

Chapitre IV: Diagramme des séquences

Nour H. BEN SLIMEN ATTAWI

UML: DIAGRAMME DES SEQUENCES

Définition

- ▶ Pour chaque cas d'utilisation, un ou plusieurs **scénario** peuvent être détaillés par un diagramme de séquences.
- ▶ Les diagrammes de séquences permettent de décrire **COMMENT** les éléments du système interagissent entre eux et avec les acteurs :
 - ▶ Les objets au cœur d'un système interagissent en s'échangeant des messages.
 - ▶ Les acteurs interagissent avec le système au moyen d'IHM (Interfaces Homme-Machine).
- ▶ Diagramme de séquences : exprime la séquence des interactions entre objets du système selon un point de vue temporel, pour réaliser le cas d'utilisation.

UML: DIAGRAMME DES SEQUENCES

Formalisme

Le diagramme de séquence comprend les éléments suivants:

- ▶ des objets,
- ▶ une ligne de vie (axe du temps),
- ▶ des périodes d'activité,
- ▶ les messages

UML: DIAGRAMME DES SEQUENCES

Formalisme

Les objets

Les objets sont de quatre types:

1. **Acteurs,**
2. **Interface utilisateurs:** utilisées pour modéliser l'interaction entre le système et les acteurs. Elles concernent des abstractions d'écrans de saisie, et d'affichage, des états, etc. qui peuvent être programmées ou générés par des macro commandes.

UML: DIAGRAMME DES SEQUENCES

Formalisme

3. **Contrôle** : elles concernent la coordination, le séquençement des autres classes (se trouvent entre les classes entité et interface utilisateur pour éviter un fort couplage entre elles). Exemple: À une facture correspond un règlement. La classe « *contrôle règlement* » permet de contrôler si le montant de la facture et le montant qui a été payé sont égaux.
4. **Entité** : représente les informations persistantes (implique souvent l'implication d'un SGBD = tables d'une BD).

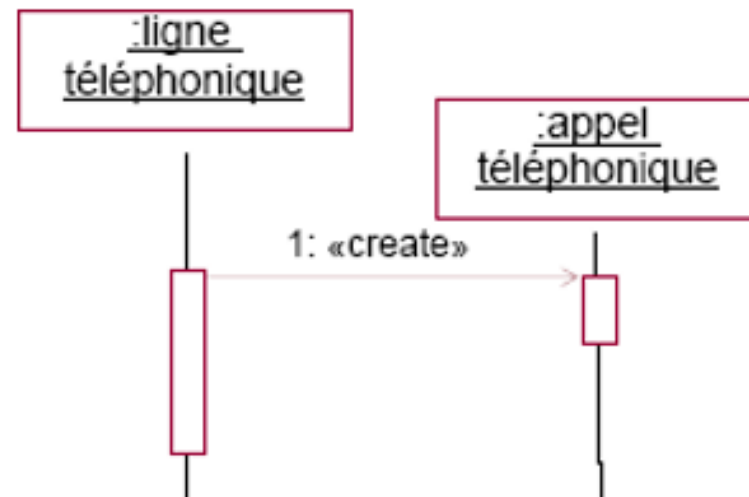
UML: DIAGRAMME DES SEQUENCES

Formalisme

La ligne de vie

Représente l'existence d'un objet sur une période de temps. Le temps se déroule de haut en bas. Les objets sont alignés en haut du diagramme. Pour les objets créés, leur ligne de vie commence au moment de la création, qui se fait à l'aide d'un message qui peut être stéréotypé «create»

Exemple:



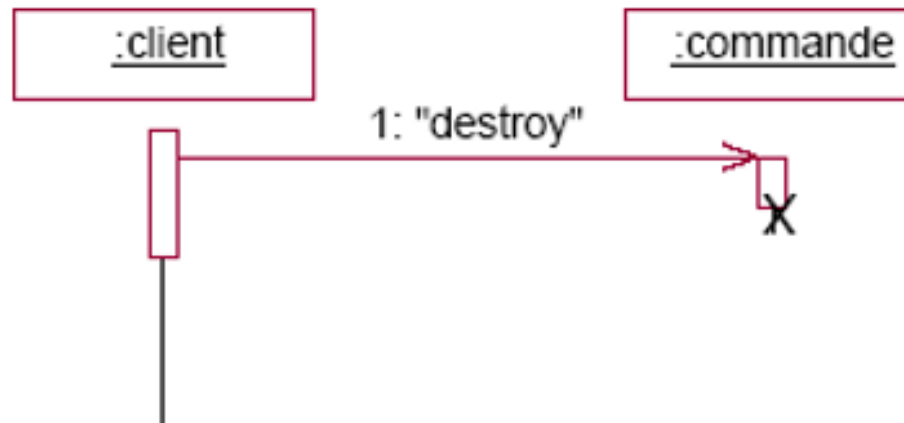
UML: DIAGRAMME DES SEQUENCES

Formalisme

La ligne de vie

Des objets peuvent être détruits pendant l'interaction, leur ligne de vie se termine à la réception du message dont le stéréotype est «destroy» et marqué d'un grand X qui symbolise la fin de leur ligne de vie.

Exemple:

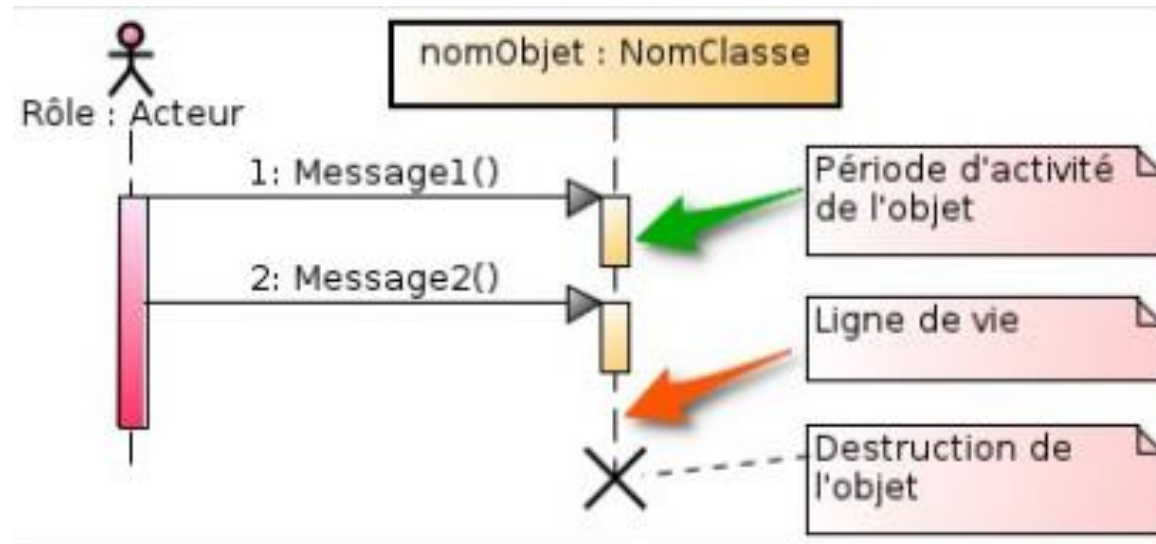


UML: DIAGRAMME DES SEQUENCES

Formalisme

La période d'activité

Lorsqu'un message est reçu, une activité est déclenchée, celle ci s'appelle activation, période d'activité. C'est une bande rectangulaire qui montre la durée pendant laquelle un objet remplit une action.



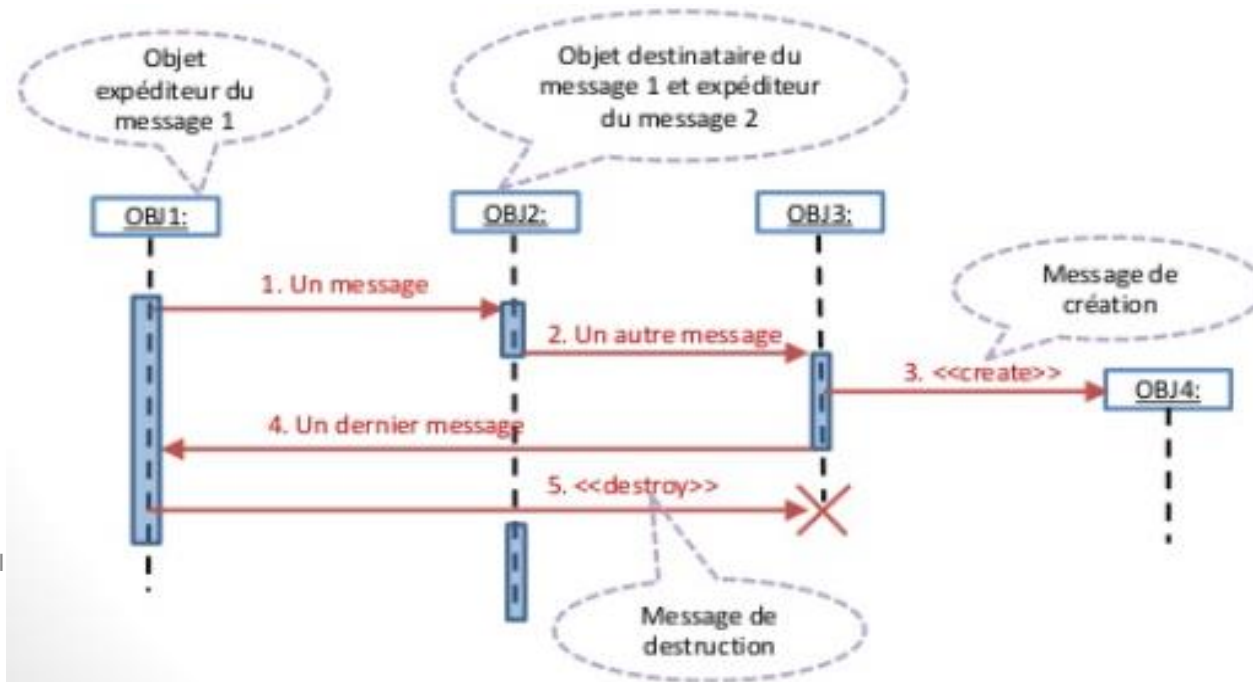
UML: DIAGRAMME DES SEQUENCES

Formalisme

Les messages

- Les messages sont représentés sous forme de flèches.
- Ils sont étiquetés par le nom de l'opération ou du signal invoqué.
- L'ordre d'envoi d'un message est déterminé par sa position sur la ligne de vie; le temps s'écoule de « haut en bas »

Exemple:



UML: DIAGRAMME DES SEQUENCES


Formalisme

Les messages

La forme la plus complète d'un message est:

numéro : [condition de garde] message (paramètres)

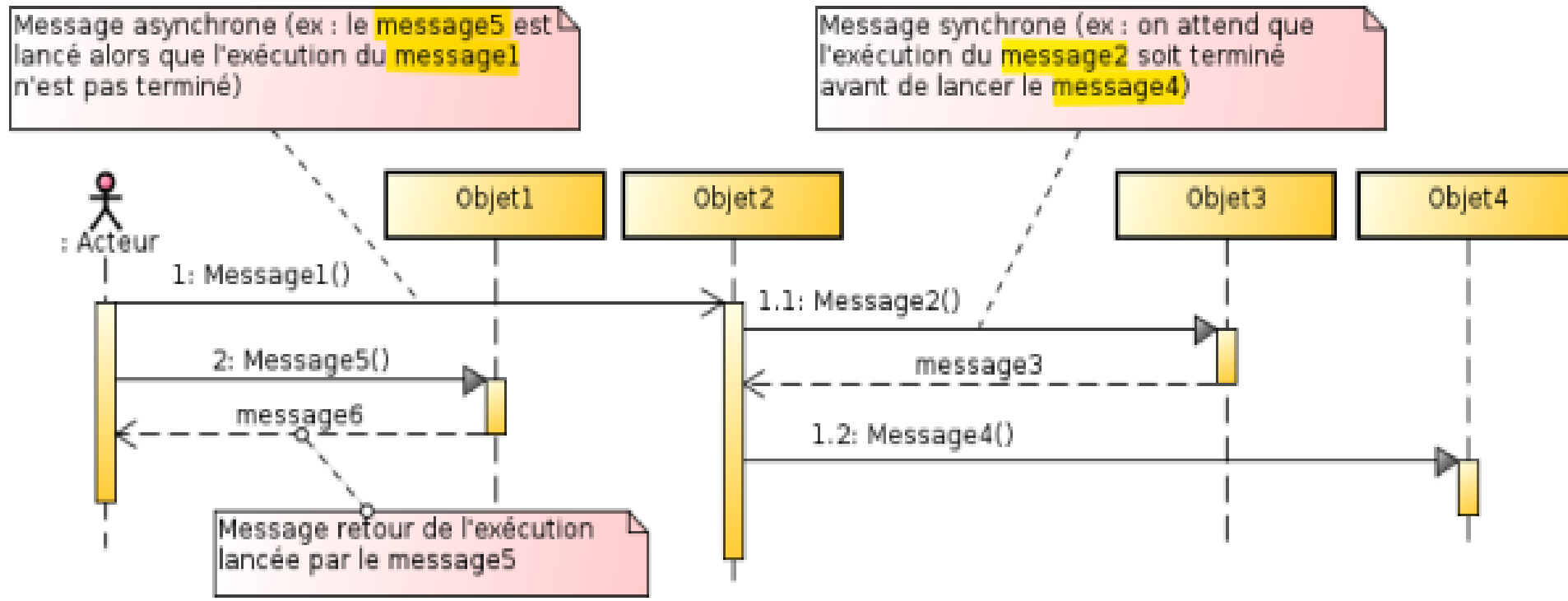
Les types de messages :

- ▶ **message synchrone** : pour lequel l'objet émetteur est bloqué et attend la fin du traitement du message par l'appelé. (————→)
- ▶ **message asynchrone** : pour lequel l'émetteur n'est pas bloqué et peut continuer son exécution même si l'appelé n'a pas fini de traiter le message (———→)
- ▶ **message réflexif**: 
- ▶ **message de retour**: Dans le cas d'un message synchrone, le retour peut ne pas être représenté (c'est implicite car l'émetteur reste bloqué jusqu'à la fin des traitements du message par le récepteur). Il peut être représenté en pointillé (return ou valeur fournie en retour).

UML: DIAGRAMME DES SEQUENCES

Formalisme

Exemple



UML: DIAGRAMME DES SEQUENCES

Fragments combinés

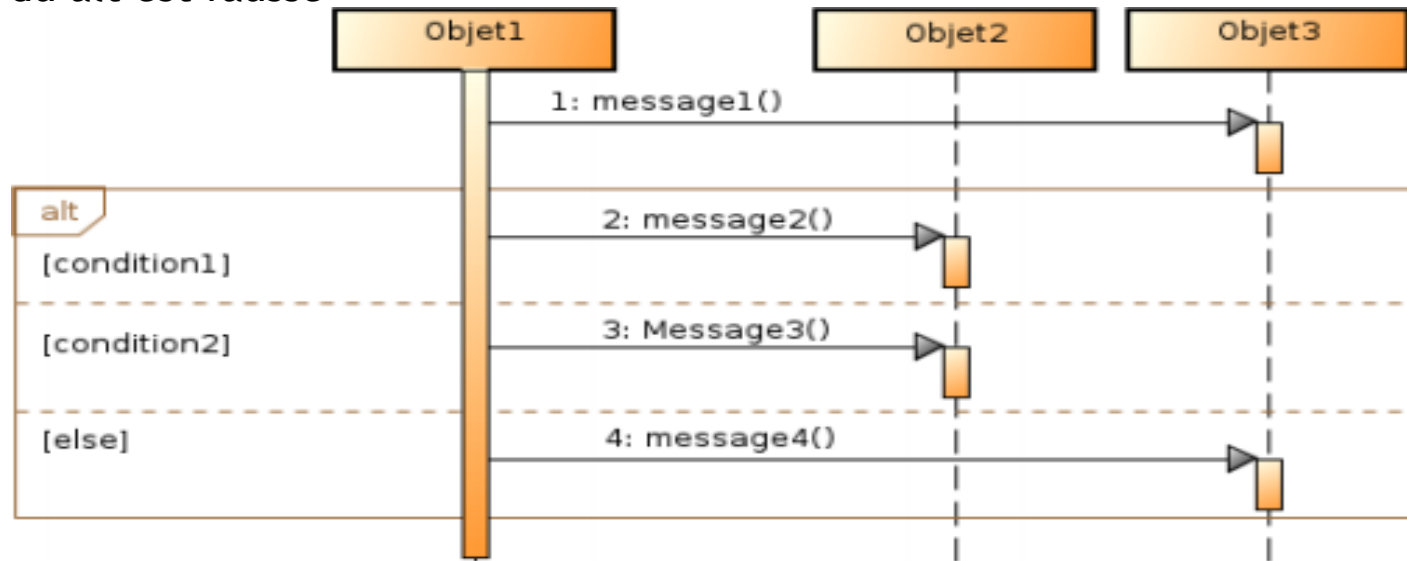
- ▶ Les fragments combinés sont des unités d'interaction définies dans UML. Ils offrent le support de nombreuses constructions comme les alternatives, les boucles, le parallélisme, etc. et confèrent au diagramme de séquence le statut d'un véritable modèle des interactions.
- ▶ Un fragment combiné est une partie du diagramme de séquence. Il est représenté par un rectangle associé à une étiquette. Celle-ci contient un opérateur qui en détermine la modalité d'exécution. Les modalités les plus utilisées sont l'**option**, l'**alternative** et la **boucle**.

UML: DIAGRAMME DES SEQUENCES

Fragments combinés

Le fragment « alt »

- L'opérateur **alt** désigne un choix ou une alternative: équivalent à SI... ALORS... SINON...
- L'utilisation de l'opérateur **else** permet d'indiquer que la branche est exécutée si la condition du alt est fausse



- Une seule branche sera réalisée dans un scénario donné.

UML: DIAGRAMME DES SEQUENCES

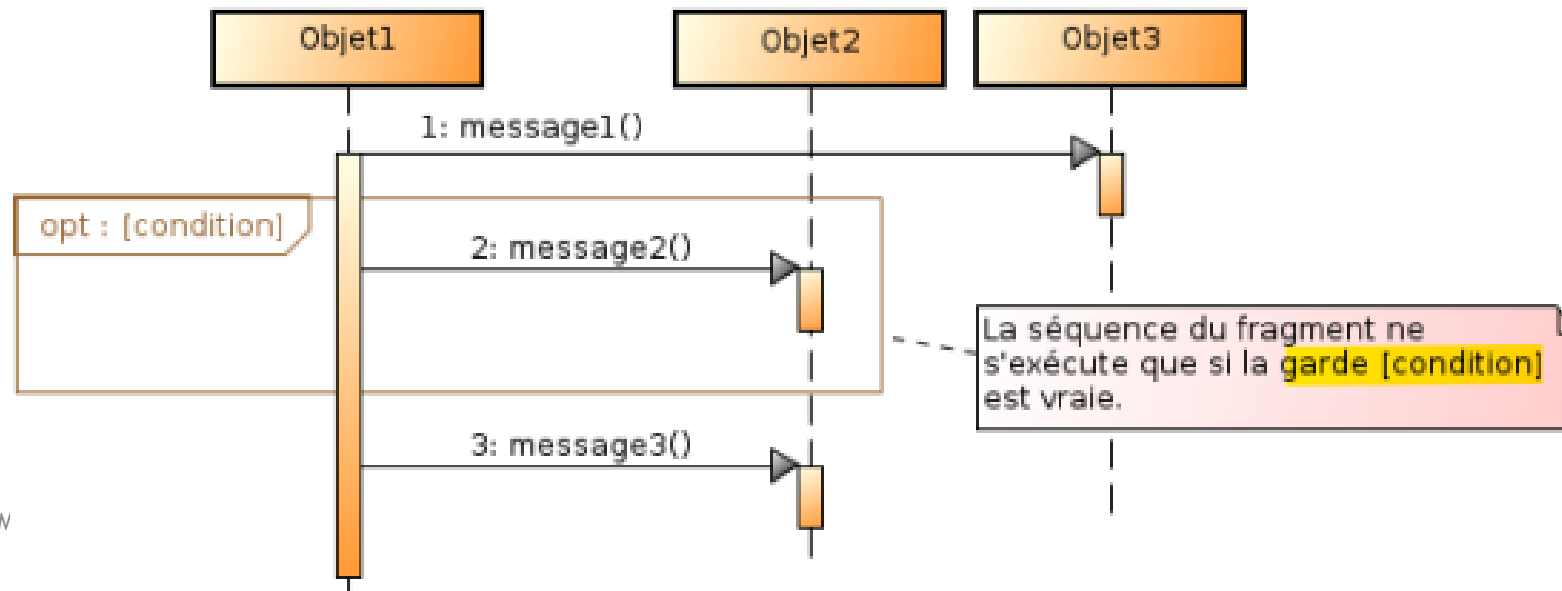
Fragments combinés

Le fragment « opt »

L'opérateur opt désigne un choix de comportement où:

- Soit l'opérateur seule s'exécute
- Soit rien ne s'exécute

Il est équivalent à SI... ALORS... ↔ il est équivalent à « alt » sans [else]



UML: DIAGRAMME DES SEQUENCES

Fragments combinés

Le fragment « loop »

L'opérateur de boucle (**loop**) exécute une itérative dont la séquence qu'elle contient est exécutée tant que la garde qui lui est associée est vraie.

La garde s'écrit de la façon suivante:

Loop [**min**, **max**, **condition**]: chaque paramètre (**min** , **max** et **condition**) est optionnel,

Le contenu du cadre est exécuté min fois, puis continue à s'exécuter tant que la condition et le nombre d'exécution de la boucle ne dépasse pas max fois.

Exemple de gardes:

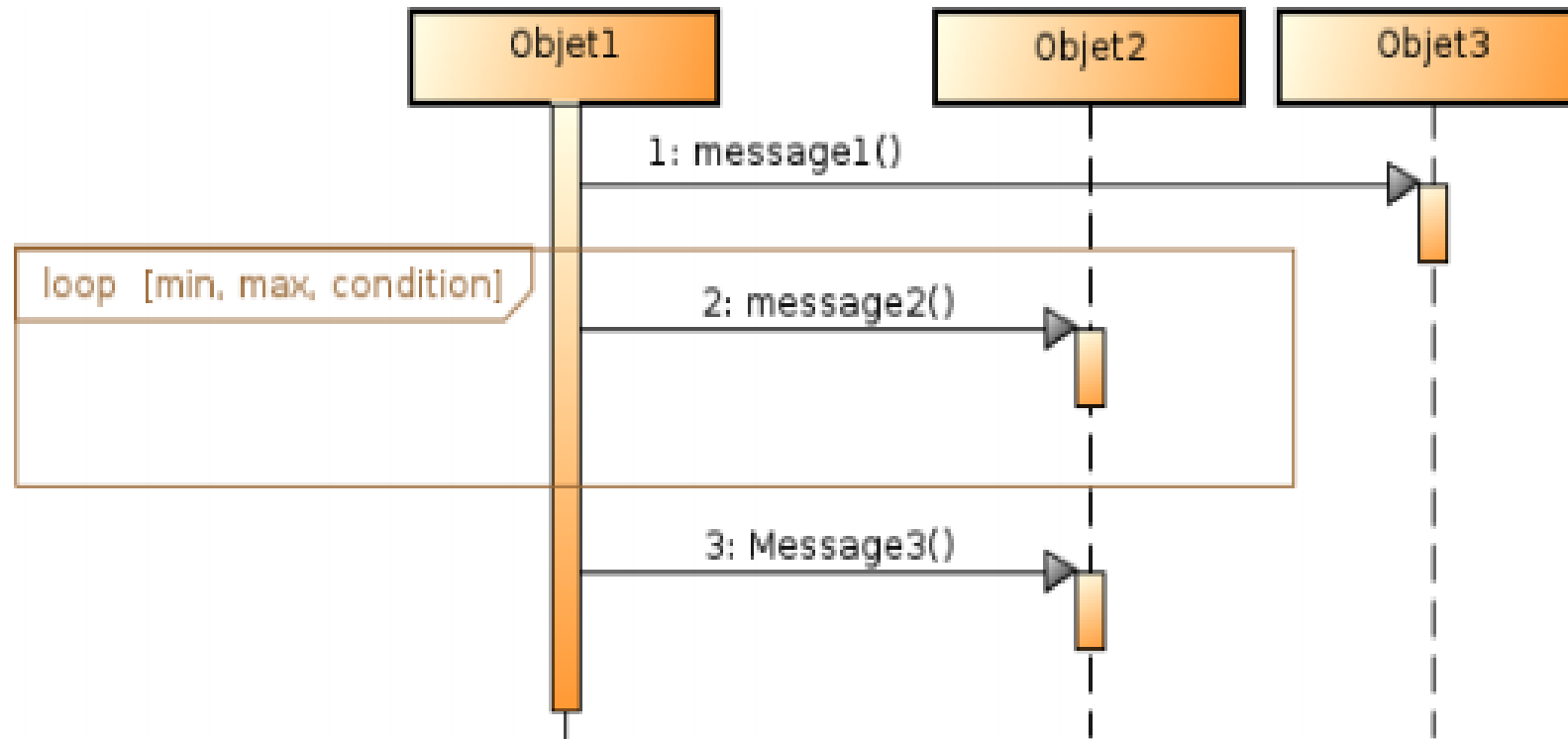
Loop[3] ➔ la séquence s'exécute 3 fois,

Loop[1,3, **code = faux**] la séquence s'exécute 1 fois puis un maximum de 2 autres fois si **code = faux**

UML: DIAGRAMME DES SEQUENCES

Fragments combinés

Le fragment « loop »

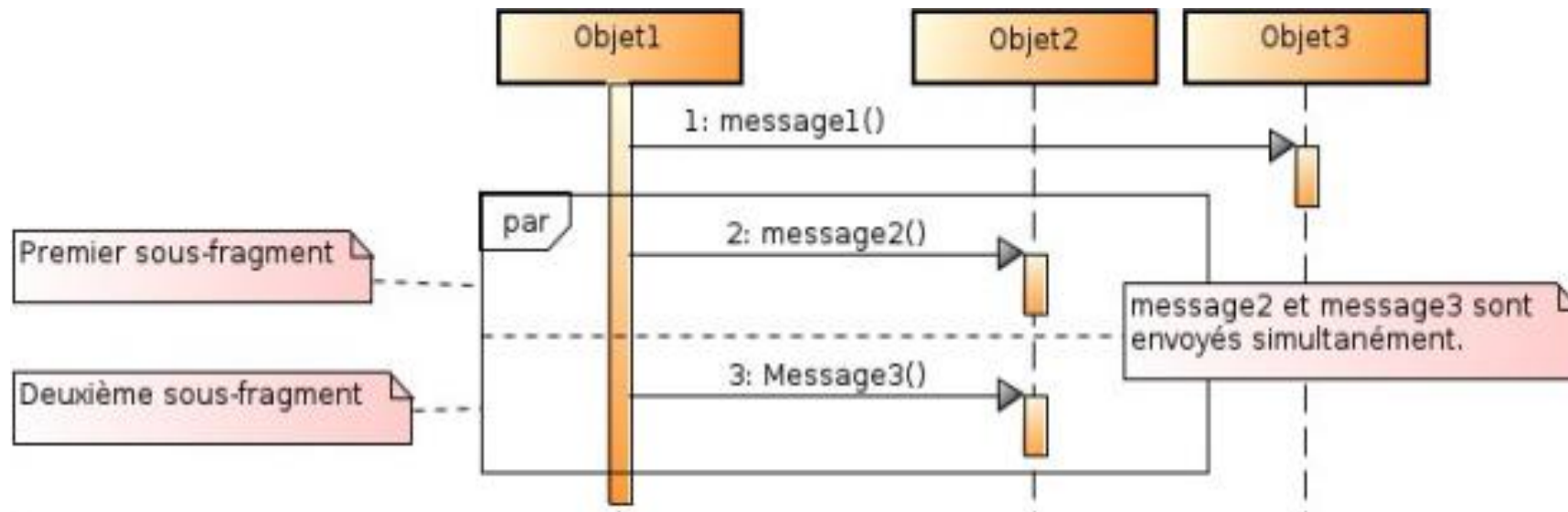


UML: DIAGRAMME DES SEQUENCES

Fragments combinés

Le fragment « par »

Un fragment d'interaction avec l'opérateur de **traitements parallèles** (**par**) contient au moins deux sous fragments (**opérandes**) séparés par des pointillés qui s'exécutent simultanément (**traitements concurrents**).



UML: DIAGRAMME DES SEQUENCES

Exemple

