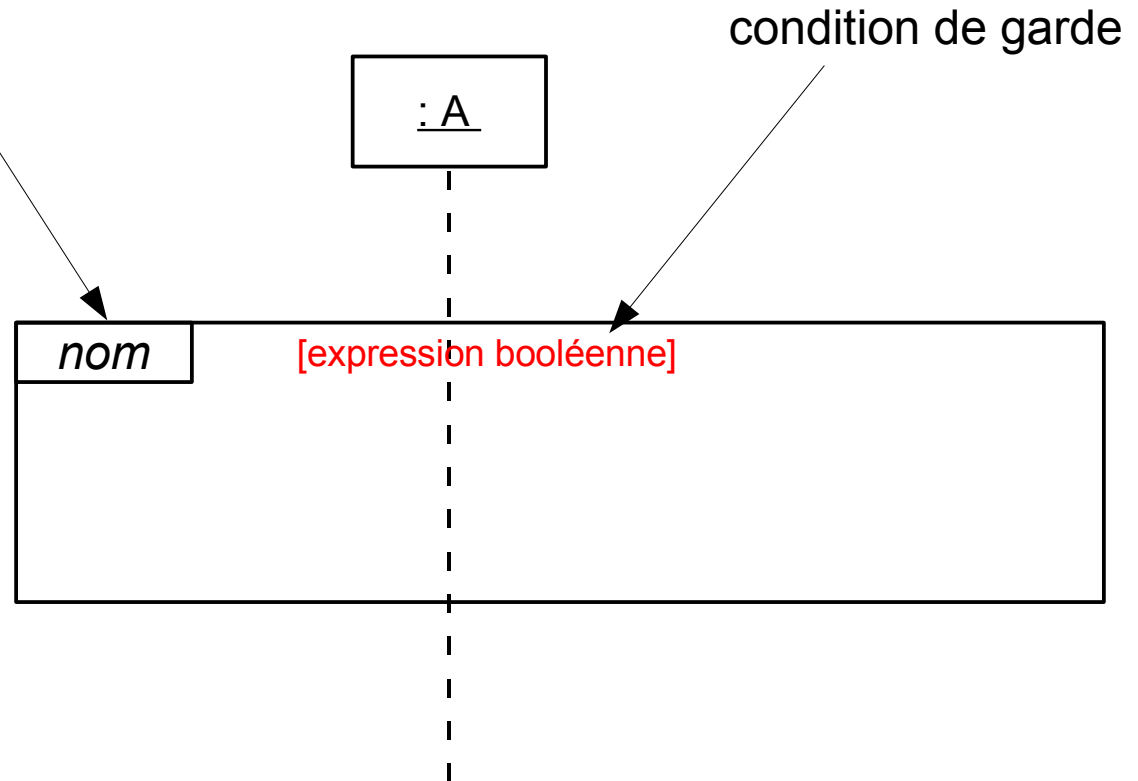


# Diagrammes de séquence

## ■ Fragments combinés

### *Types*

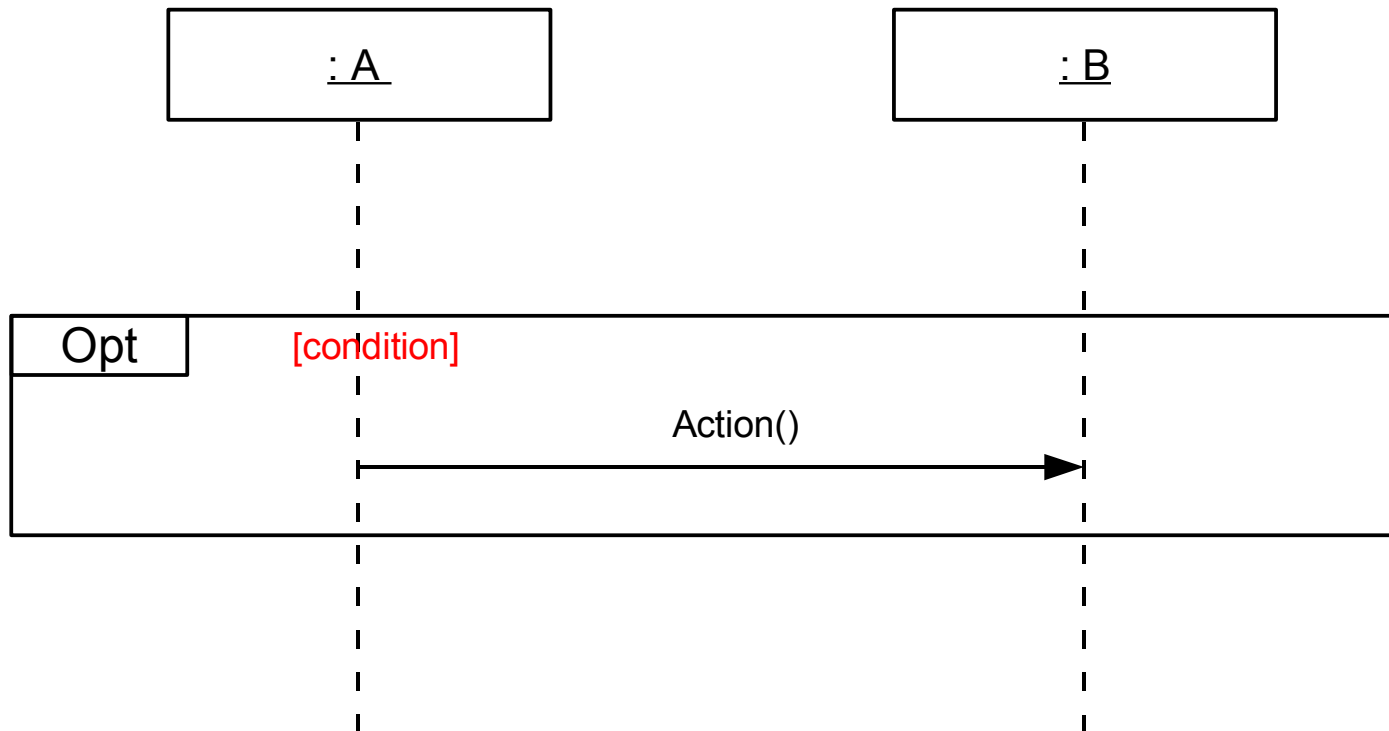
- 1 Opt
- 2 Loop
- 3 Alt
- 4 Break
- 5 Critique
- 6 Ref
- 7 ...



# Diagrammes de séquence

## ■ Fragments combinés

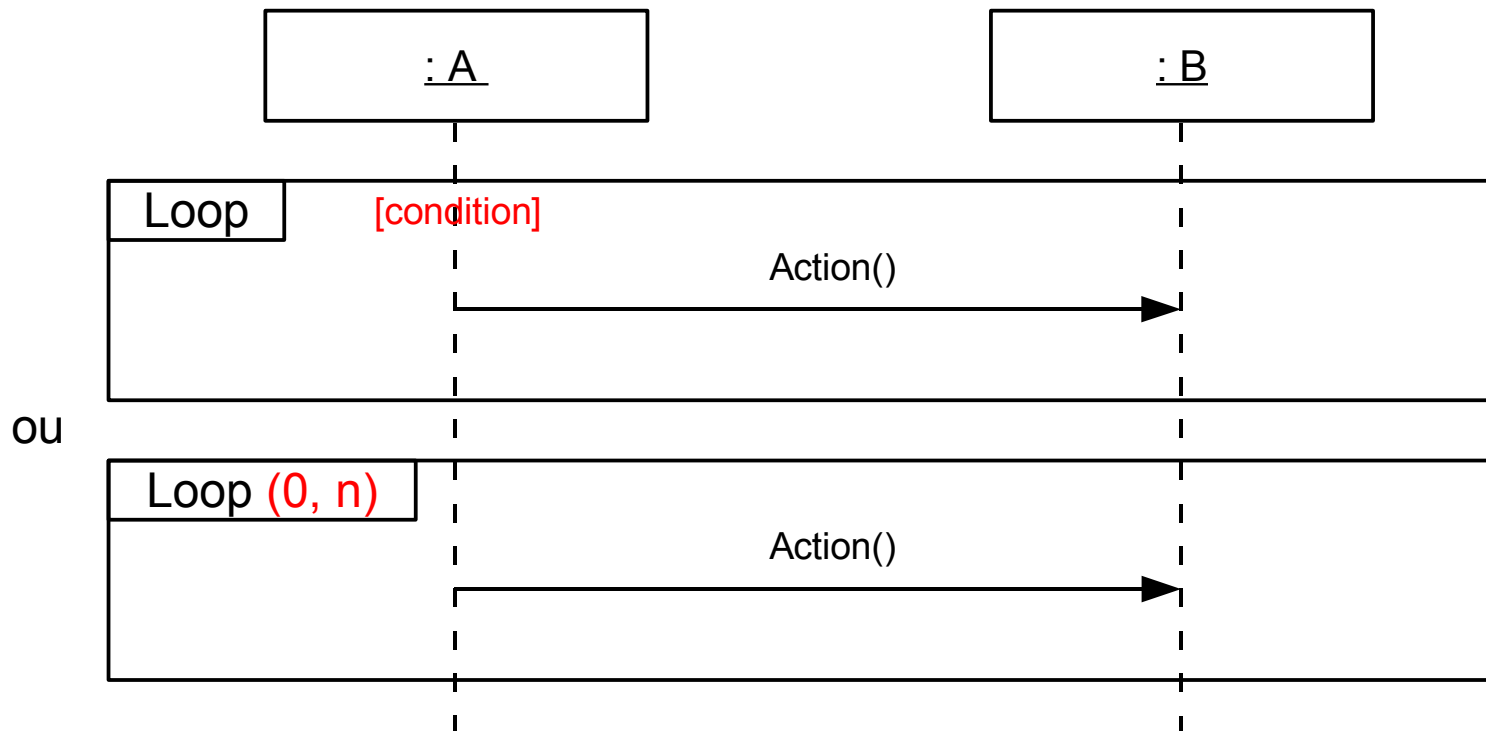
Opt : Fragment parcouru si une condition est vérifiée



# Diagrammes de séquence

## ■ Fragments combinés

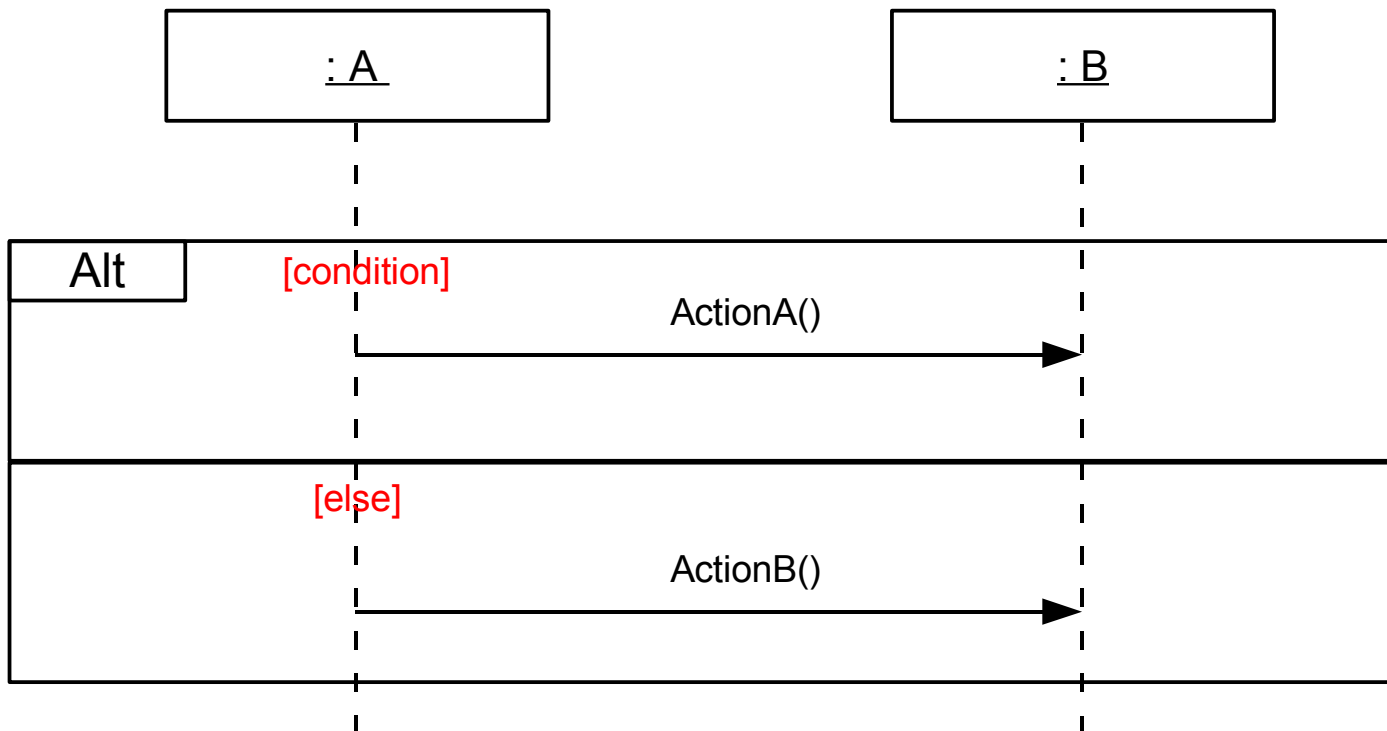
**Loop** : Répétition du fragment tant que la condition est vérifiée



# Diagrammes de séquence

## ■ Fragments combinés

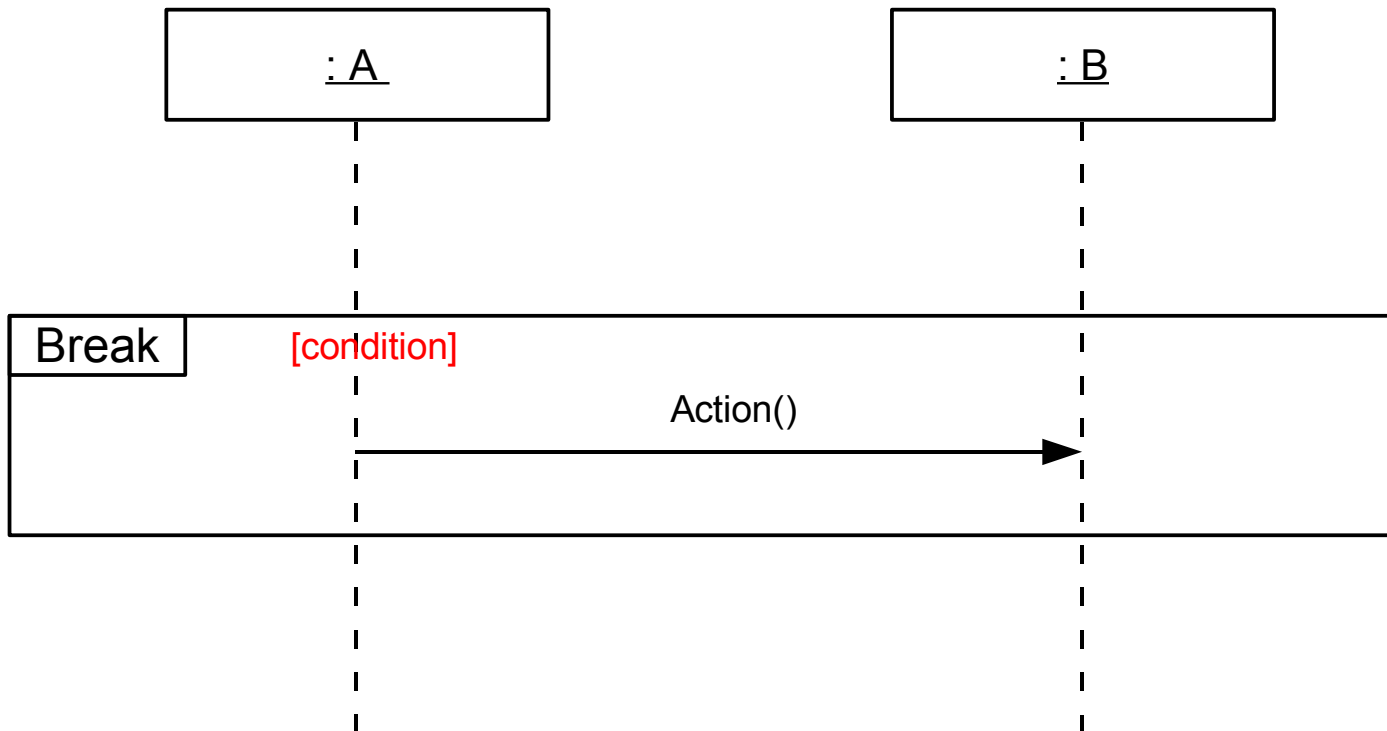
Alt : Equivalent à la structure de contrôle "*si .. alors .. sinon*".



# Diagrammes de séquence

## ■ Fragments combinés

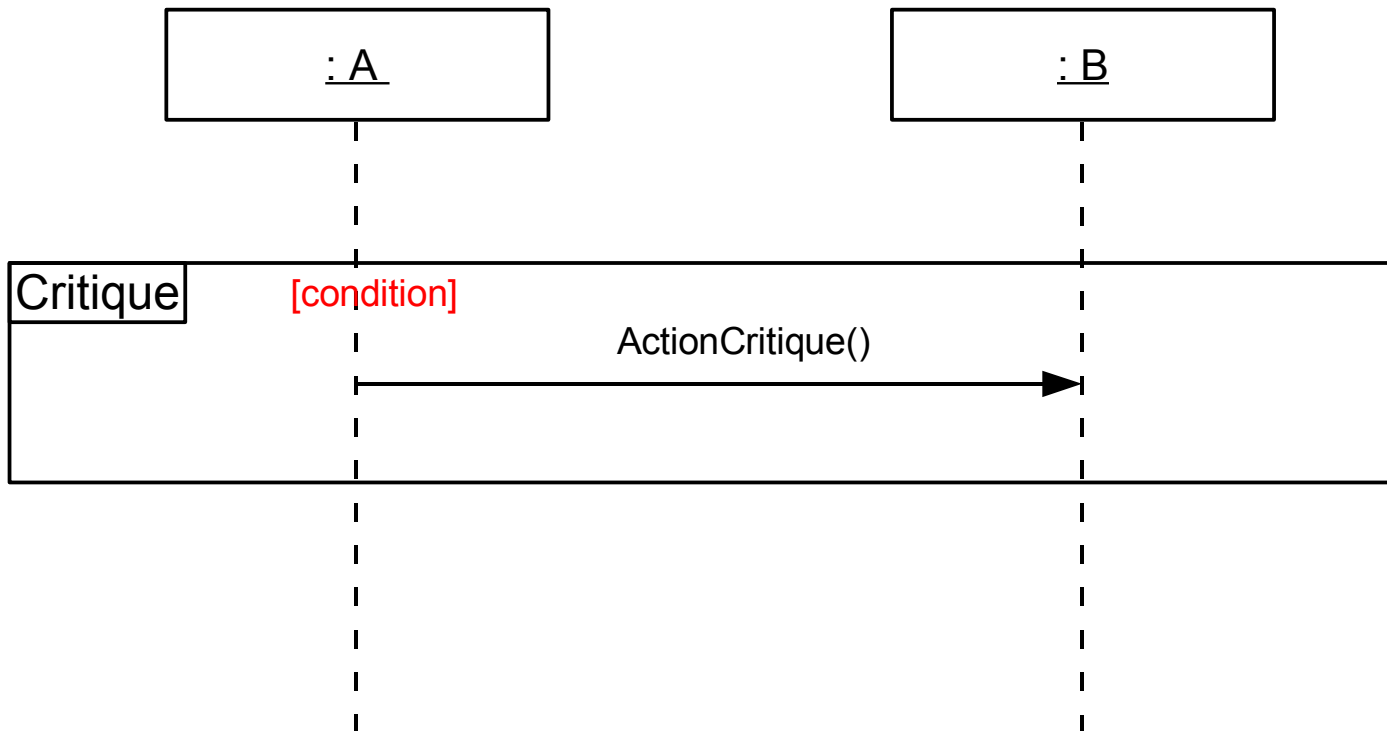
**Break** : Fragment exécuté et met fin au fragment englobant



# Diagrammes de séquence

## ■ Fragments combinés

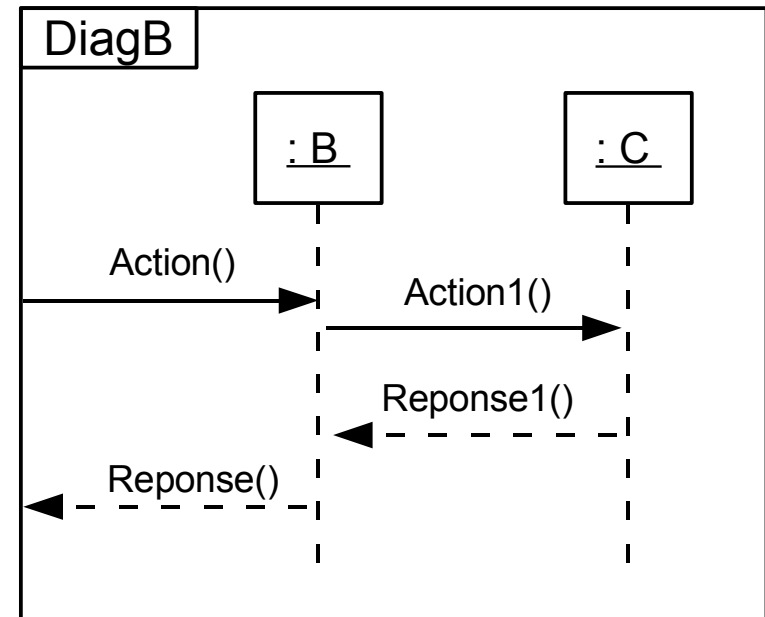
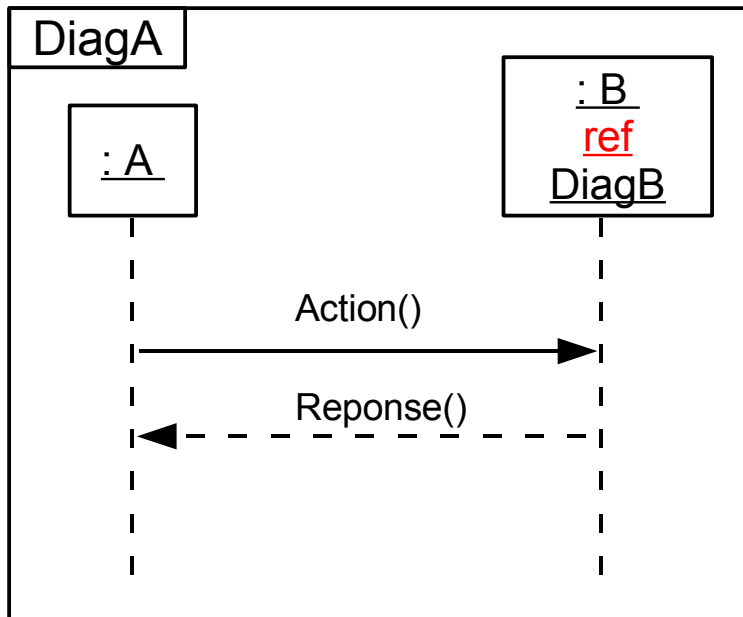
**Critique** : Fragment exécuté de façon atomique (d'un bloc)



# Diagrammes de séquence

- Décomposition : mot clef "ref"

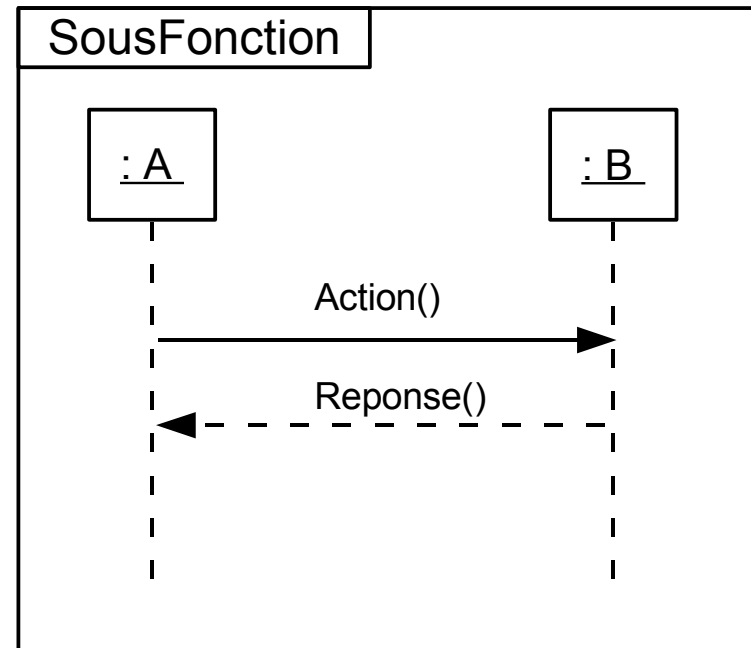
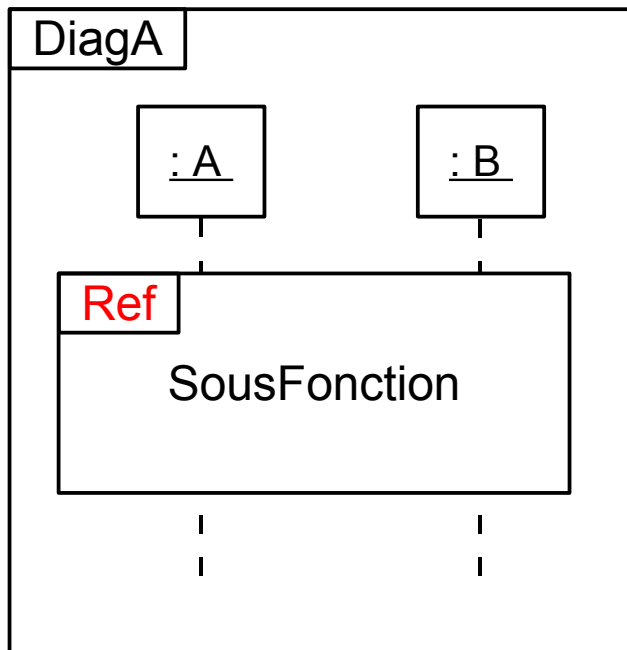
Méthode n°1



# Diagrammes de séquence

- Décomposition : mot clef "ref"

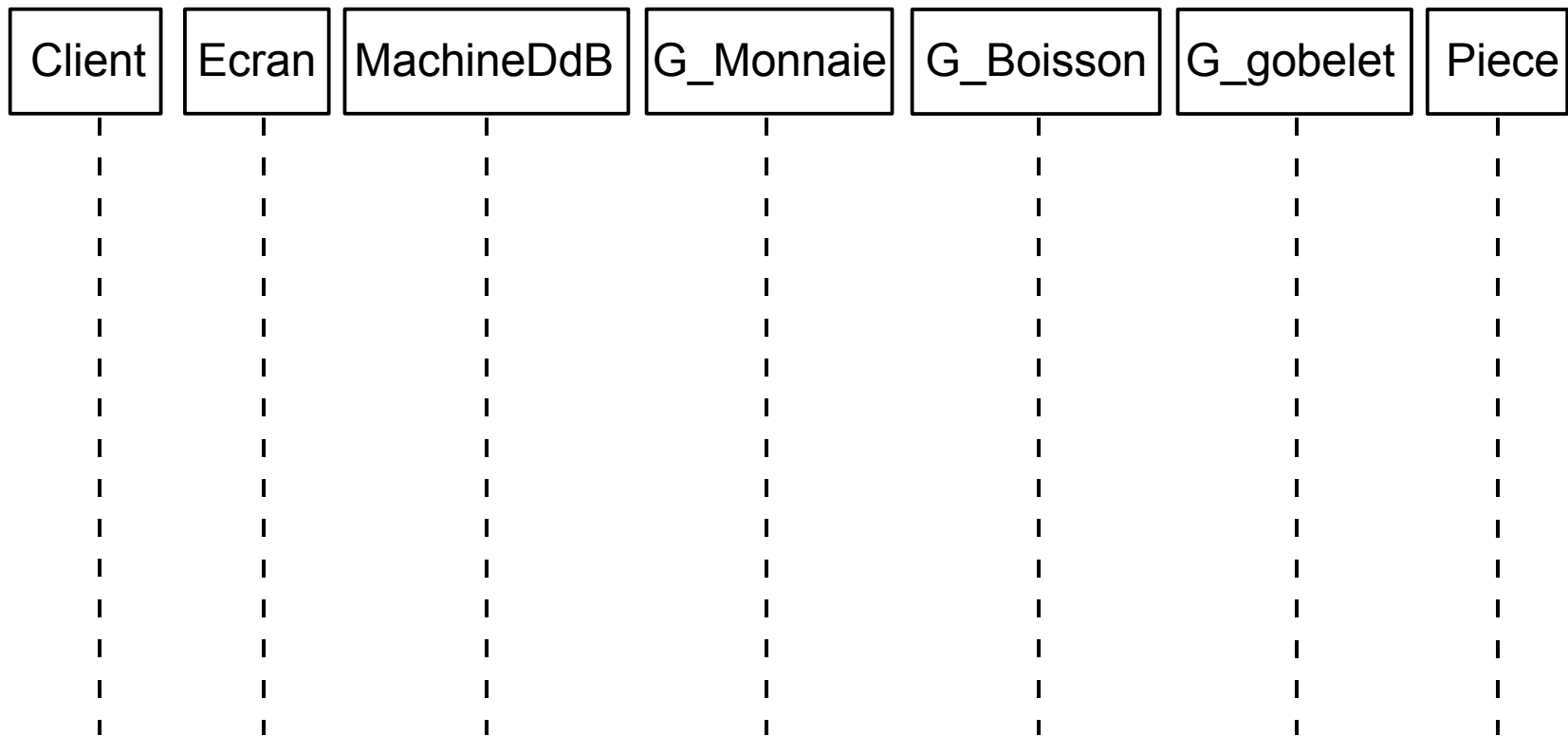
## Méthode n°2





# Diagrammes de séquence

## ■ Exercice : distributeur de boissons



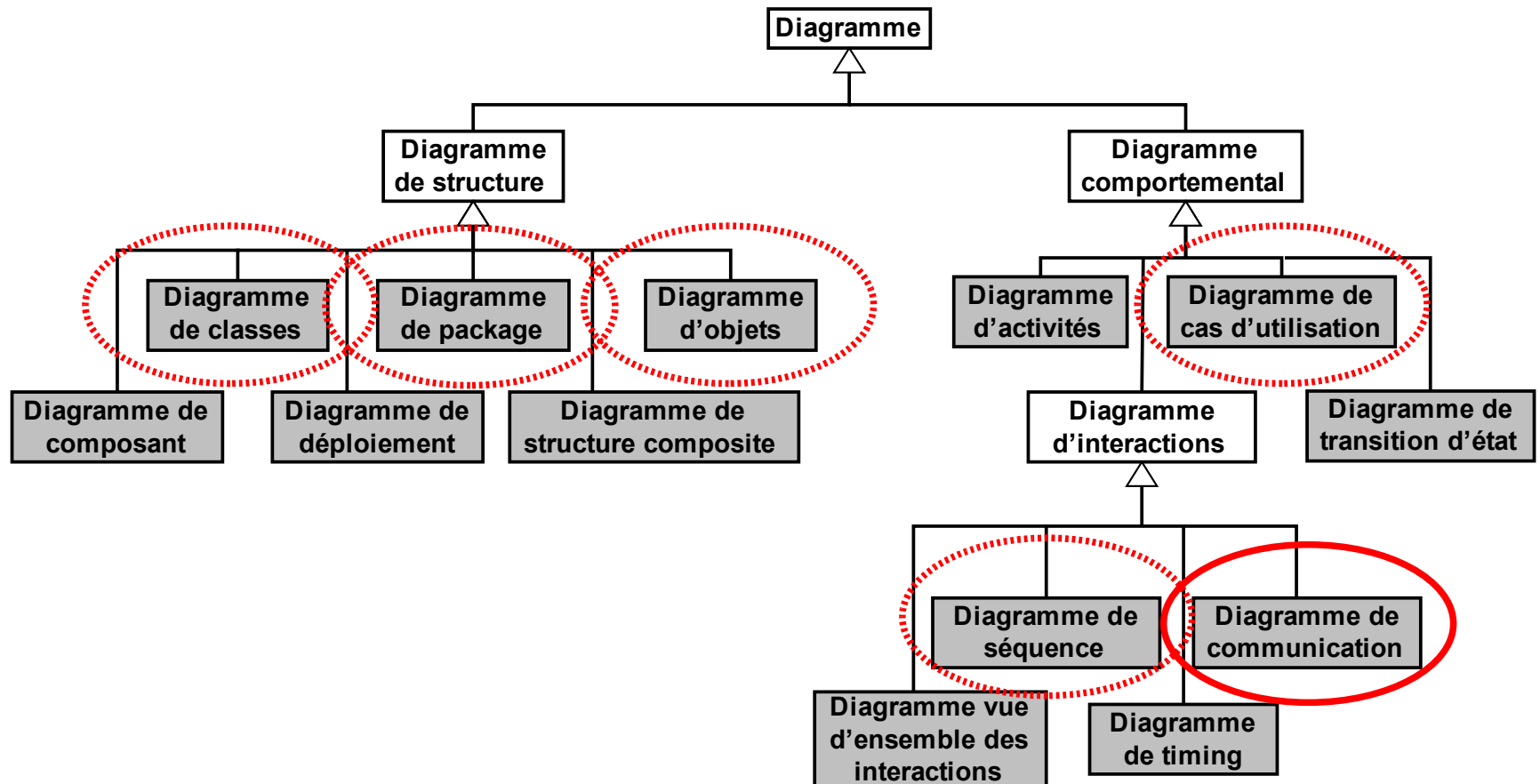


## II.6. Diagrammes d'interaction

### 1. Dia. de communication

# Introduction

## ■ Les diagrammes d'UML



# Diagrammes de communication

- Objectif

- Extension du diagramme d'objet
  - 1 - le contexte = modèle objets
  - 2 - les interactions = les scénarios
- Montrent les interactions entre objets du point de vue statique et dynamique
  - 1 - Objets = instances de classes
  - 2 - Lien = instance des associations définies dans le diagramme de classes

# Diagrammes de communication

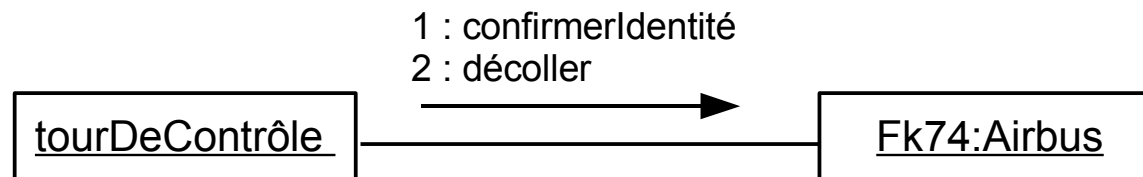
- Concepts

- Objets / acteurs

- Messages séquentiels numérotés

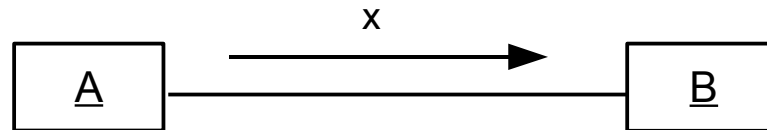
Echangés entre objets et non entre classes

- Exemple :



# Diagrammes de communication

## ■ Exemples



4 : Afficher(x,y) // message simple

3.3.1 : Afficher(x,y) // message imbriqué

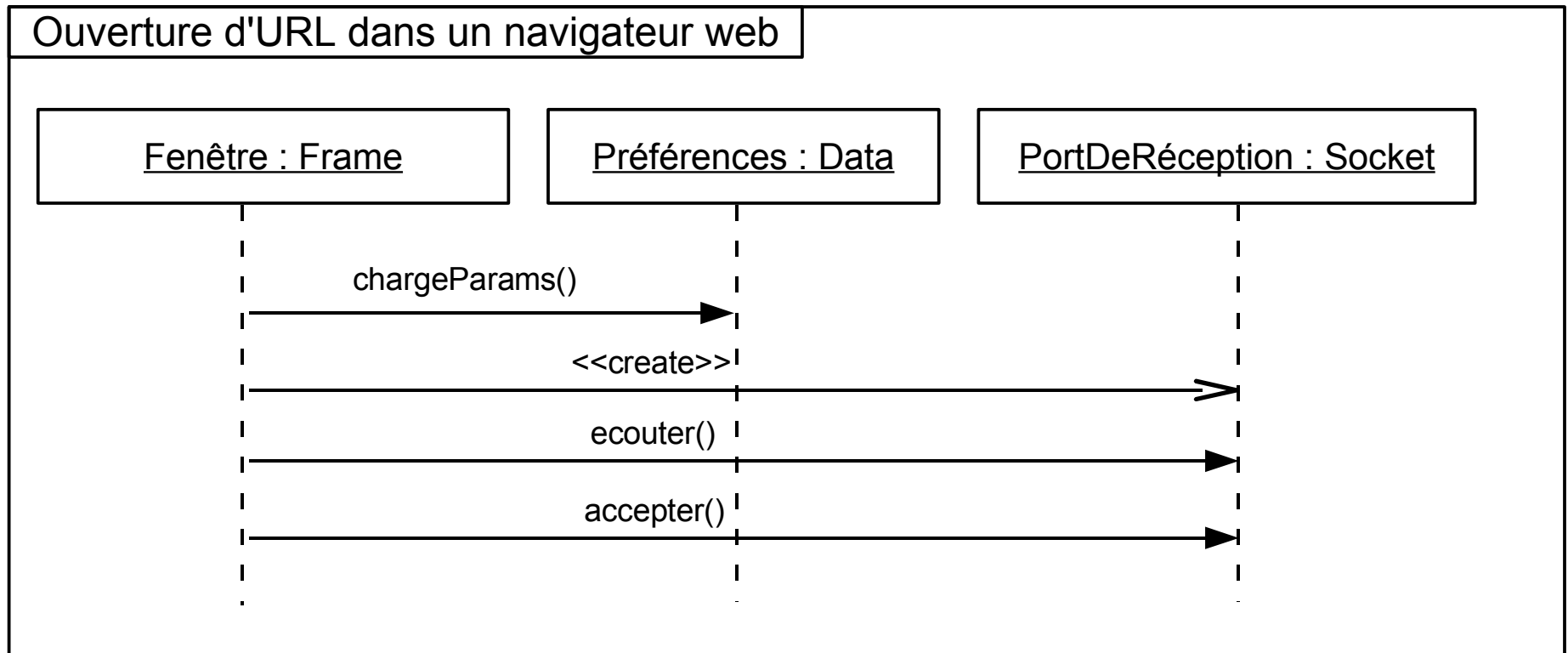
4.2 : âge := Soustraire( Aujourd'hui, DateDeNaissance ) // message imbriqué  
avec valeur retournée

[Age >= 18 ans] 6.2 : Voter() // message conditionnel

# Diagrammes de communication

## ■ Exercice

Transformer ce diagramme de séquence en diagramme de communication



# Diagrammes de communication

- Exemple : distributeur de boissons

