

Exposé 4: Diagrammes de séquences

GL

2^{ème} année

Diagramme de séquence (1)

objectif

En phase d'analyse: Montrer

- Interactions entre **acteurs** et **systeme**
- Décrire les scénarios des cas d'utilisation

En phase de conception: Montrer

- Interactions entre **objets**
- Réfléchir à l'affectation de responsabilités aux objets:
 1. Qui crée les objets ?
 2. Qui permet d'accéder à un objet ?
 3. Quel objet reçoit un message provenant de l'IHM (Interface homme machine) ?

Remarques:

- Avoir un faible couplage et une forte cohésion
- Elaboration en parallèle avec les diagrammes de classes

Diagramme de séquence (1)

objectif: en bref

- **Représentation temporelle** des interactions entre les objets
- **Chronologie** des messages échangés entre les objets et avec les acteurs

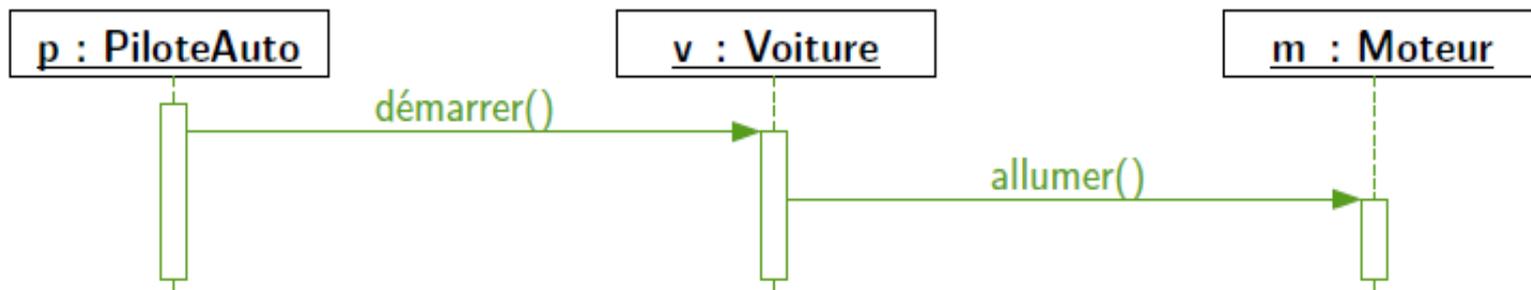


Diagramme de séquence (1)

éléments et principe

Éléments du diagramme de séquence

- **Acteurs**
- **Objets** (instances)
- **Messages** (cas d'utilisation, appels d'opération)

Principes de base : Représentation graphique de la chronologie des **échanges de messages** avec le système ou au sein du système

- « **Vie** » de chaque entité représentée verticalement
- **Échanges** de messages représentés horizontalement

Diagramme de séquence (4) ligne de vie

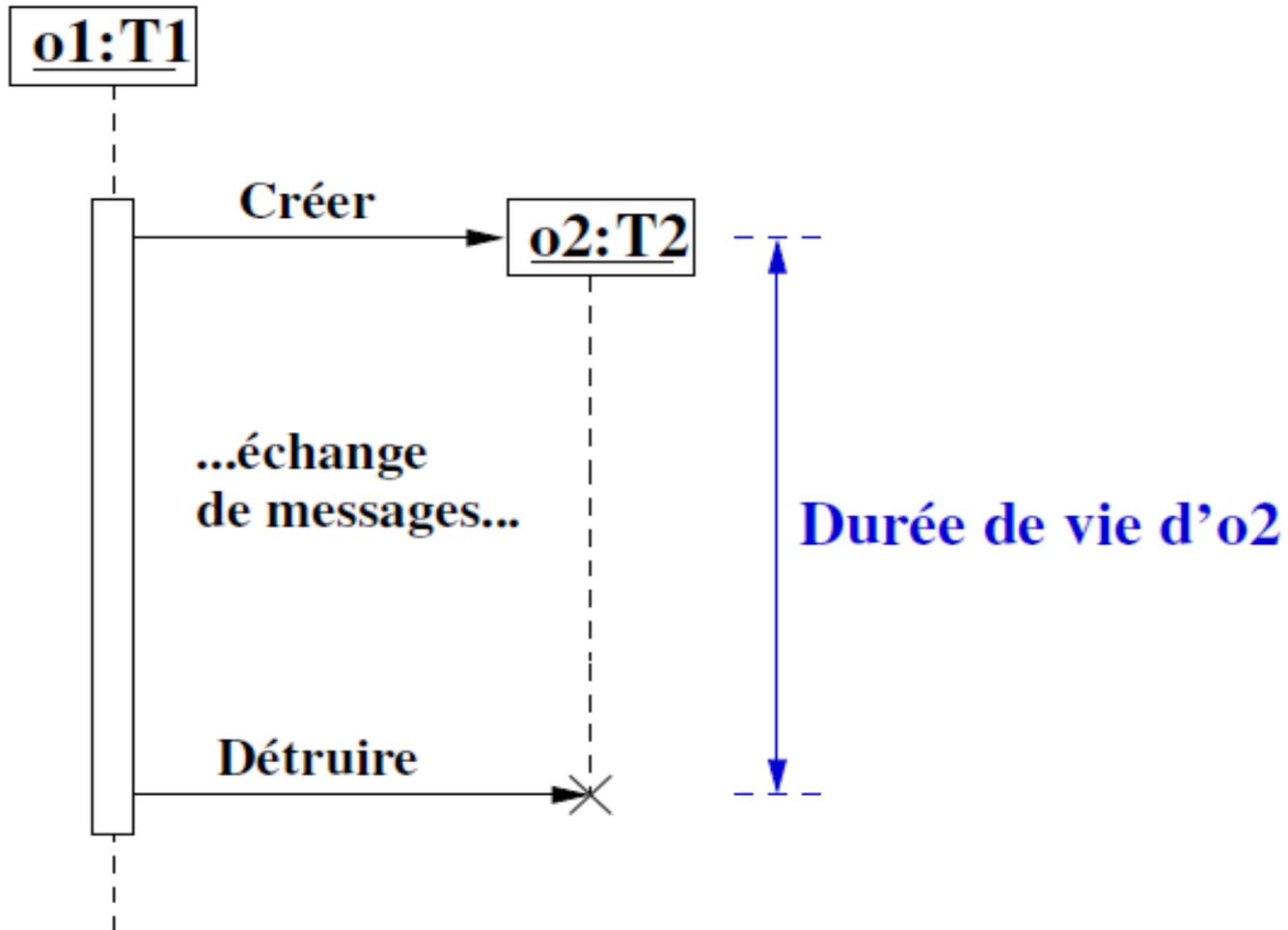


Diagramme de séquence (2) communication

- temps : axe vertical
- Objets : axe horizontal

Exemple :

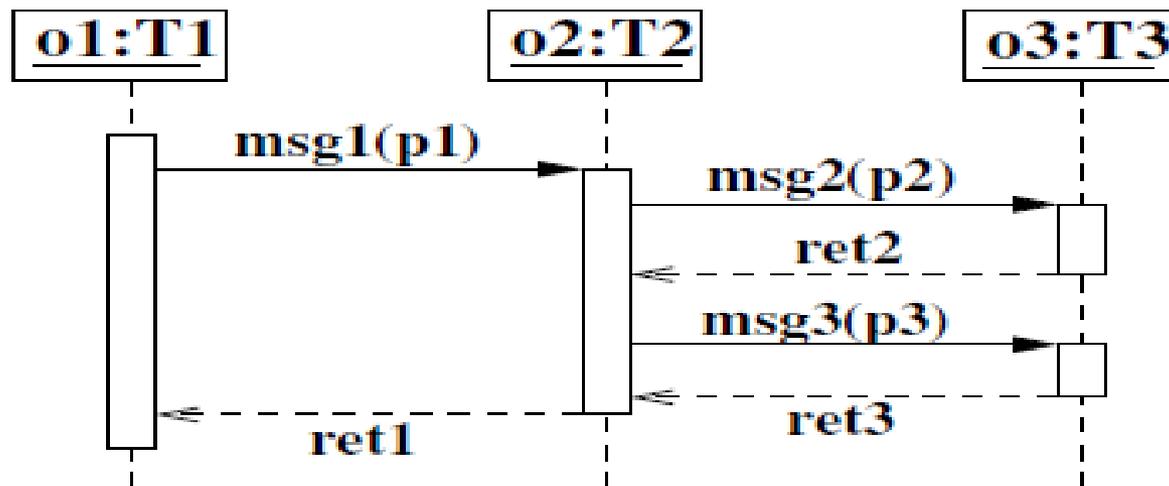


Diagramme de séquence (3) communication

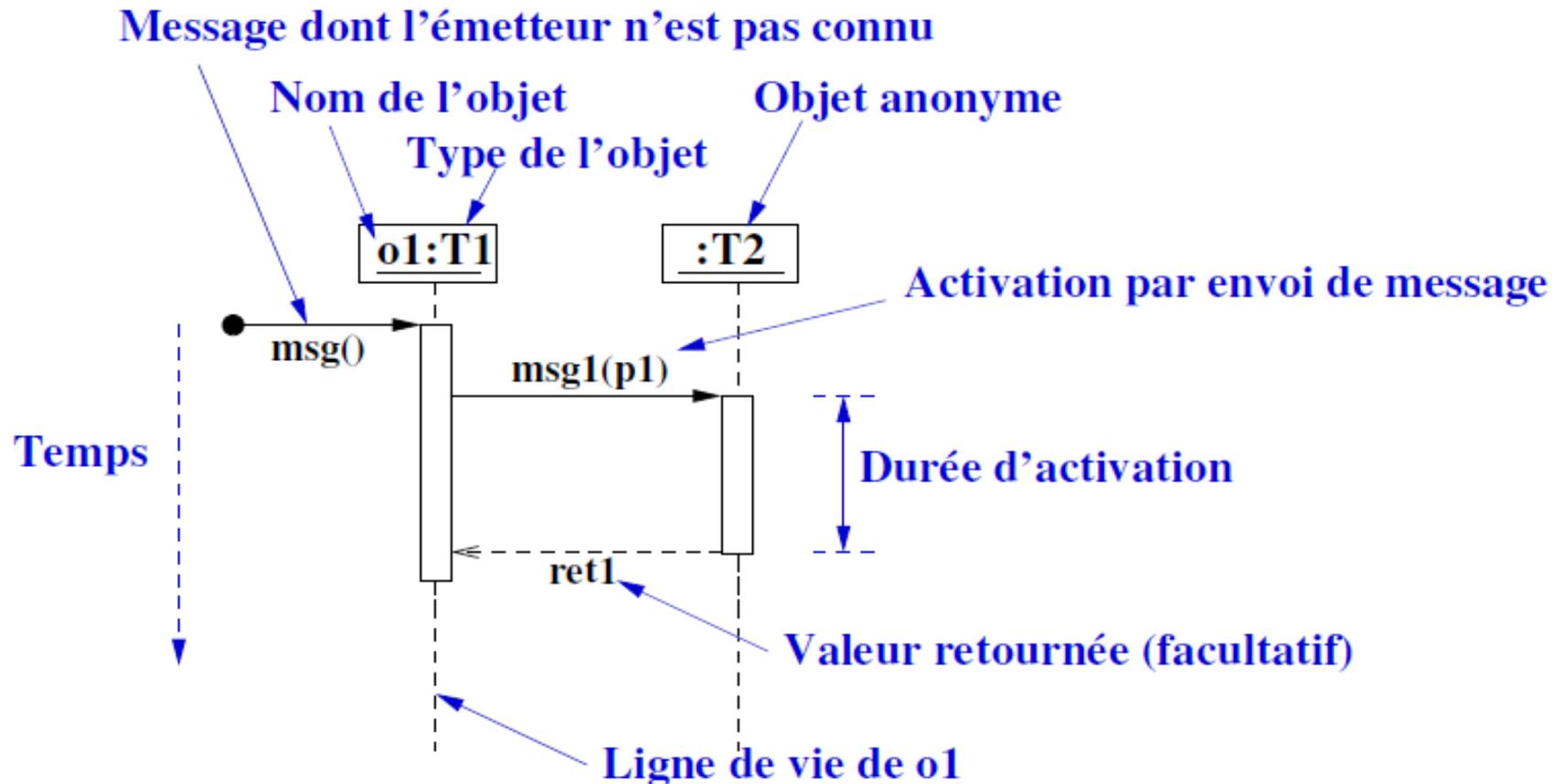


Diagramme de séquence (1) avec acteur

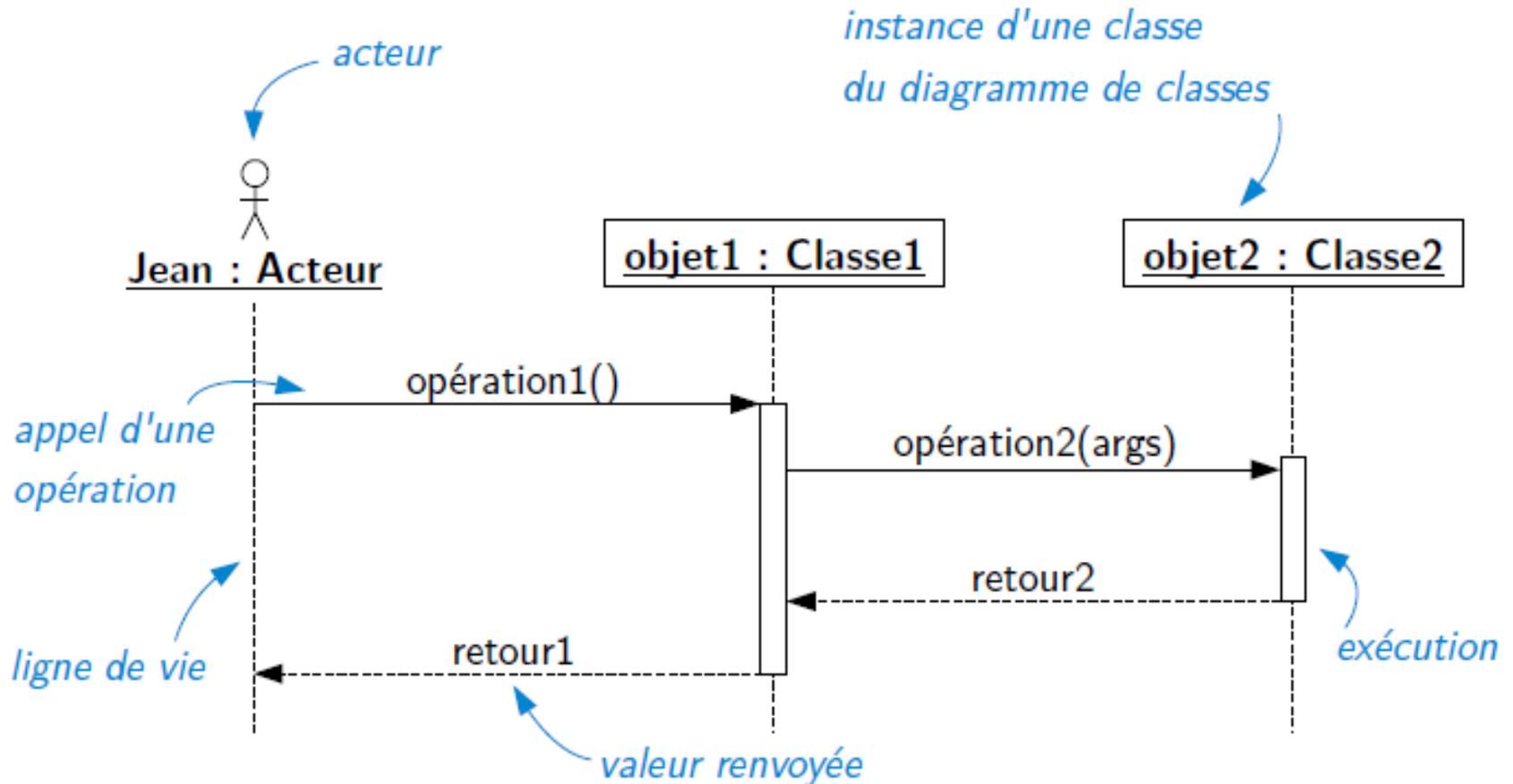


Diagramme de séquence (1) avec acteur: exemple

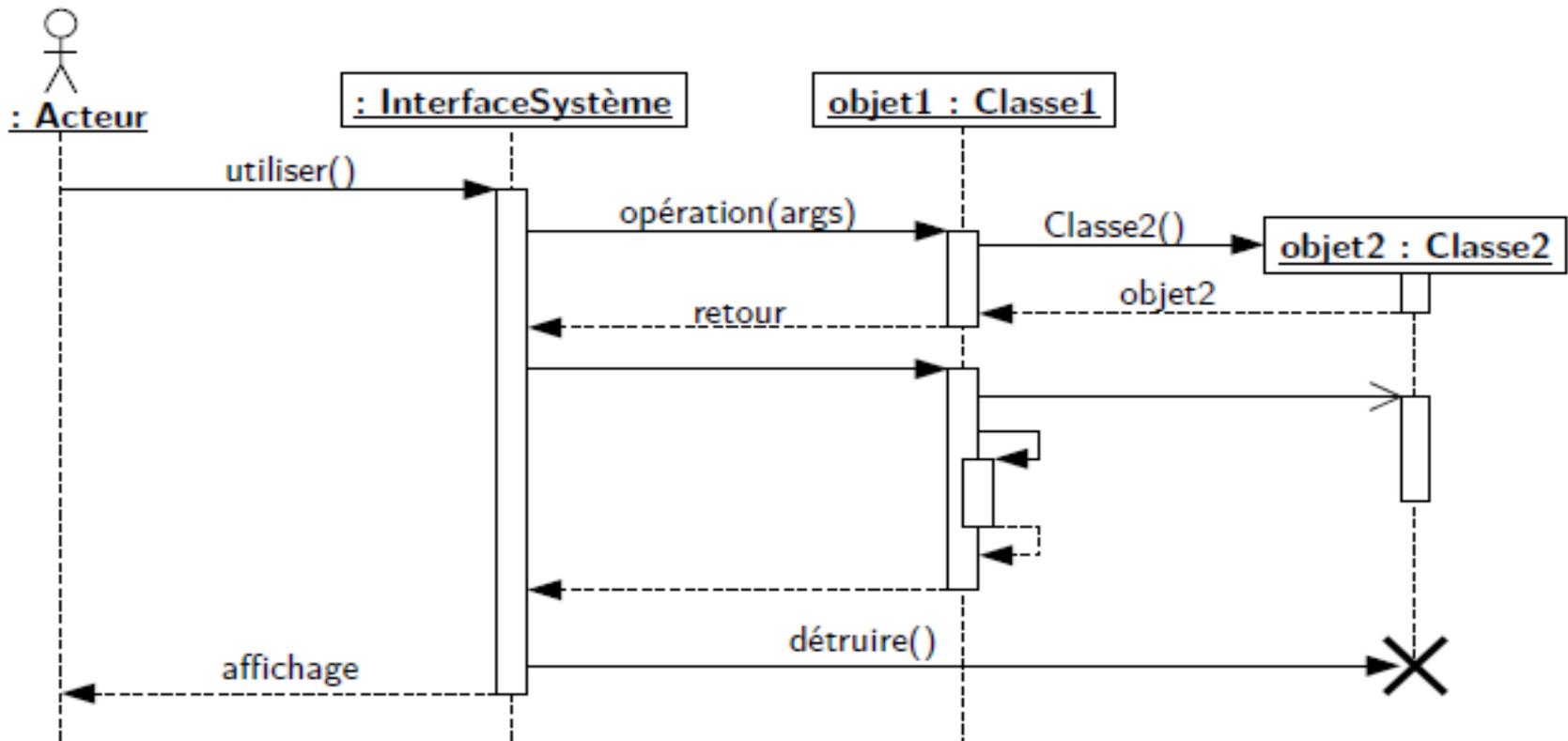
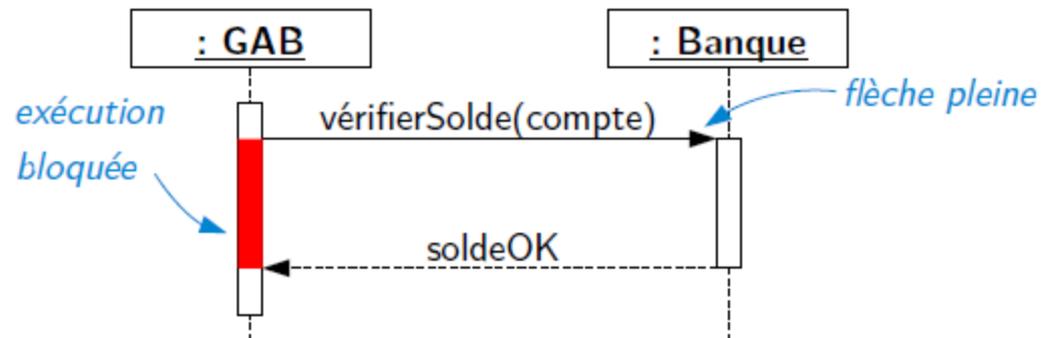


Diagramme de séquence (6)

messages: synchrone vs asynchrones

Message synchrone : Émetteur bloqué en attente du retour



Message asynchrone : Émetteur non bloqué, continue son exécution

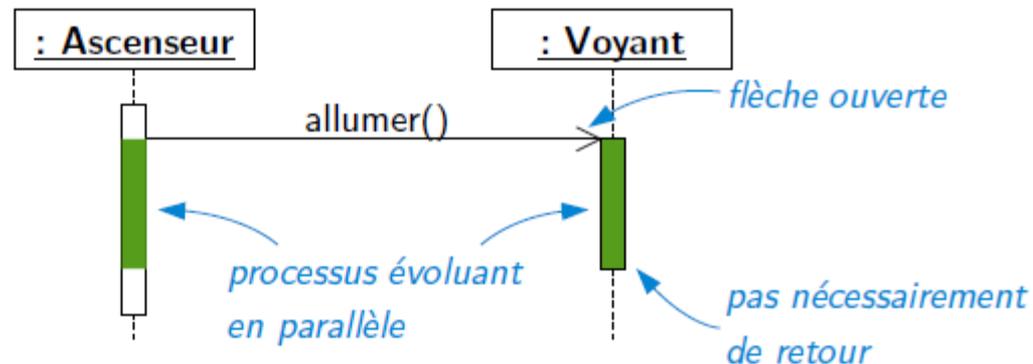


Diagramme de séquence (6) messages réflexifs

Messages réflexifs

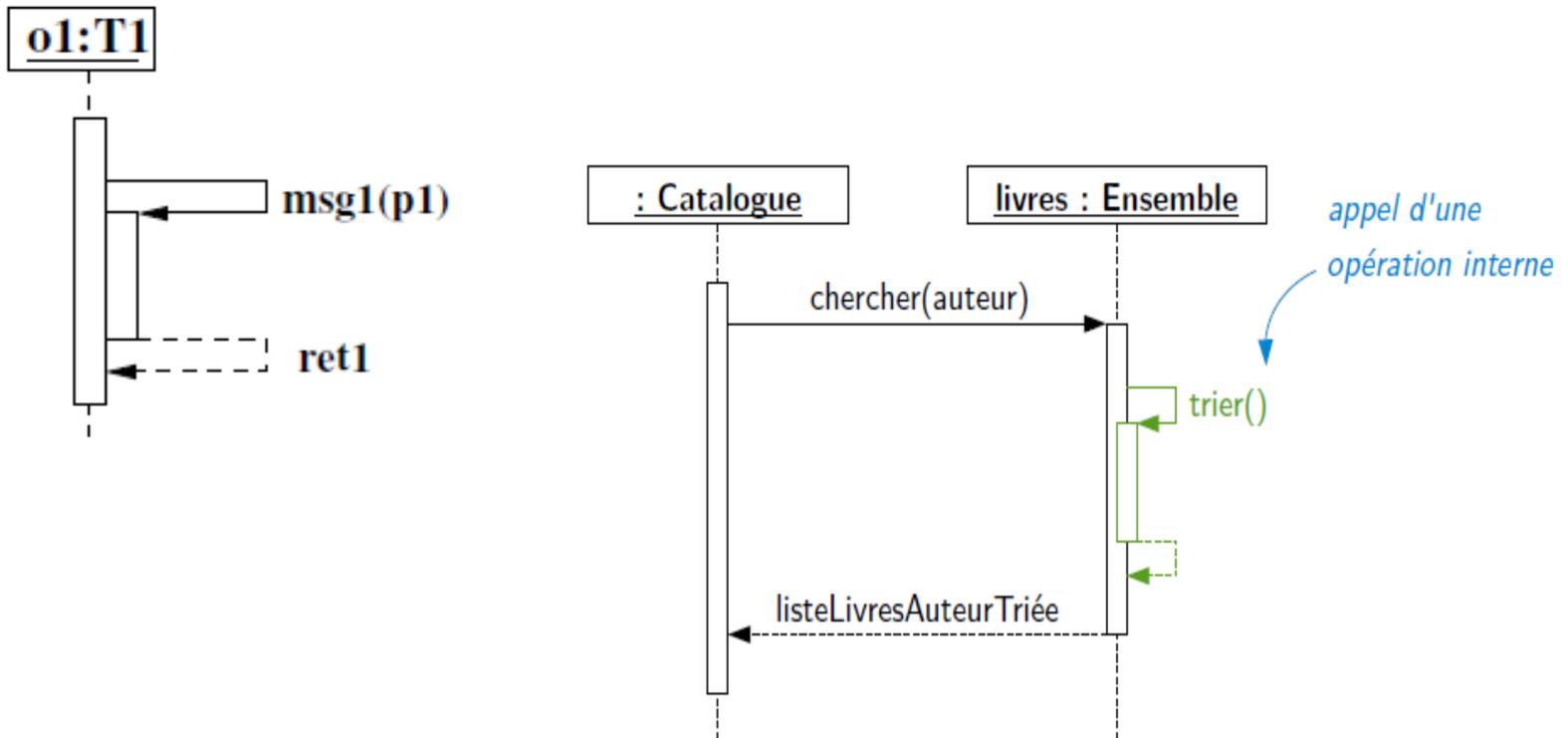


Diagramme de séquence (6)

messages: create, destroy

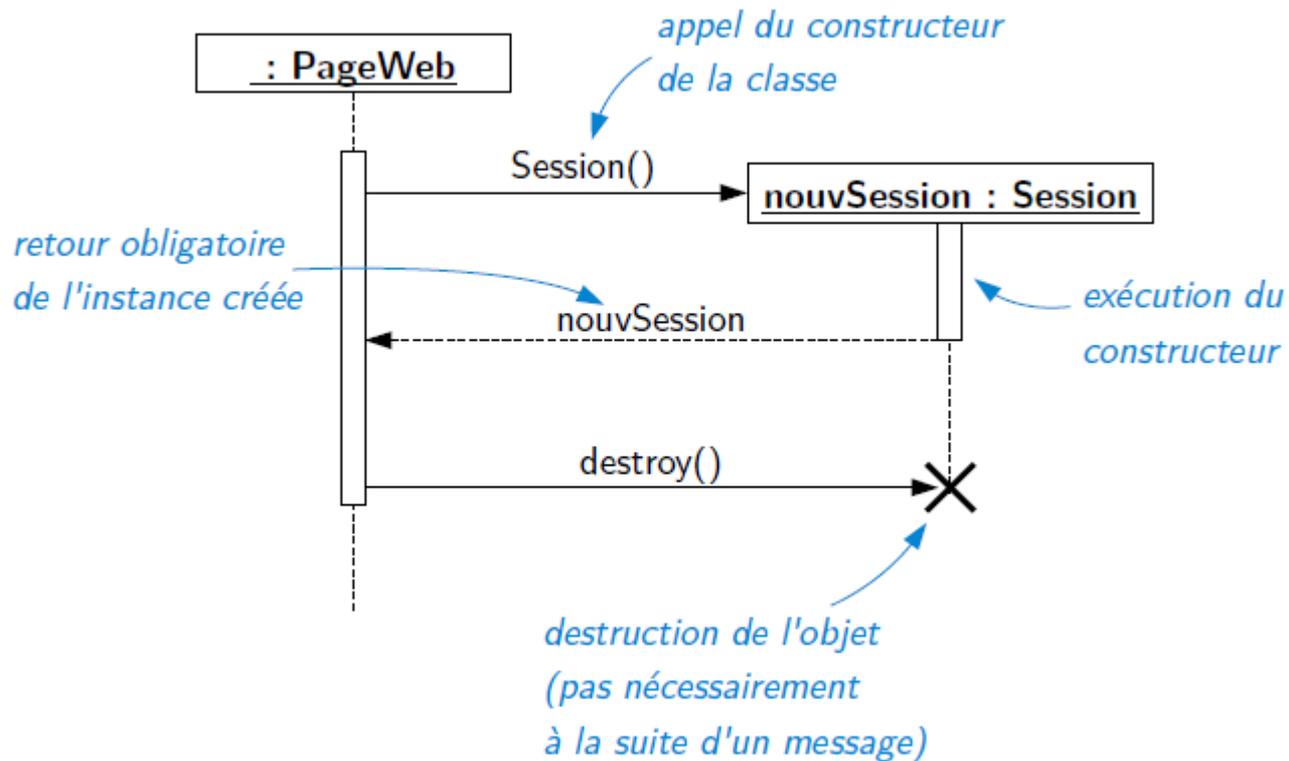


Diagramme de séquence (7)

exprimer des contraintes temporelles

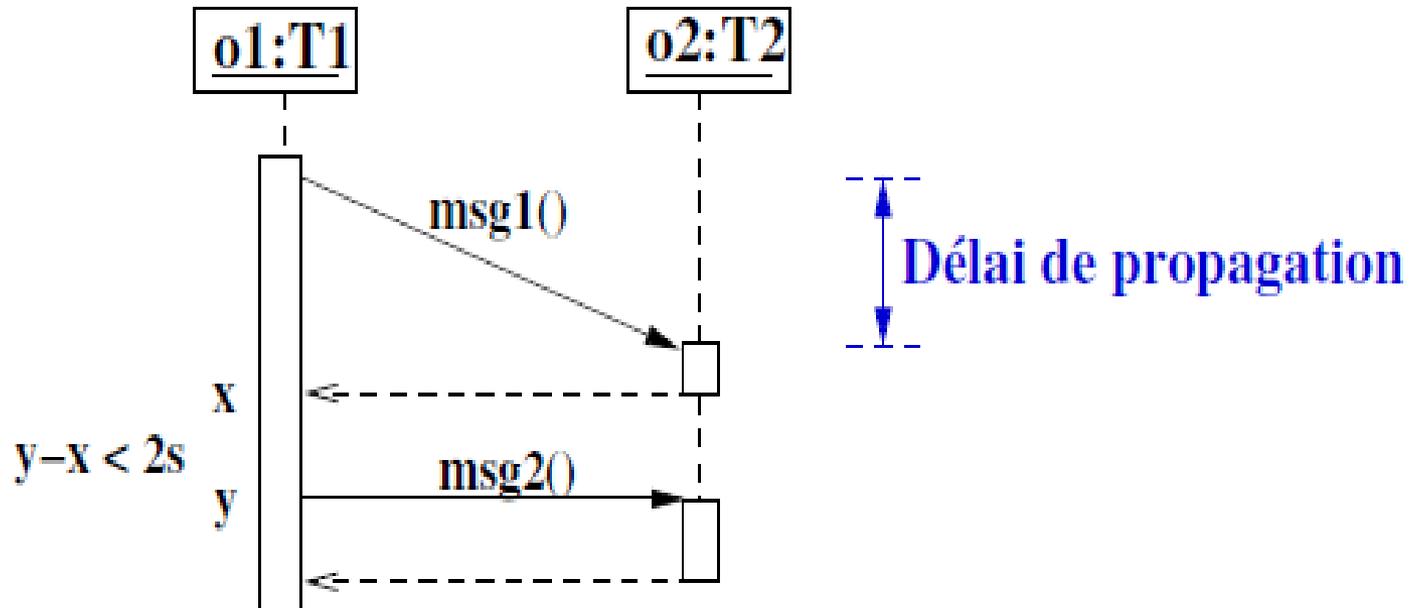


Diagramme de séquence (8) exprimer des choix

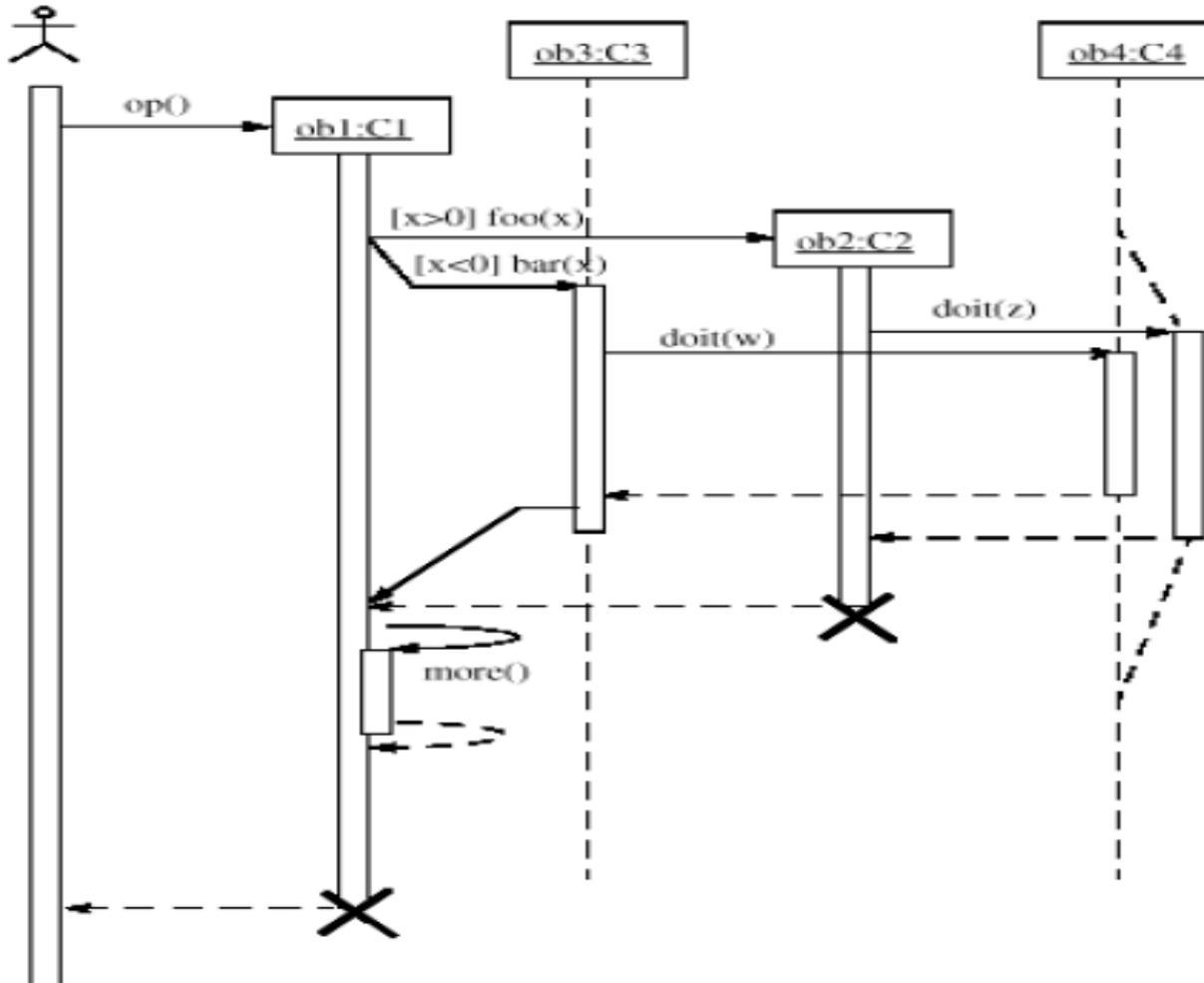


Diagramme de séquence (6)

Structuration: usage des cadres

- **alt** : fragments multiple alternatifs (si alors sinon)
- **loop** : le fragment s'exécute plusieurs fois
- **ref** : référence à une interaction dans un autre diagramme
- **par** : fragment parallèle (traitements concurrents)
- **opt** : fragment optionnel
- **region** : région critique (un seul thread à la fois)

Diagramme de séquence (6)

structure: Alternatives

Principe : Condition à l'envoi d'un message

Notation :

- Deux diagrammes
- Bloc d'alternative **alt**

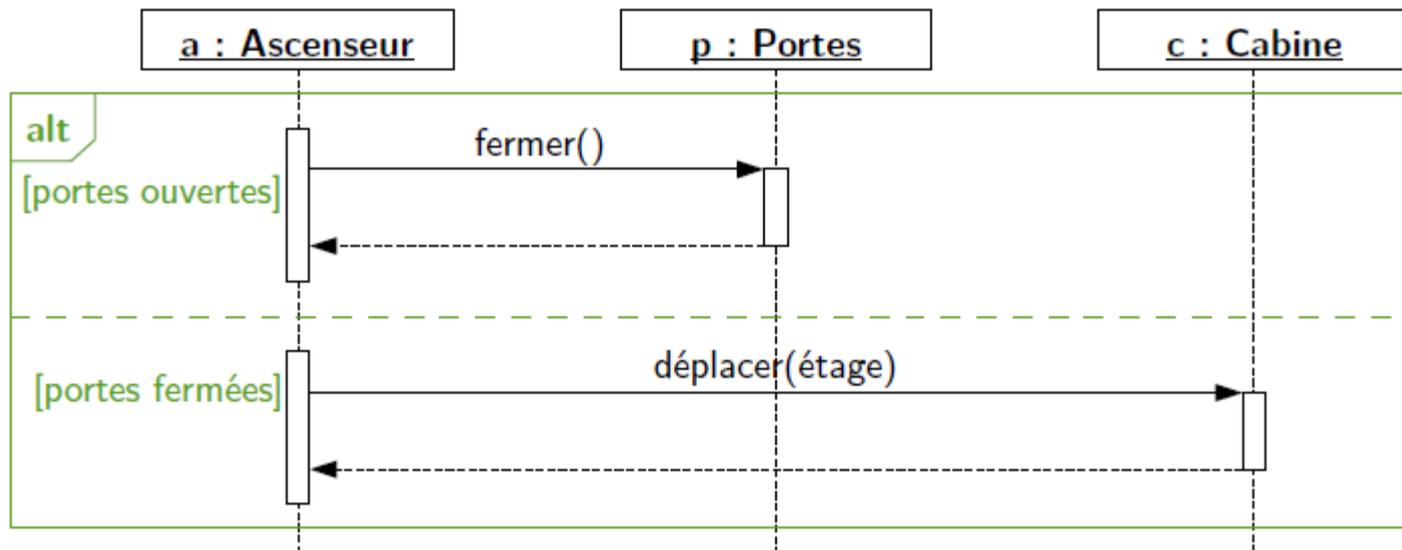


Diagramme de séquence (6)

structure: Alternatives

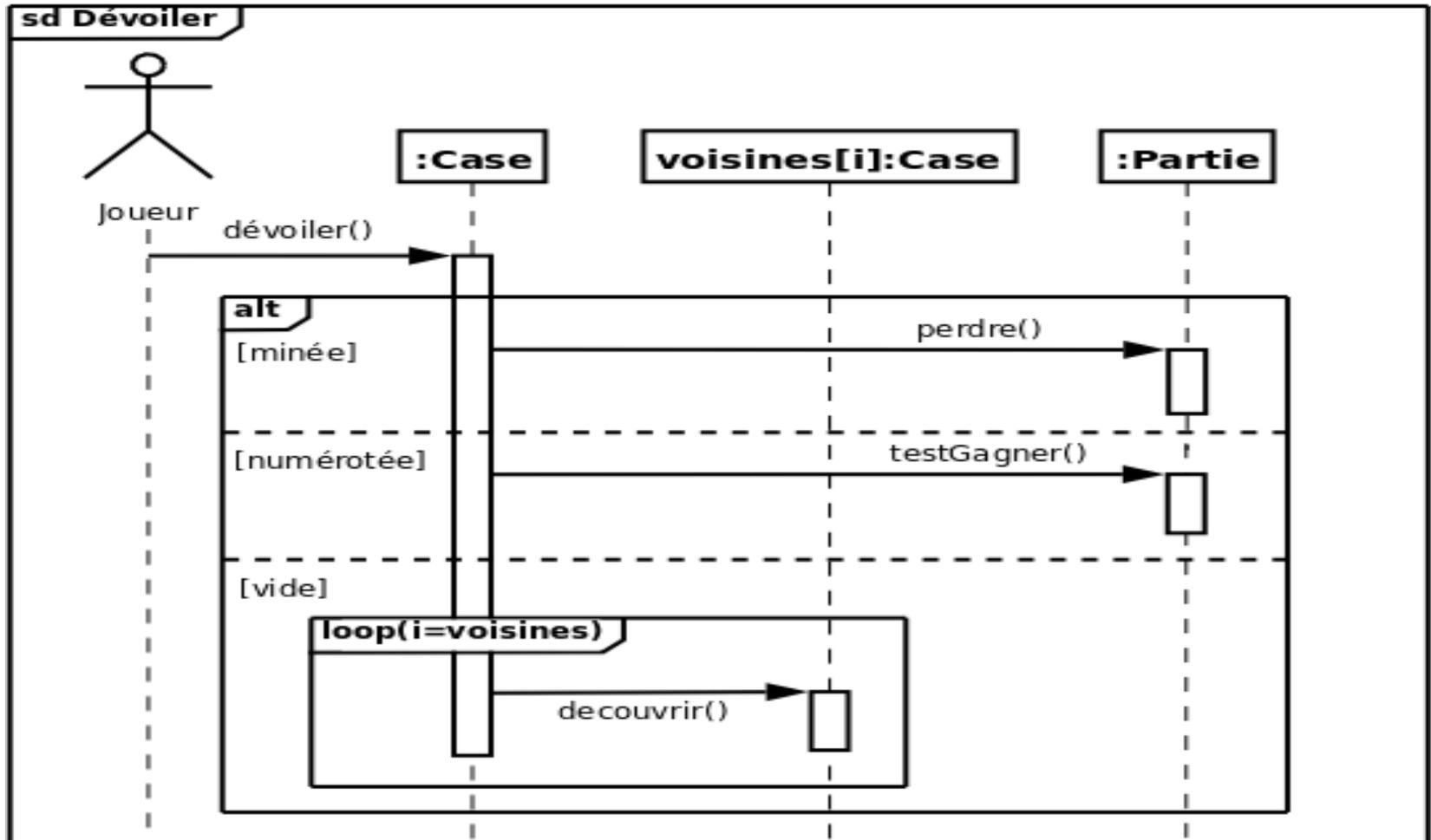


Diagramme de séquence (6)

structure: **boucle**

- Pour répéter un enchaînement

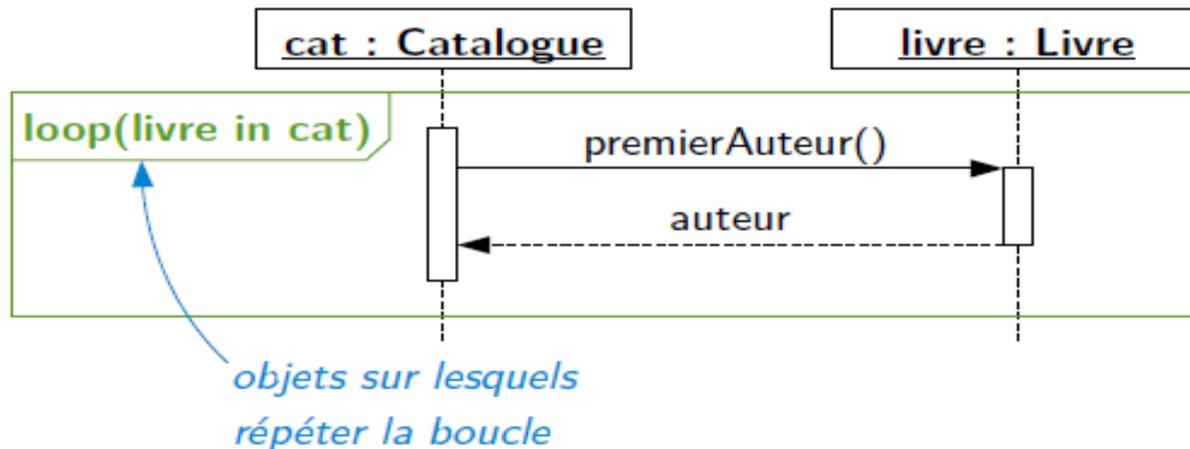


Diagramme de séquence (8)

Structure: **références**

- On peut faire référence à un autre diagrammes

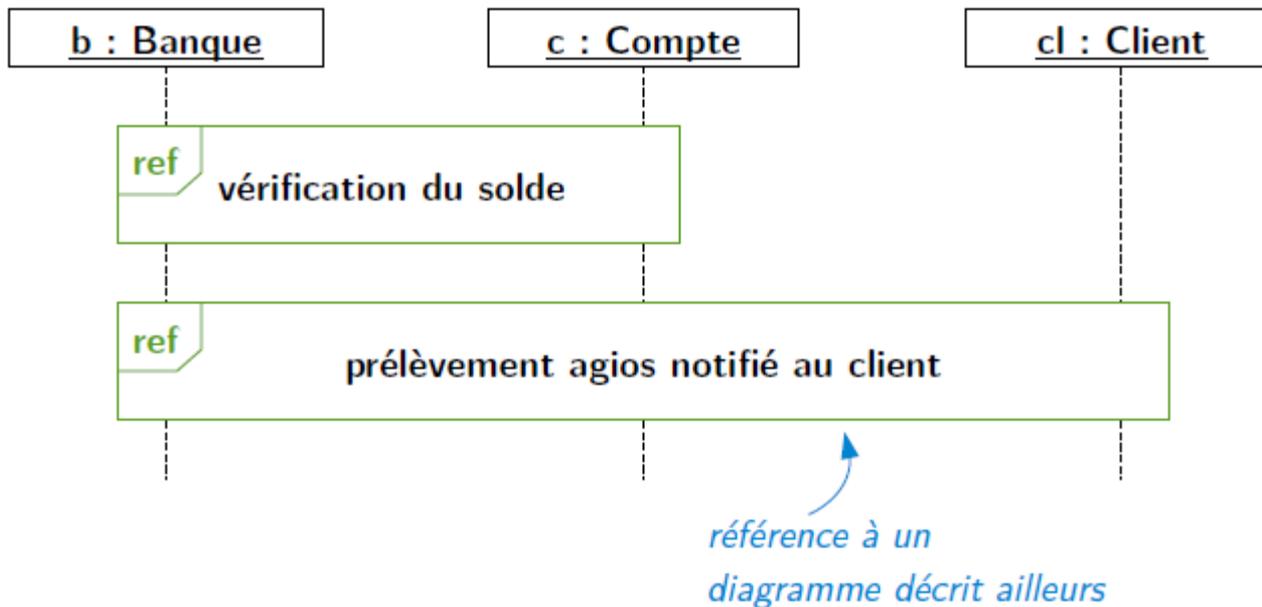


Diagramme de séquence (8)

structure: références

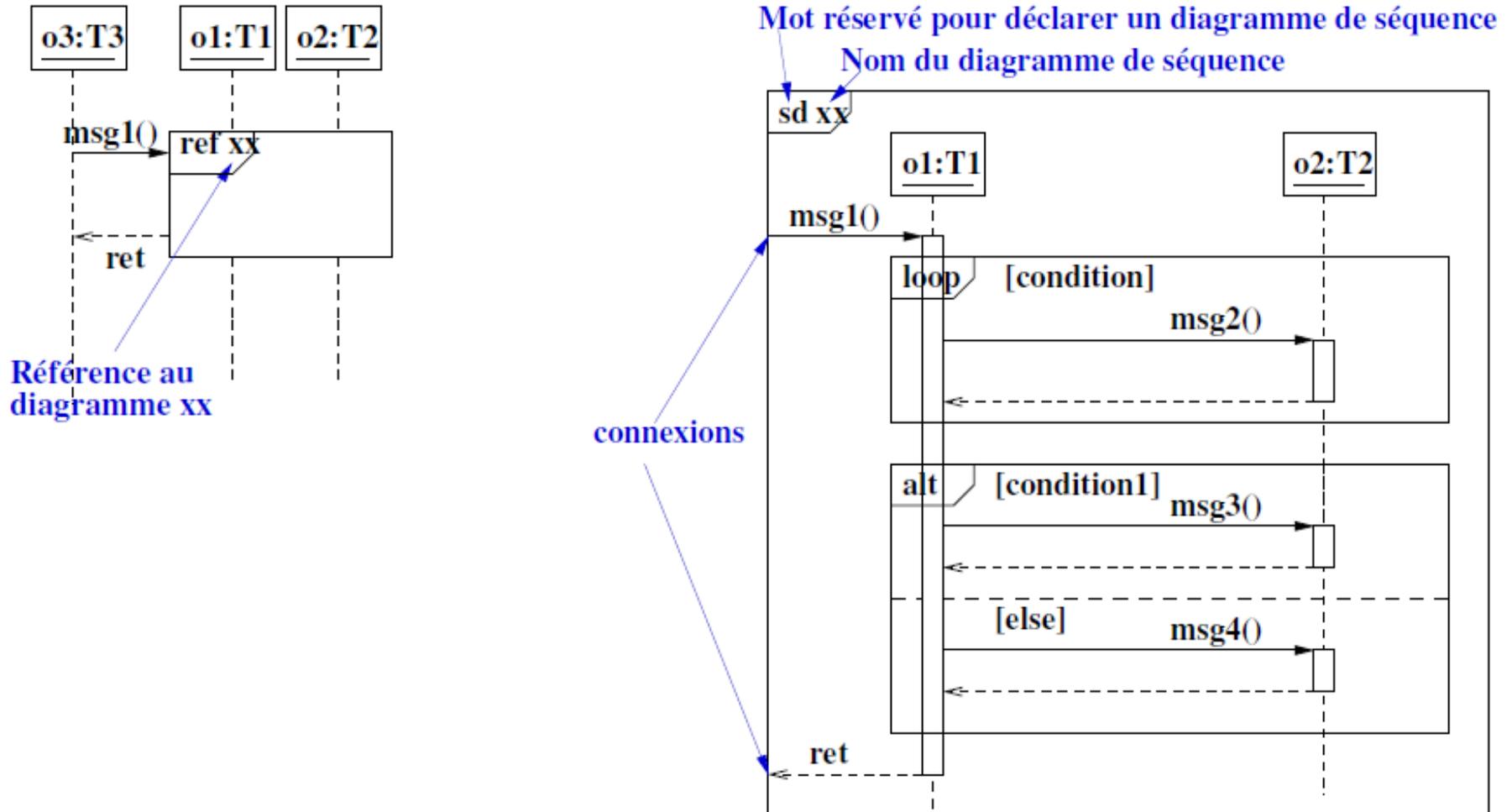


Diagramme de séquence (8) structure: **parallèle**

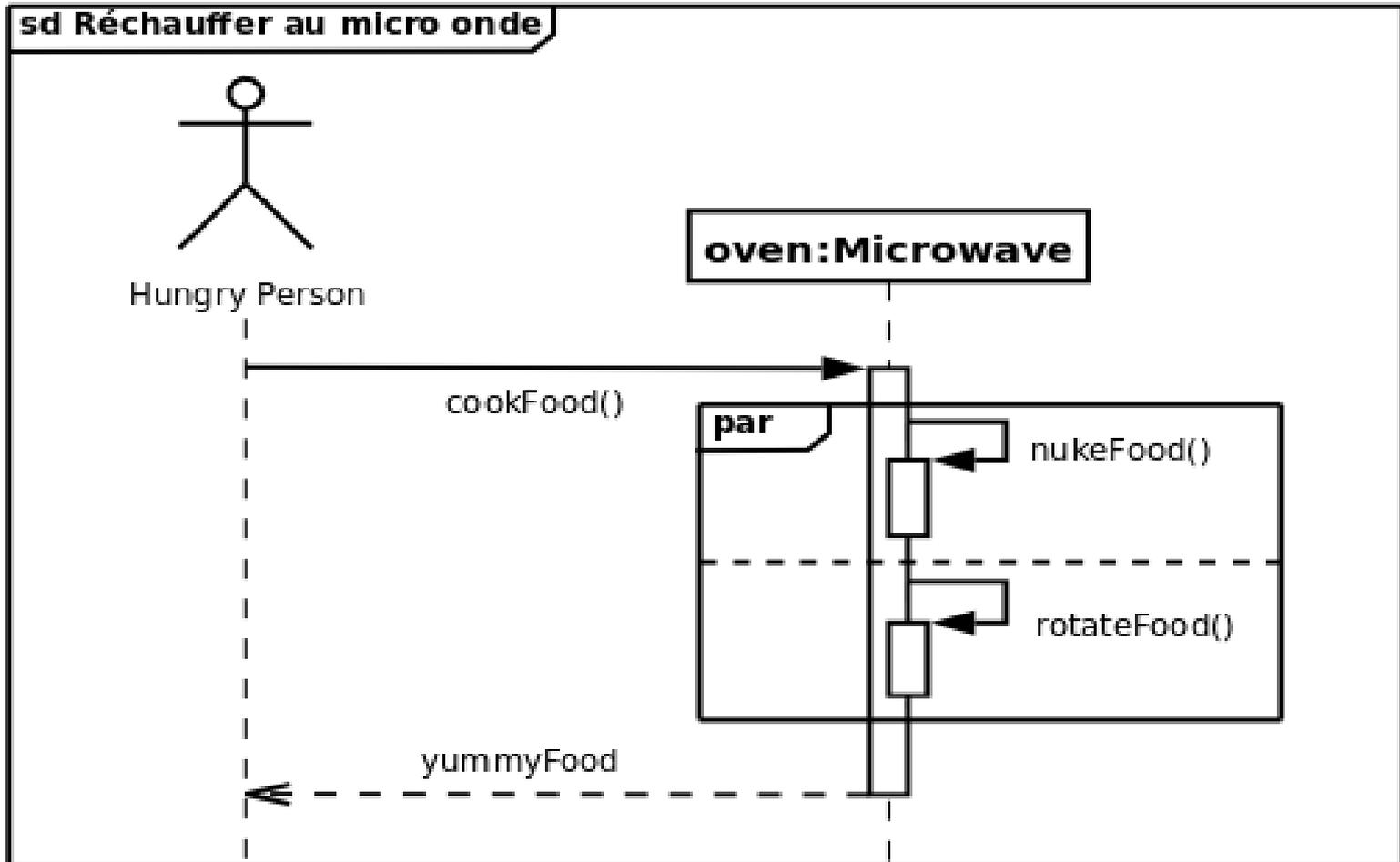


Diagramme de séquence (9) relation entre DS et DC

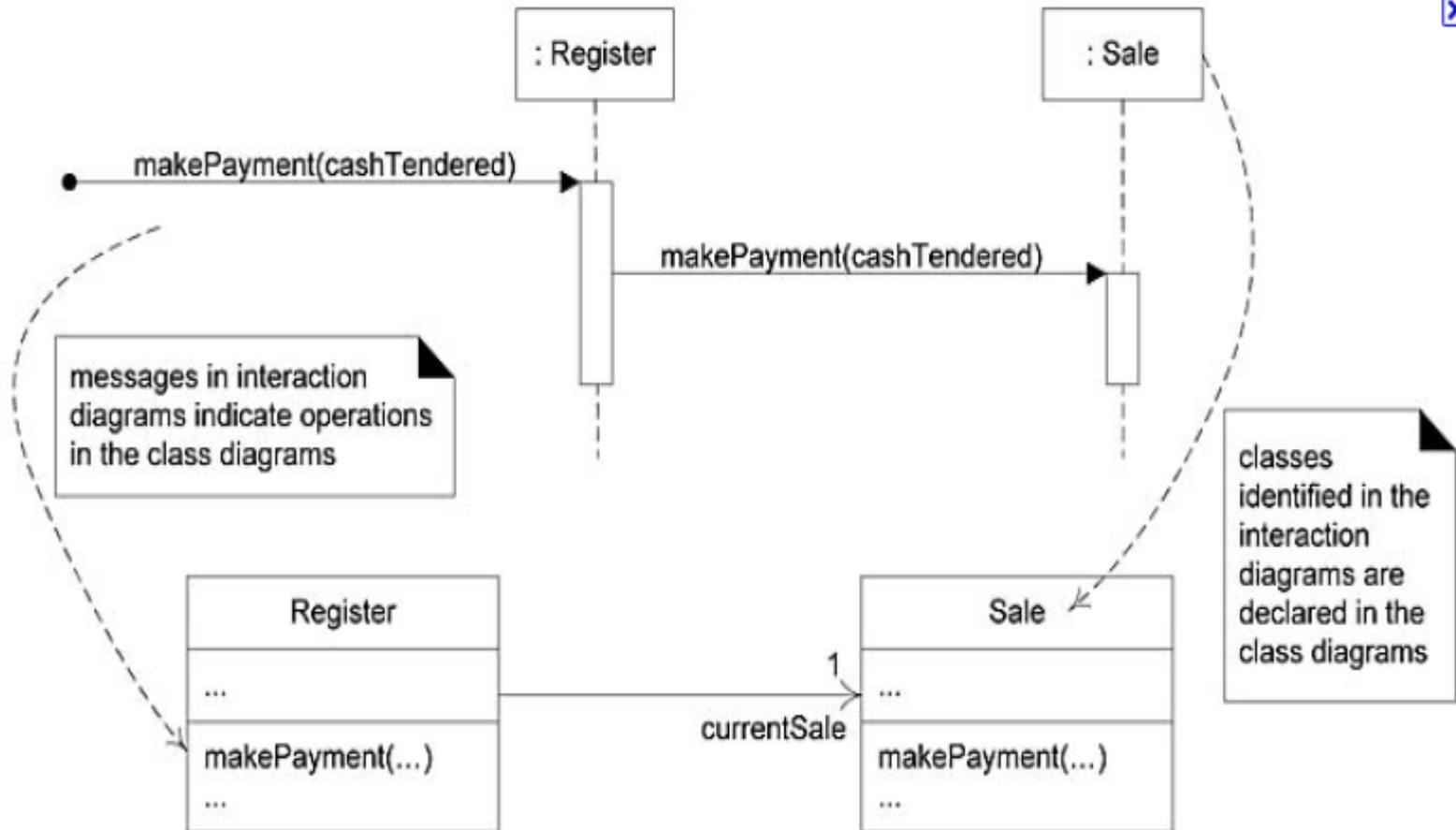
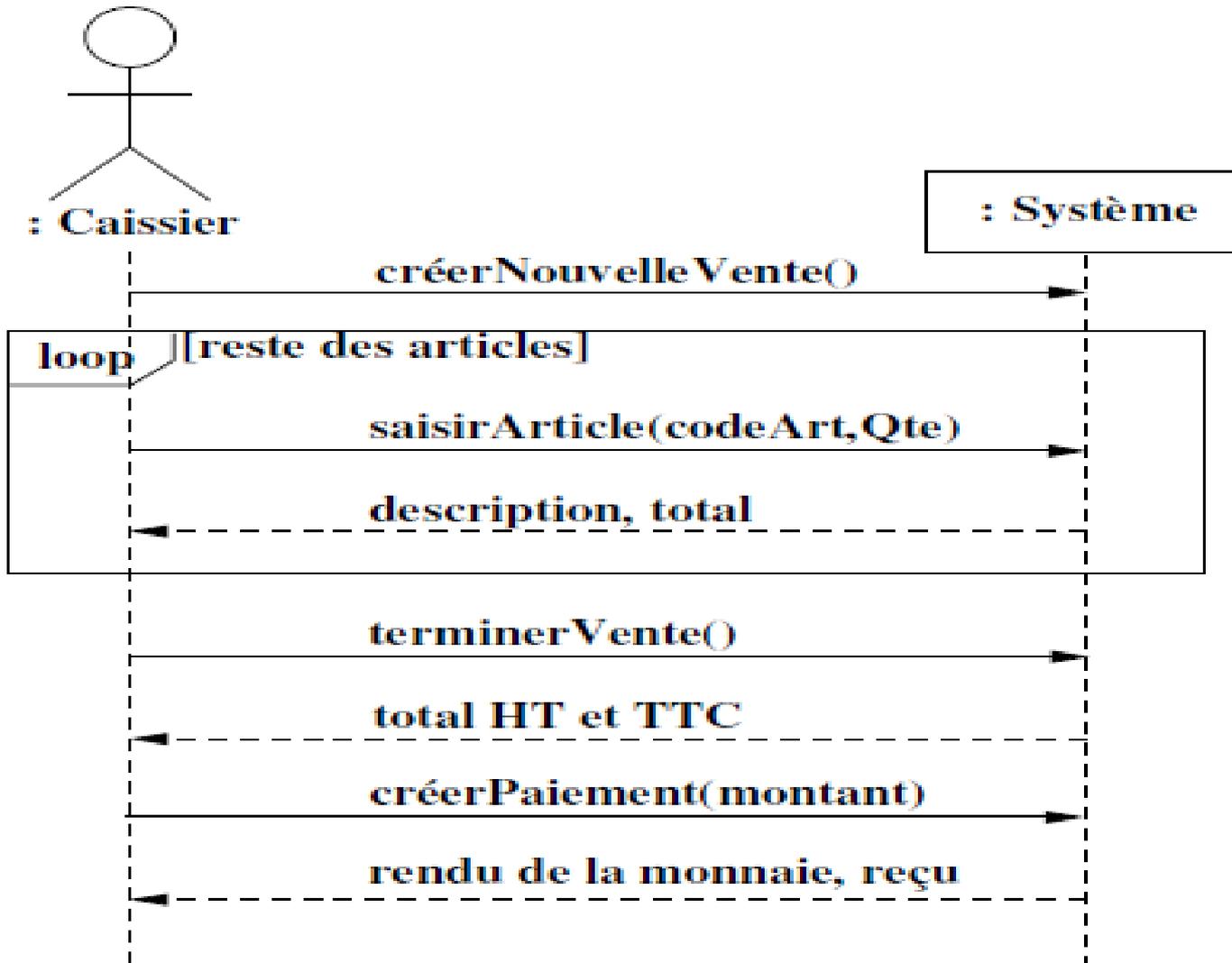


Diagramme de séquence (10) relation entre DS et DCU



Exemple

