

LOG121

Conception orientée objet

Diagrammes de séquences UML

Enseignante: Souad Hadjres

Diagramme de séquences

2

- Un diagramme de séquence montre la dynamique d'un scénario
 - ▣ Montre comment se déroule le scénario
 - ▣ C'est un diagramme d'objets

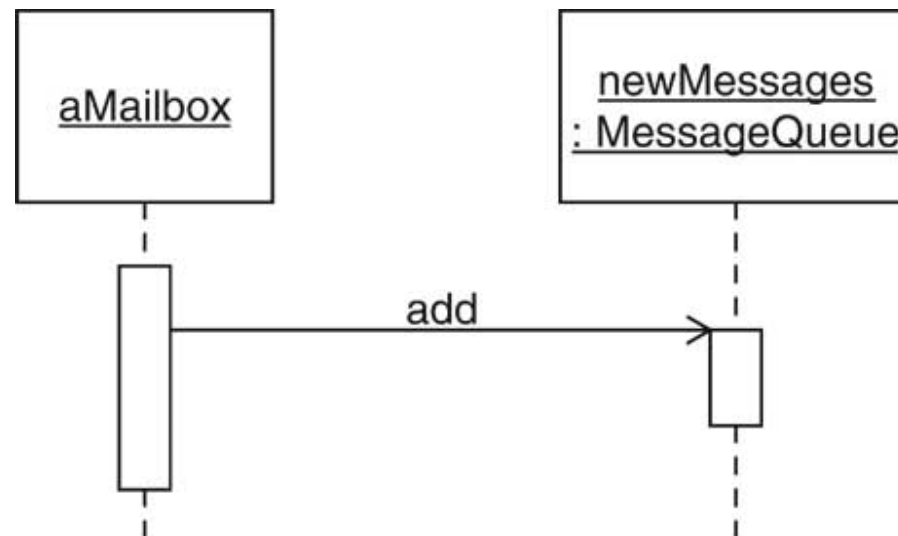


Diagramme de séquences

3

- Il décrit dans un ordre chronologique l'interaction entre les objets
- Il montre les objets impliqués dans un scénario
- Il montre l'ordre d'appels des méthodes de ces objets

Diagramme de séquences

4

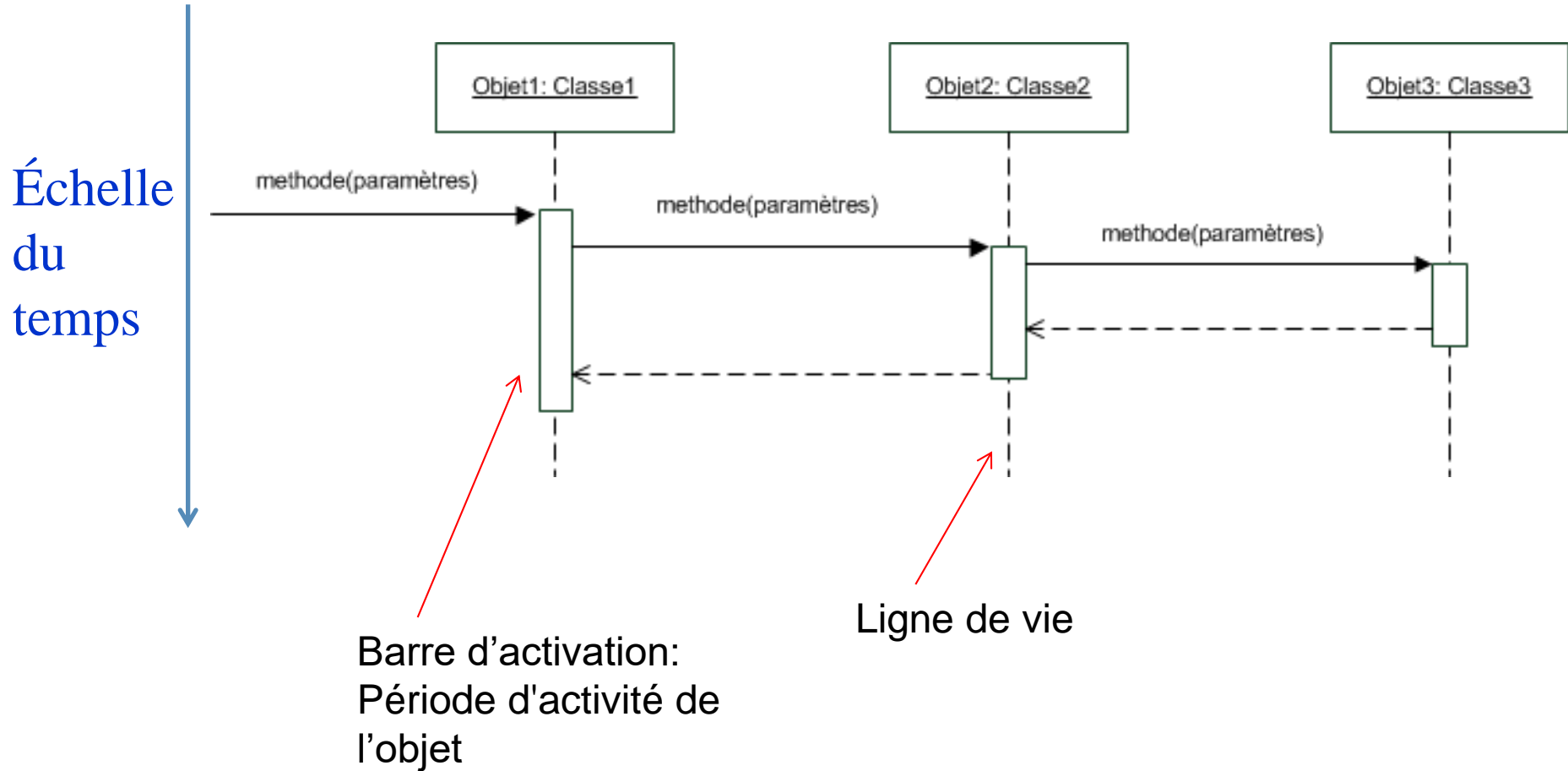
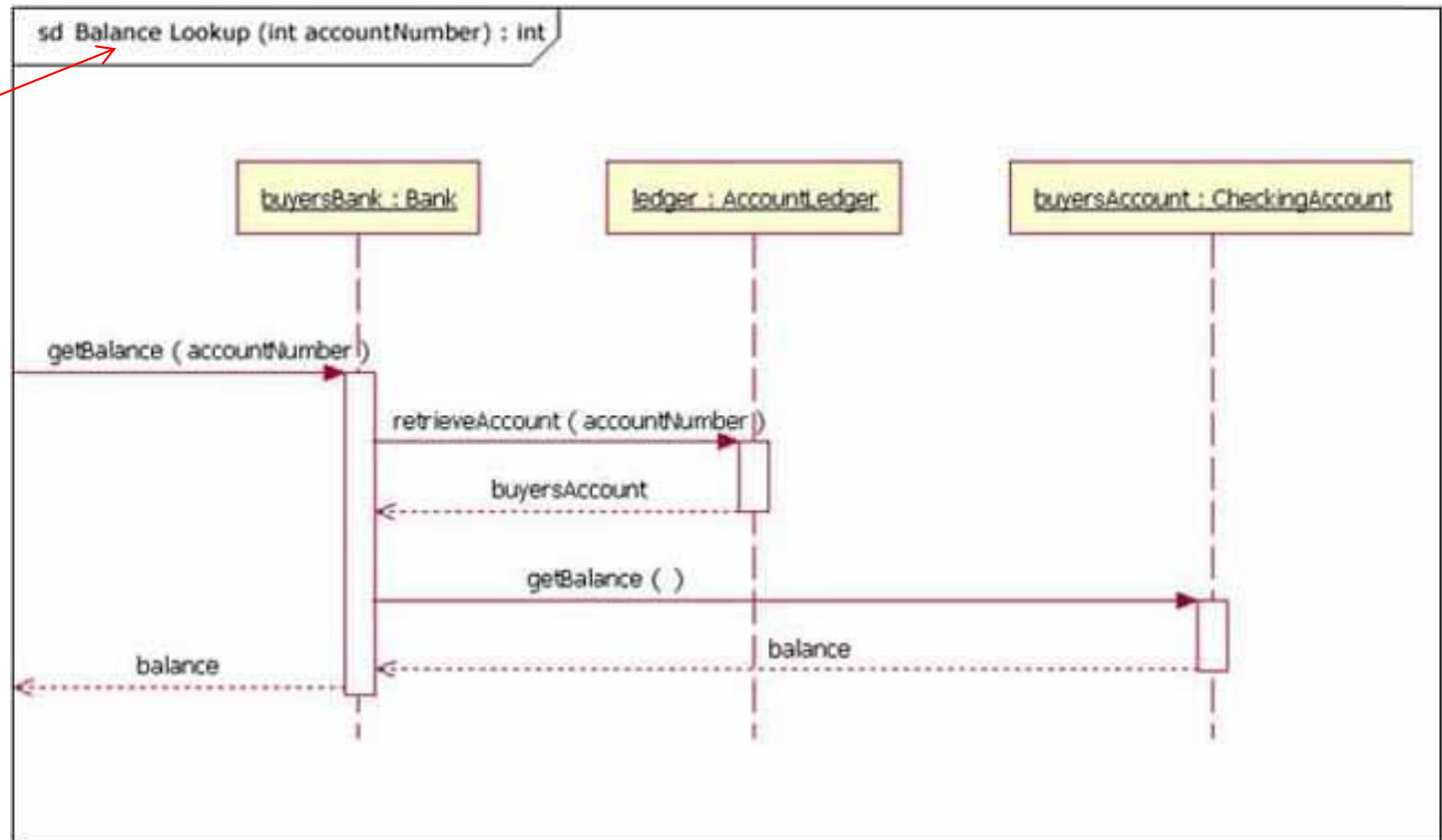


Diagramme de séquences

5

□ Exemple

Nom du diagramme de séquence et ses paramètres



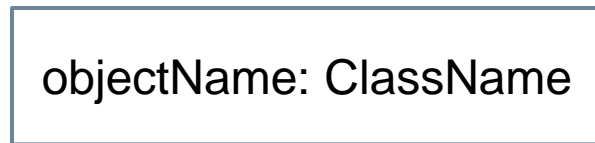
Source: <http://www.ibm.com/developerworks/rational/library/>

Diagramme de séquences

6

- Deux façons de désigner l'objet interagissant

Instance nommée de la classe



Instance anonyme de la classe

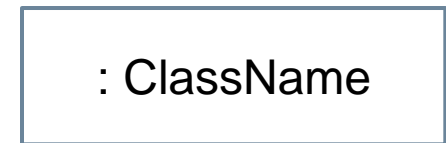


Diagramme de séquences

7

□ Notation des collections

Instance d'une collection contenant des objets de type Vente

ventes : ArrayList<Vente>



A rectangular box containing the text 'ventes : ArrayList<Vente>'. A vertical dashed line extends downwards from the bottom center of the box.

Instance sélectionnée dans une collection

ventes[i] : Vente



A rectangular box containing the text 'ventes[i] : Vente'. A vertical dashed line extends downwards from the bottom center of the box.

Diagramme de séquences

8

□ Représentation des classes

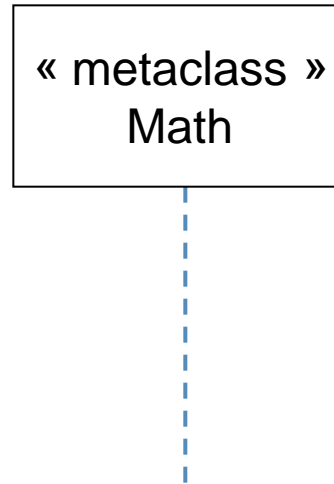


Diagramme de séquences

9

□ Représentation des interactions

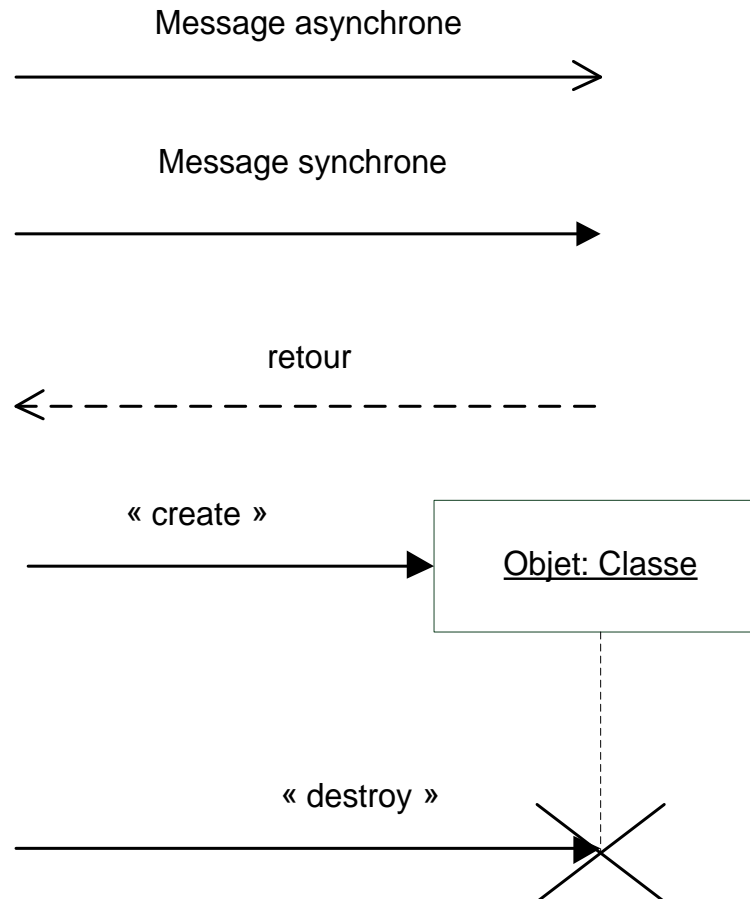


Diagramme de séquences

10

- Deux façons d'illustrer le retour d'un message

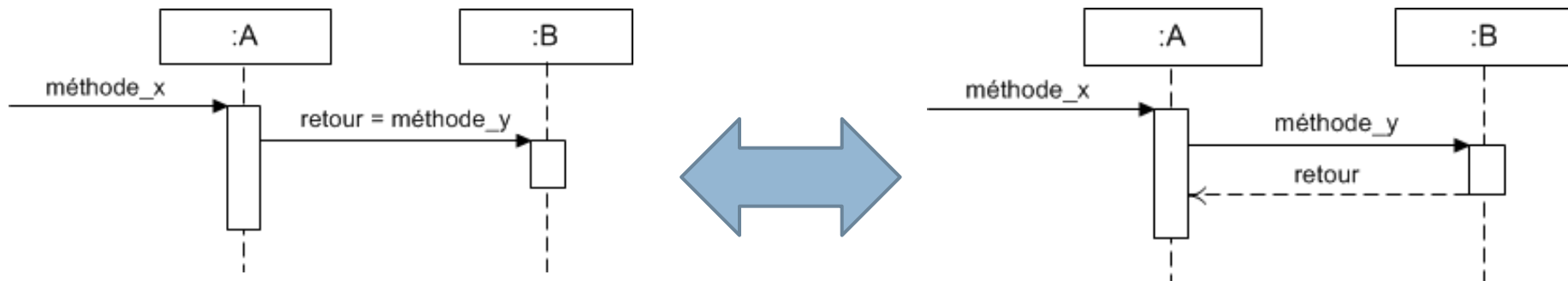


Diagramme de séquences

11

- Syntaxe des messages échangés entre objets

`variable_retour = message(paramètre : type_paramètre) : type_retour`

- Exemples

`créer_compte()`

`retirer(montant)`

`retirer(montant: Monnaie)`

`d = getDateCréation()`

`solde = getSolde(c:Compte): Monnaie`

Diagramme de séquences

12

- Un objet peut créer un autre objet
 - ▣ Placer l'instance créée au bon endroit selon le moment où elle a été créée

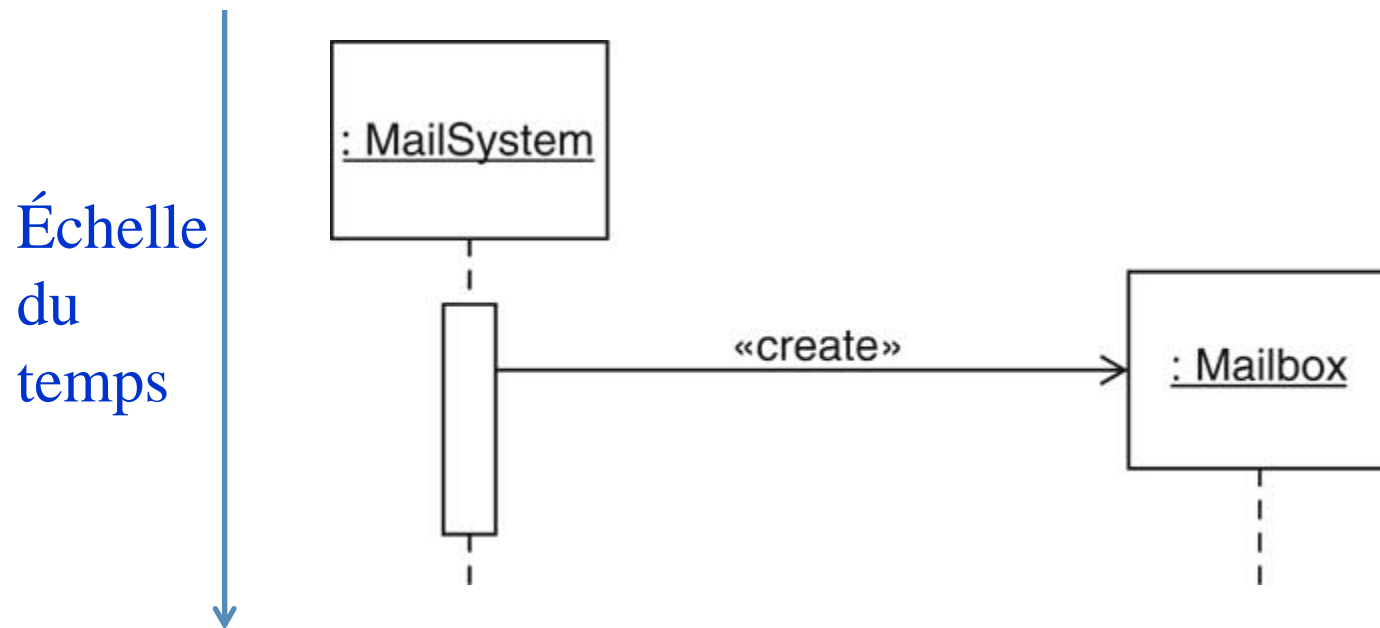


Diagramme de séquences

13

- Un objet peut détruire un autre objet
 - ▣ Il faut terminer la ligne de vie de l'objet détruit

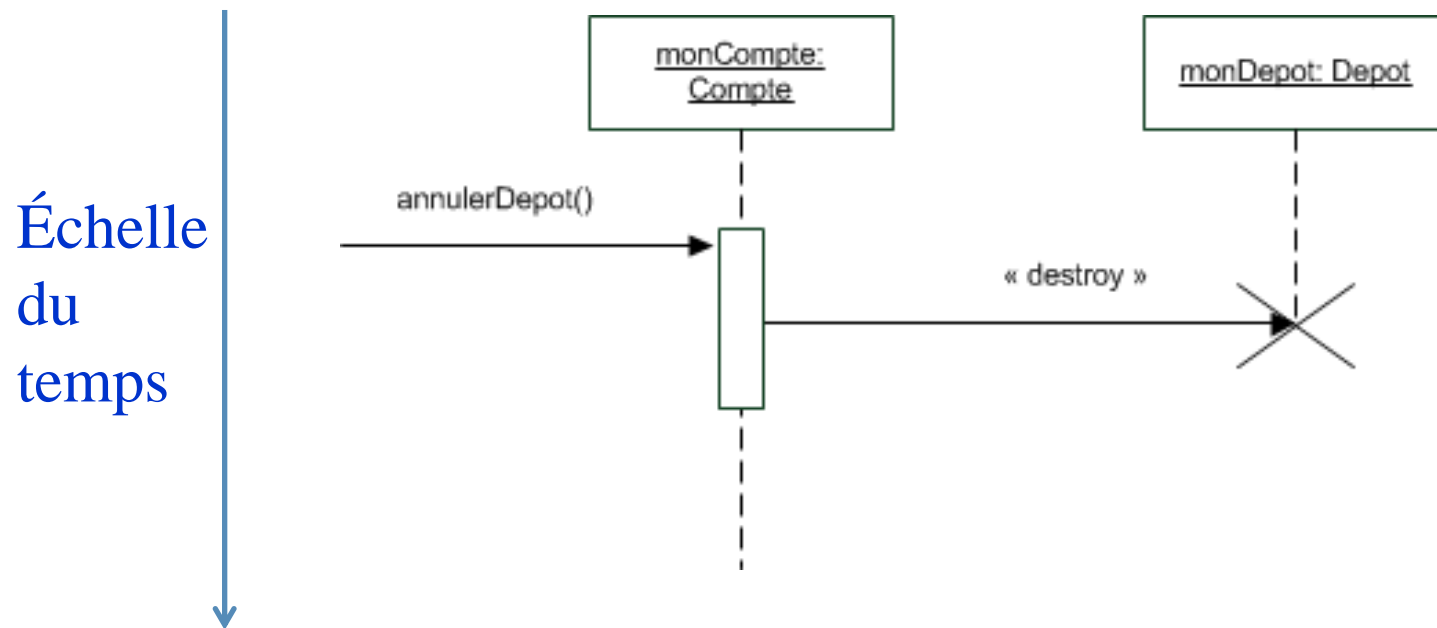


Diagramme de séquences

14

- Une méthode peut appeler une autre méthode du même objet

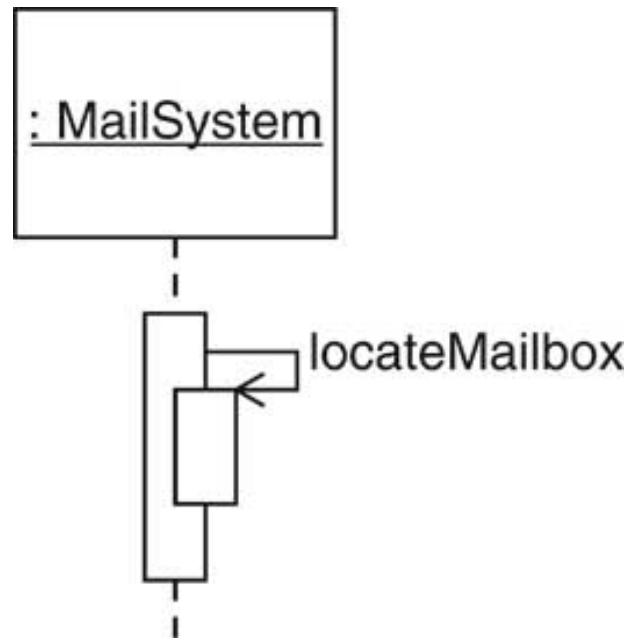
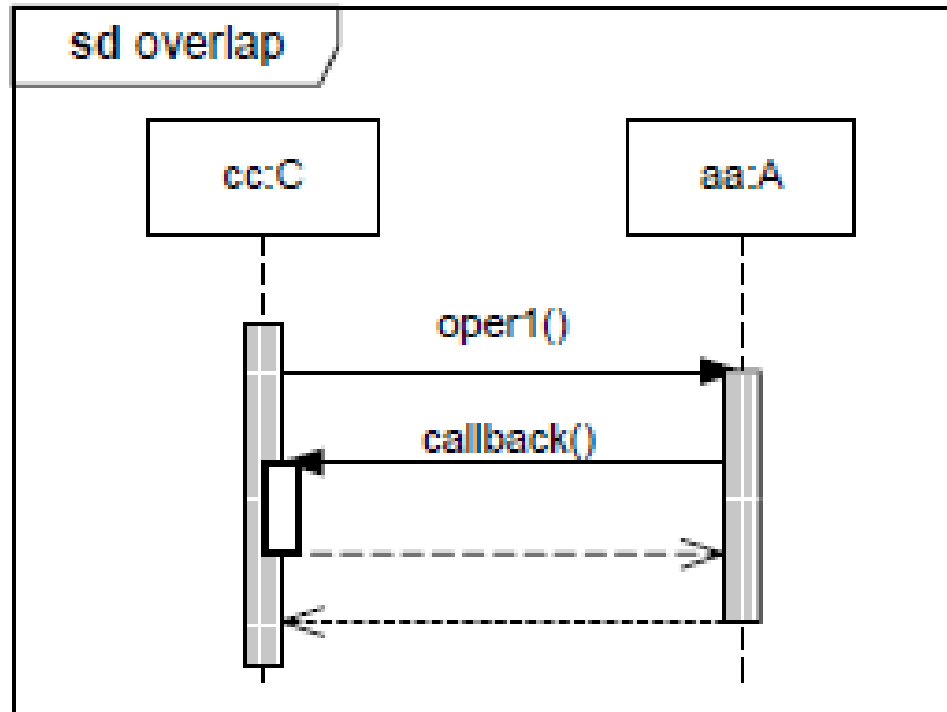


Diagramme de séquences

15

- Appel chevauchant: Exemple extrait de la spécification UML



<http://www.uml.org/>

Diagramme de séquences

16

- Penser à une pile d'exécution

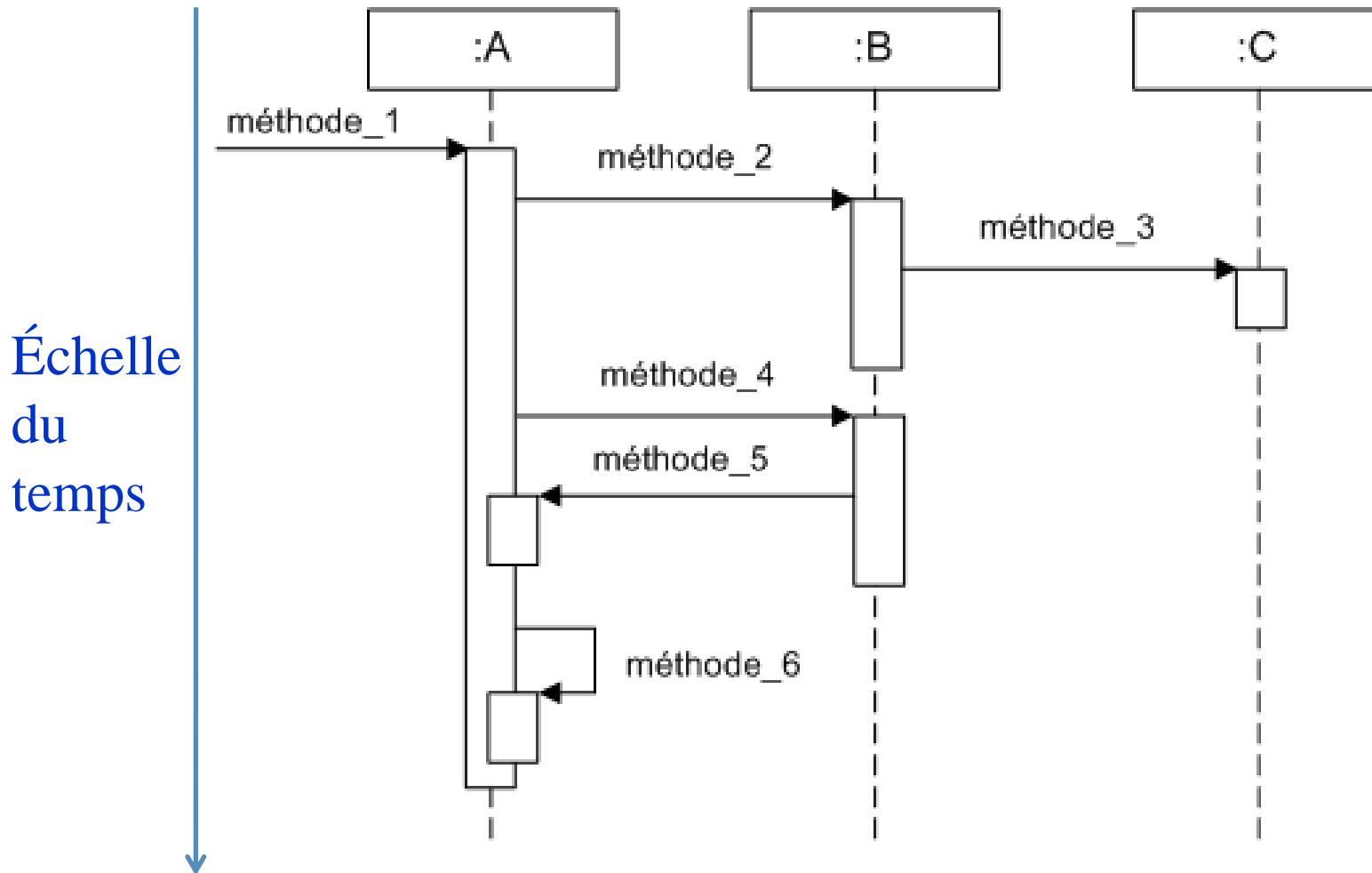


Diagramme de séquences

17

- Cas d'utilisation : Laisser un message (Extrait du livre de référence)

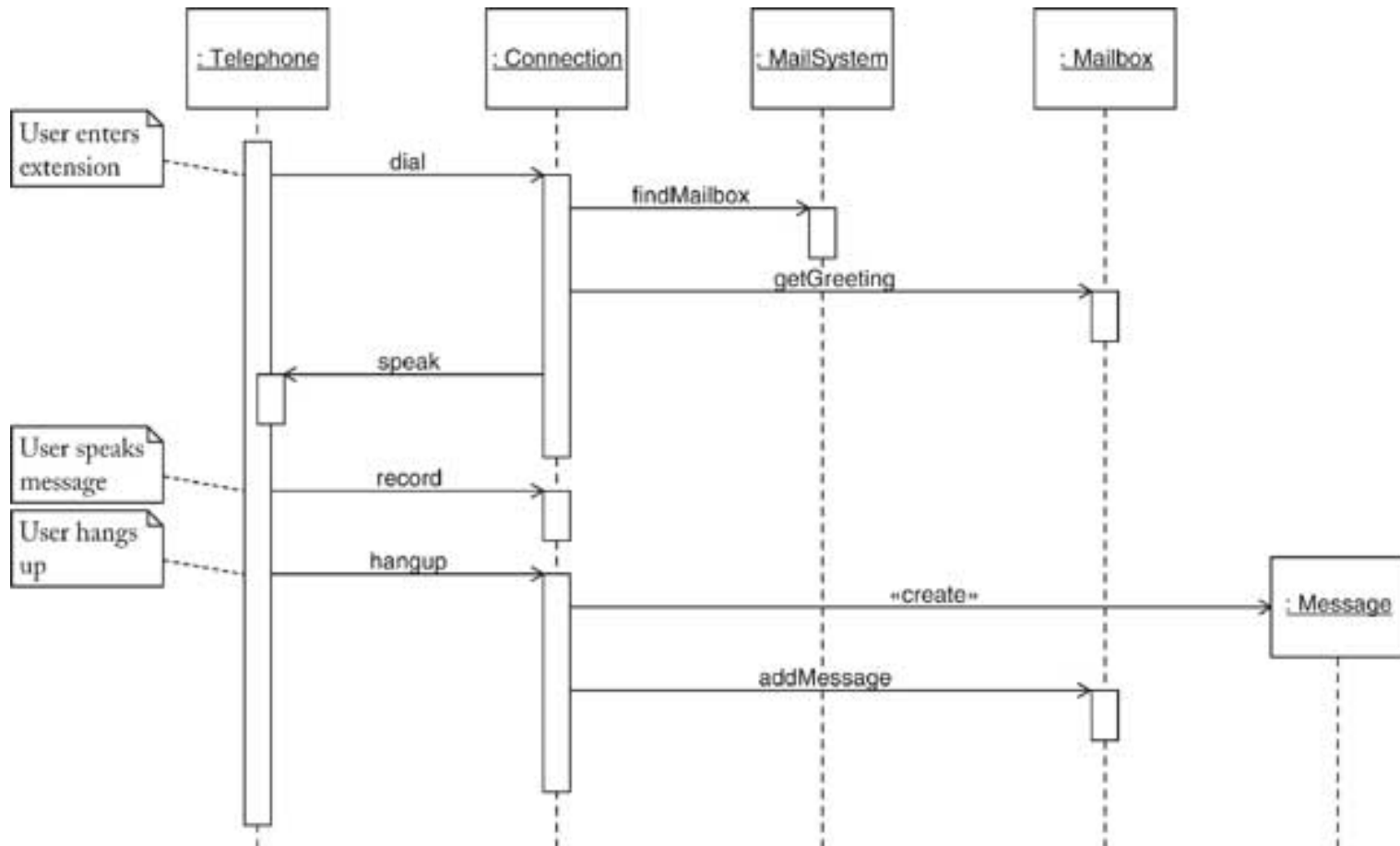


Diagramme de séquences

18

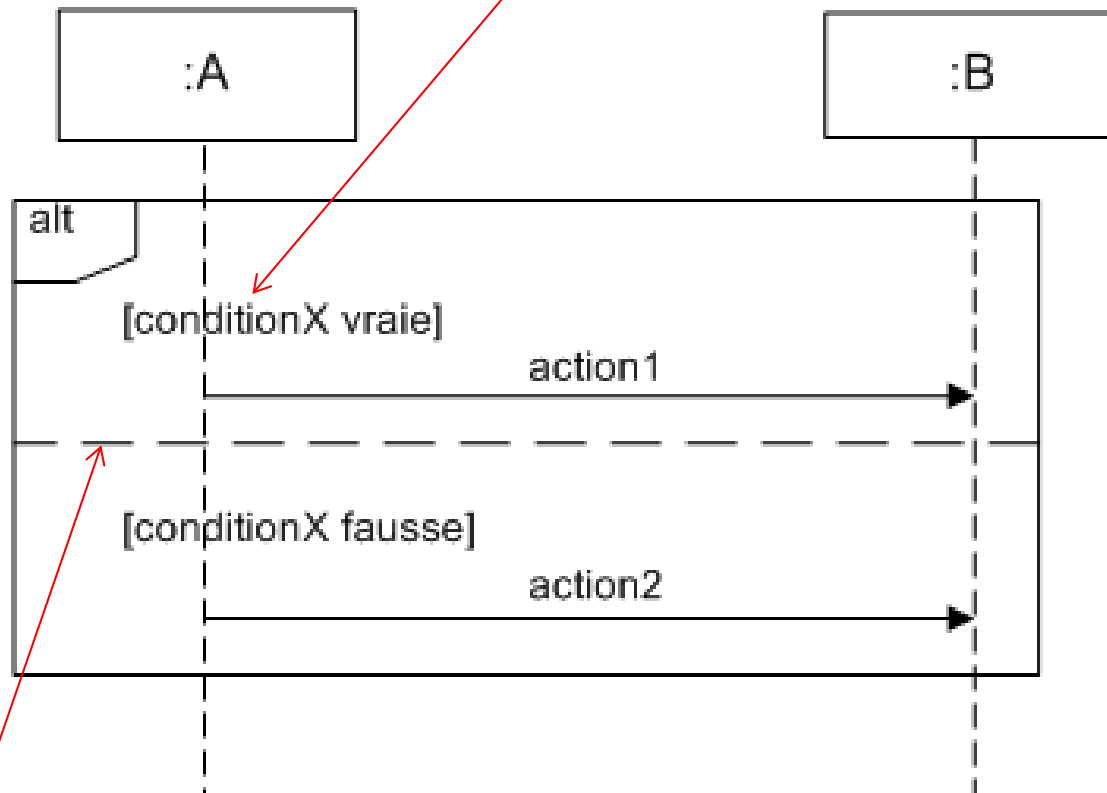
- UML offre d'autres notations qui permettent de
 - ▣ représenter des boucles et des alternatives
 - ▣ faire appel à un autre diagramme de séquence pour des raisons de lisibilité et compréhension
 - ▣ représenter des ensembles d'interactions strictement séquentiels ou parallèles

Diagramme de séquences

19

□ Notation des alternatives

La condition est placée sur la ligne de vie de l'objet qui effectue sa vérification.

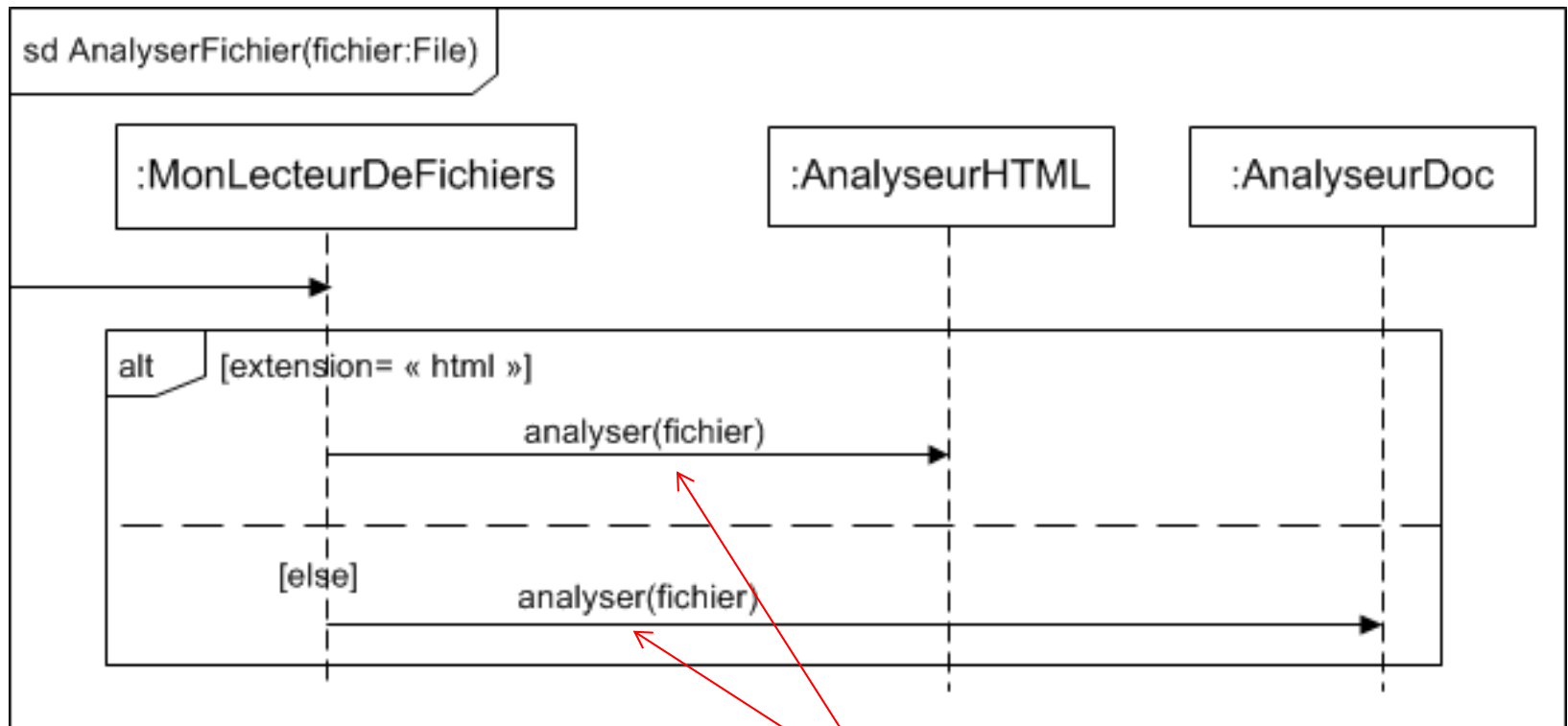


Les séquences alternatives sont séparées par une ligne

Diagramme de séquences

20

□ Notation des alternatives: Exemple



Messages conditionnels mutuellement exclusifs

Diagramme de séquences

21

□ Messages conditionnels (optionnels) : Exemple

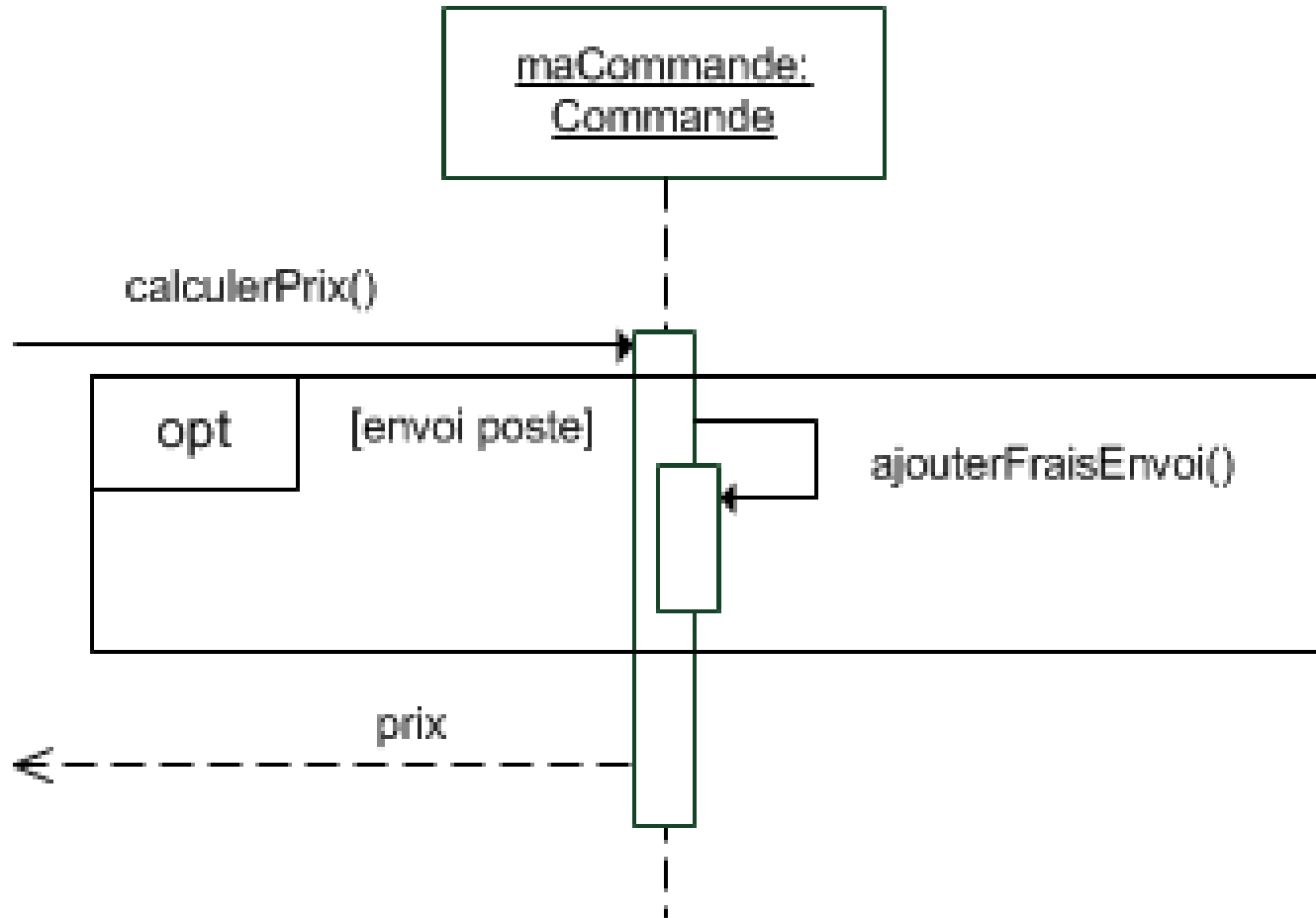


Diagramme de séquences

22

□ Notation des boucles: Exemple

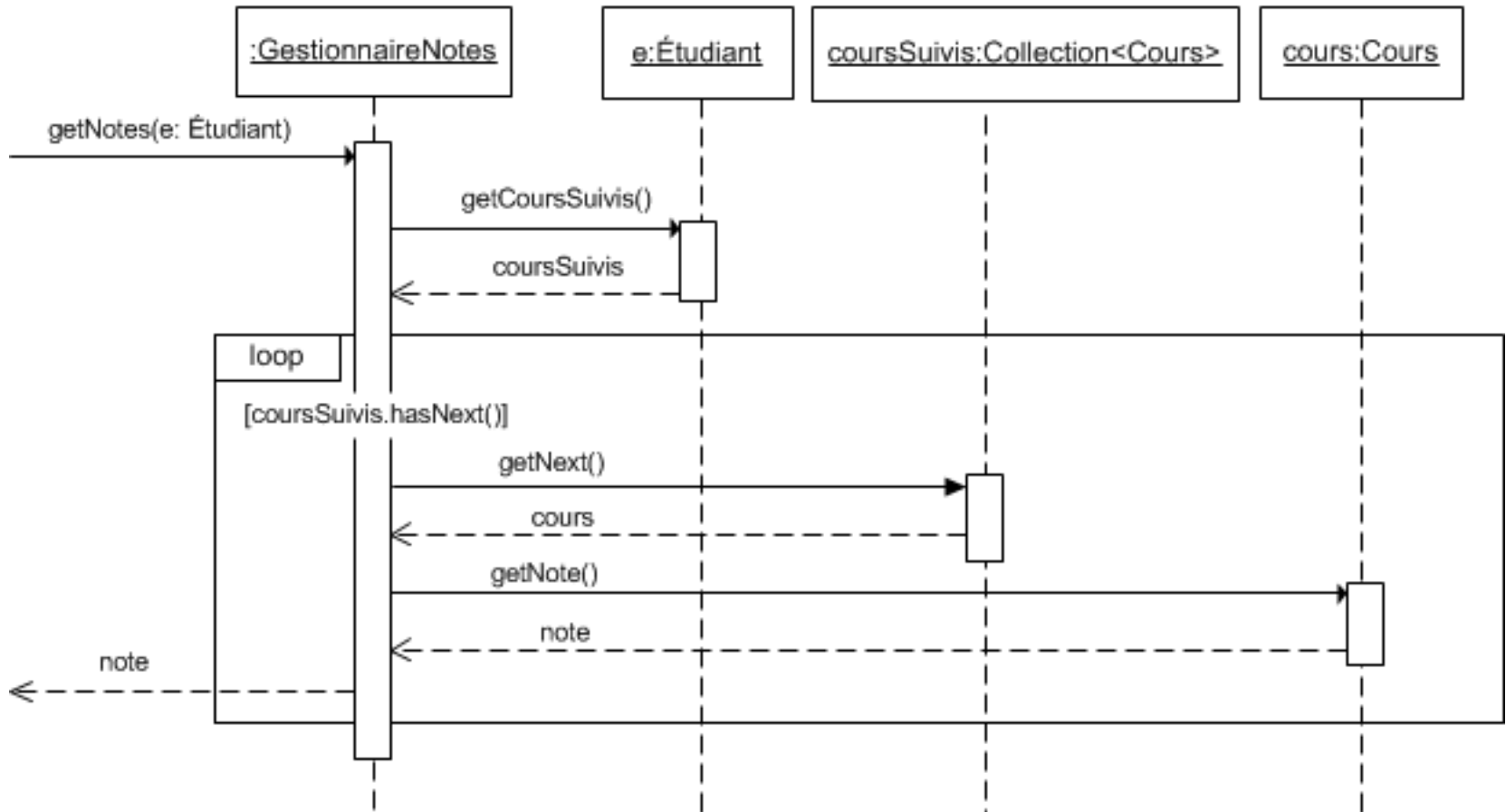
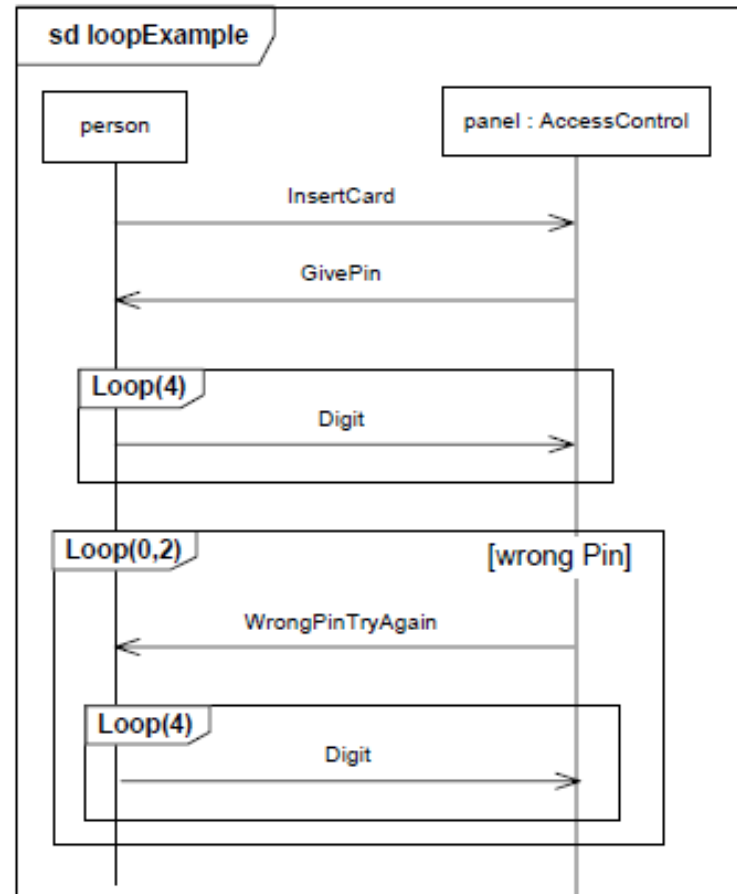


Diagramme de séquences

23

- Exemple extrait de la spécification UML montrant un exemple avec des boucles imbriquées



<http://www.uml.org/>