

Diagramme de séquences

Formateur: Y.Naji

Analyse et conception orientée objet (UML)

Définitions

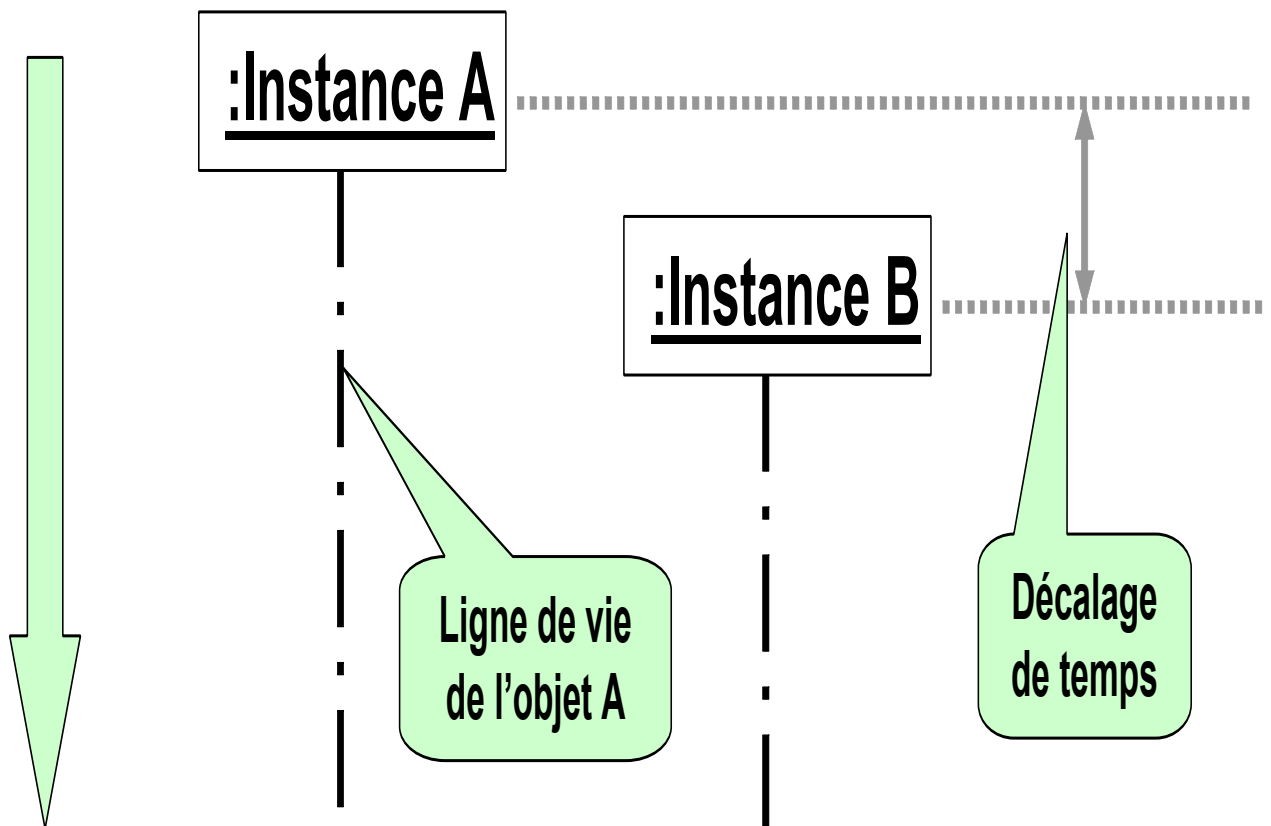
- Le diagramme de séquences est une documentation des cas d'utilisation.
- Le diagramme de séquences représente la succession chronologique des opérations réalisées par un acteur.
- Une interaction est la description d'un comportement qui se focalise sur l'échange observable d'informations entre **instances** (**acteurs** ou **objets**) connectables.
- Le diagramme de séquences est un diagramme d'interaction qui représente les objets participant à une interaction particulière.
- Il représente les messages échangés entre les objets.
- Ces messages seront transformés en opérations de classe lors de la phase de conception.

Les composants du diagramme

- La ligne de vie d'un objet
- Le message
- Barre d'activation d'un objet
- Exemple
- Création et suppression d'un objet

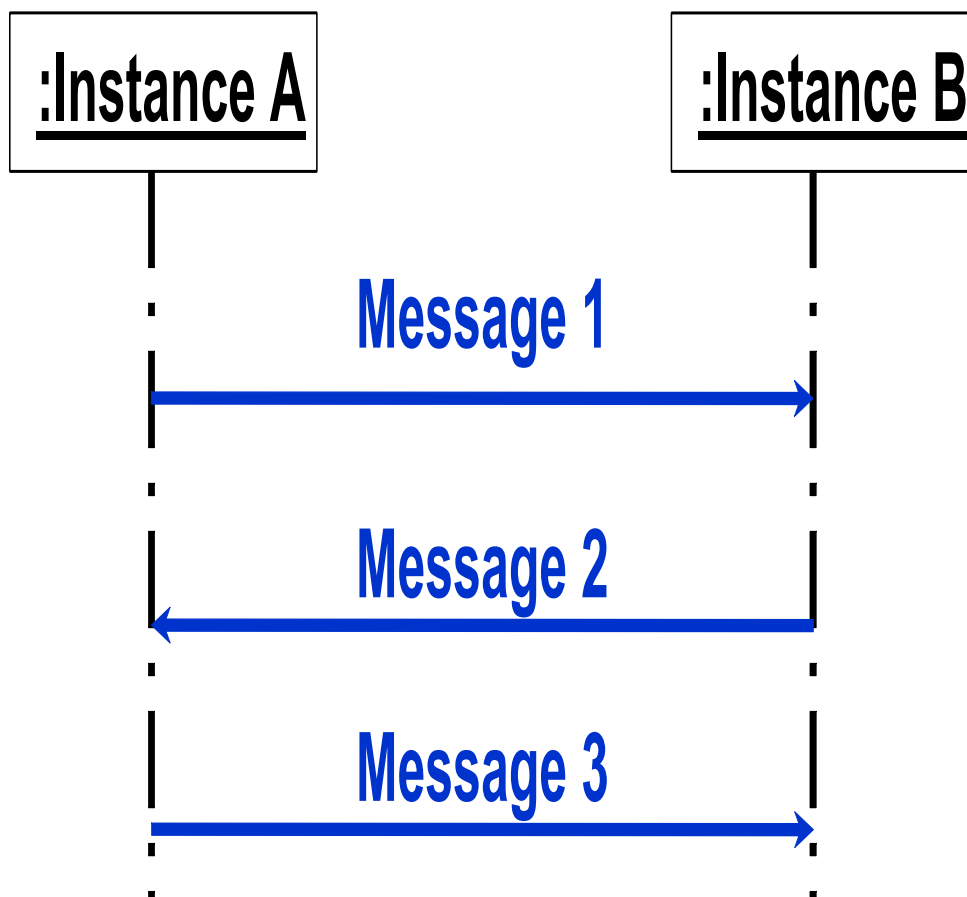
La ligne de vie d'un objet

- C'est un composant qui représente la vie d'un objet dans le contexte de la séquence d'événements.
- Par convention, le temps est représenté comme s'écoulant du haut vers le bas le long des "lignes de vie" des instances.
- Les objets qui sont créés vers la fin de la séquence n'apparaissent pas toujours en haut du diagramme, mais peuvent apparaître à l'endroit où ils sont créés.



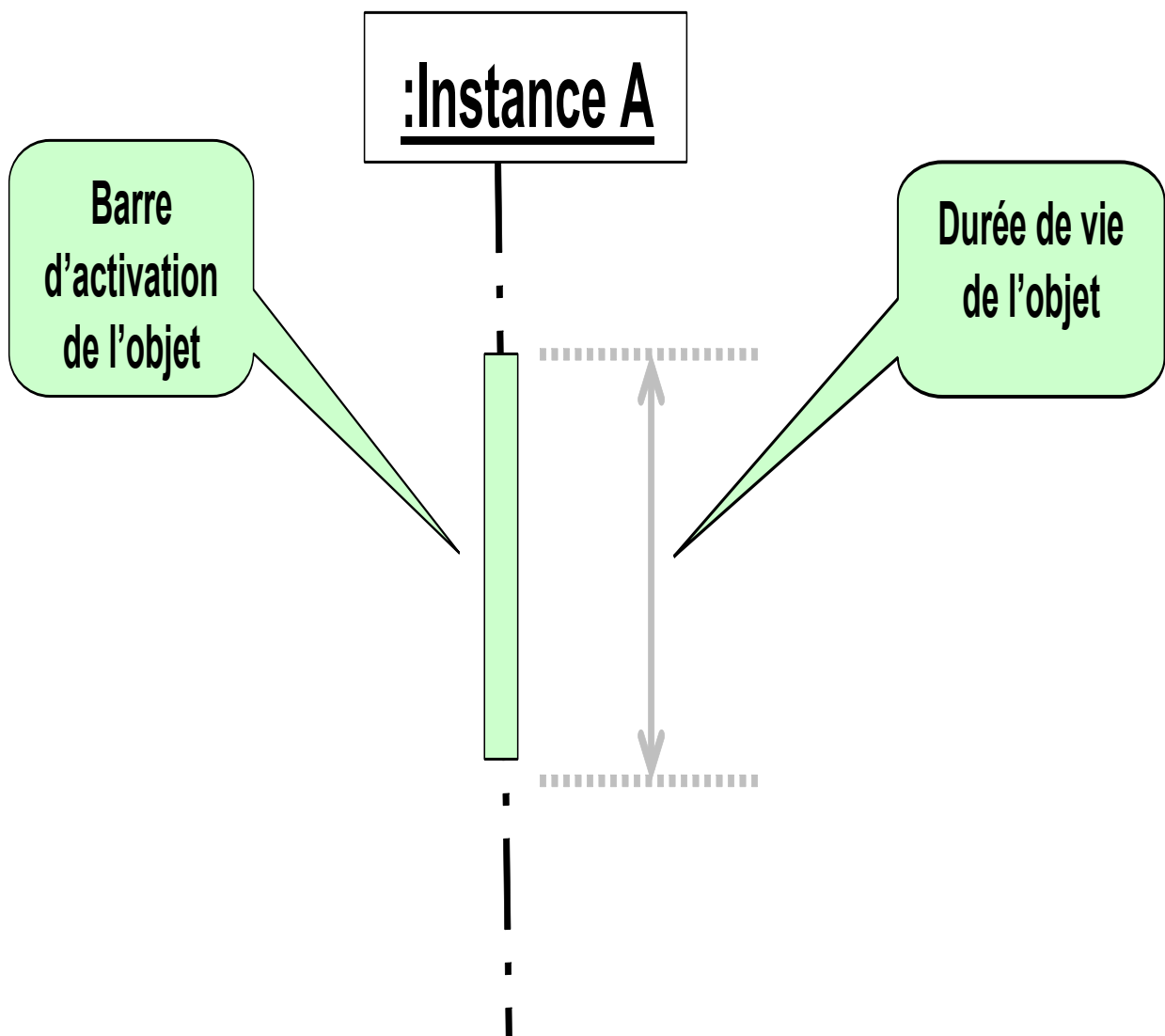
Le message

- Un message est une information circulant entre deux instances selon l'ordre chronologique (du haut vers le bas). Il existe plusieurs types de messages.



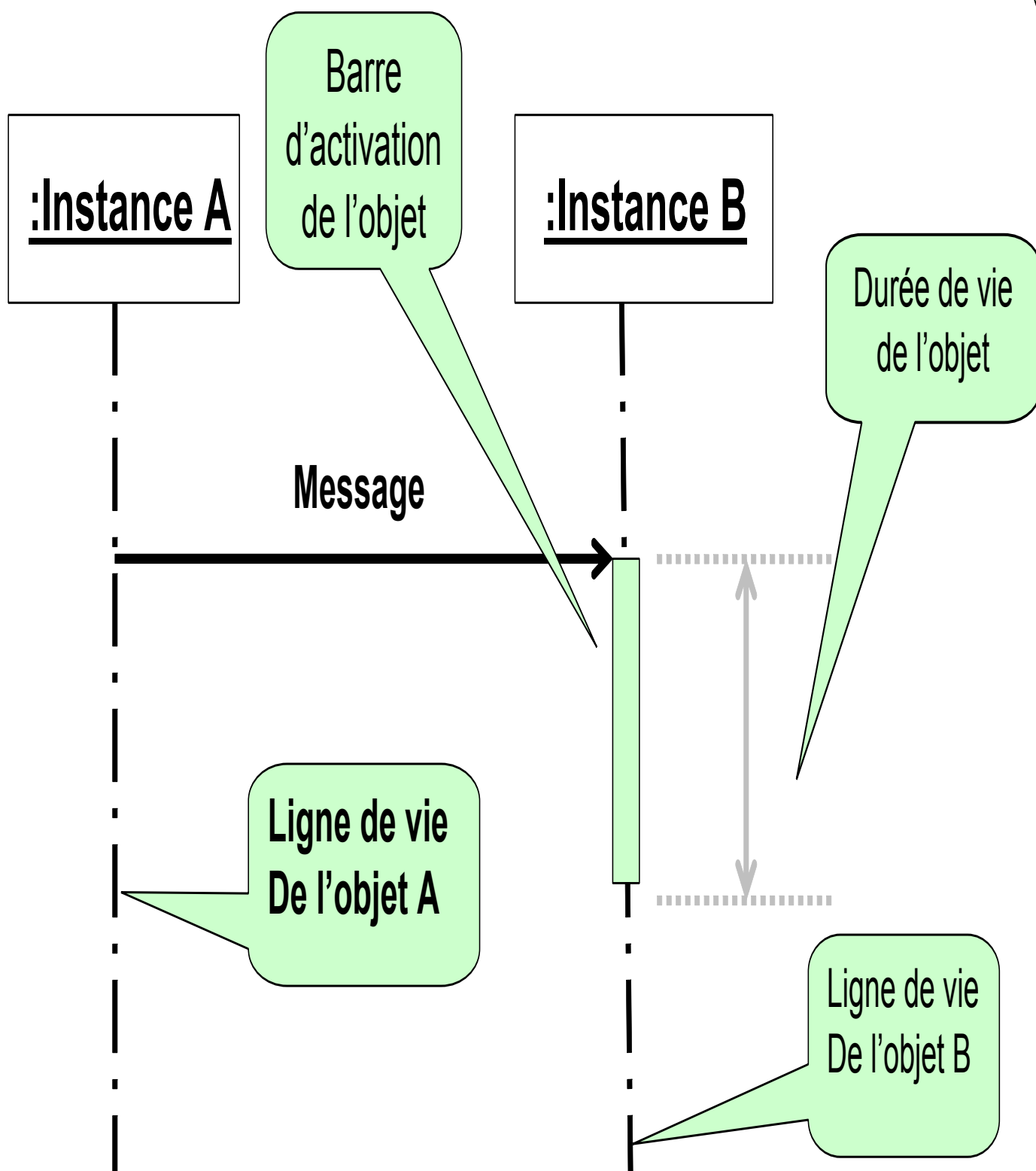
Barre d'activation d'un objet

- On représente les différentes périodes d'activité d'un objet au moyen d'une barre rectangulaire superposée à la ligne de vie de l'objet.



Exemple

Unité: Diagramme de séquences

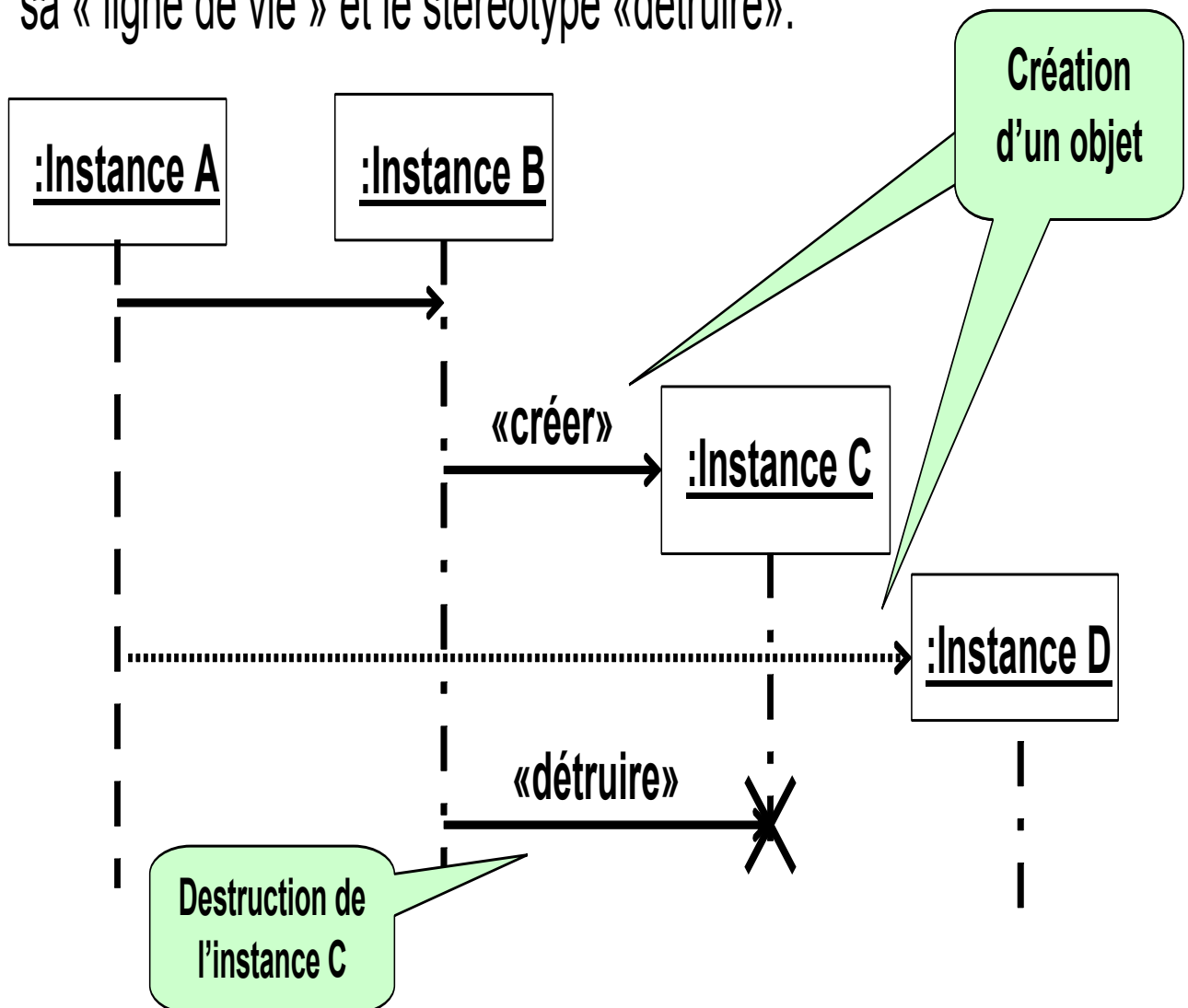


Formateur: Y.Naji

Analyse et conception orientée objet (UML)

Création et suppression d'objet

- On représente la création d'un objet au moyen d'une ligne pointillée allant de l'objet source vers l'objet destination, ou bien avec une flèche simple et le stéréotype «créer».
- La mort (suppression) d'un objet est représentée par une "croix" sur sa « ligne de vie » et le stéréotype «détruire».



Types de messages (1/5)

- Message simple

C'est un message dont on ne spécifie aucune caractéristique d'envoi ou de réception particulière



- Message minuté

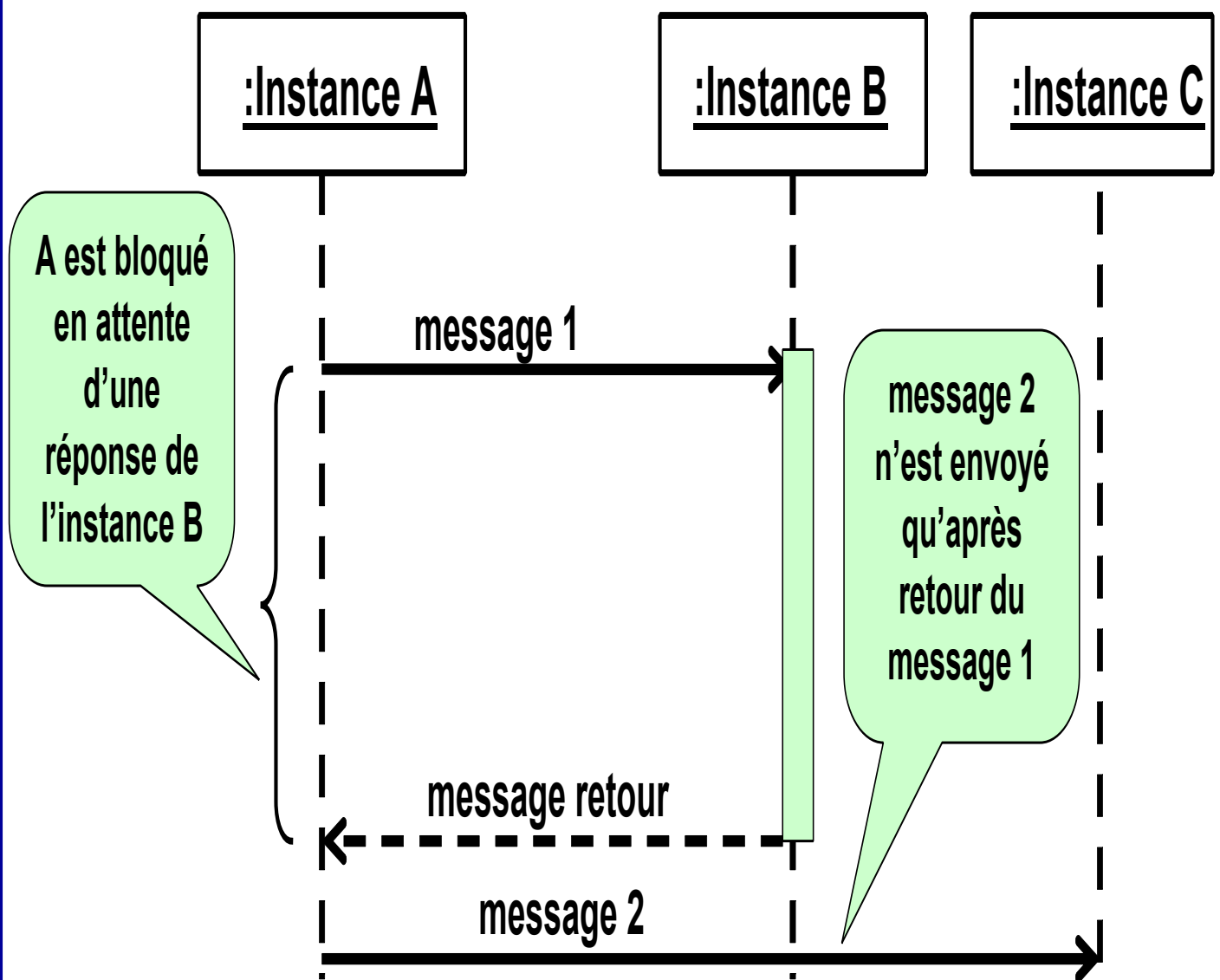
C'est un message qui **bloque** l'expéditeur pendant un temps donné en attendant la prise en compte du message par le récepteur. L'expéditeur est libéré si la prise en compte n'a pas eu lieu pendant le délai spécifié.



Types de messages (2/5)

● Message synchrone:

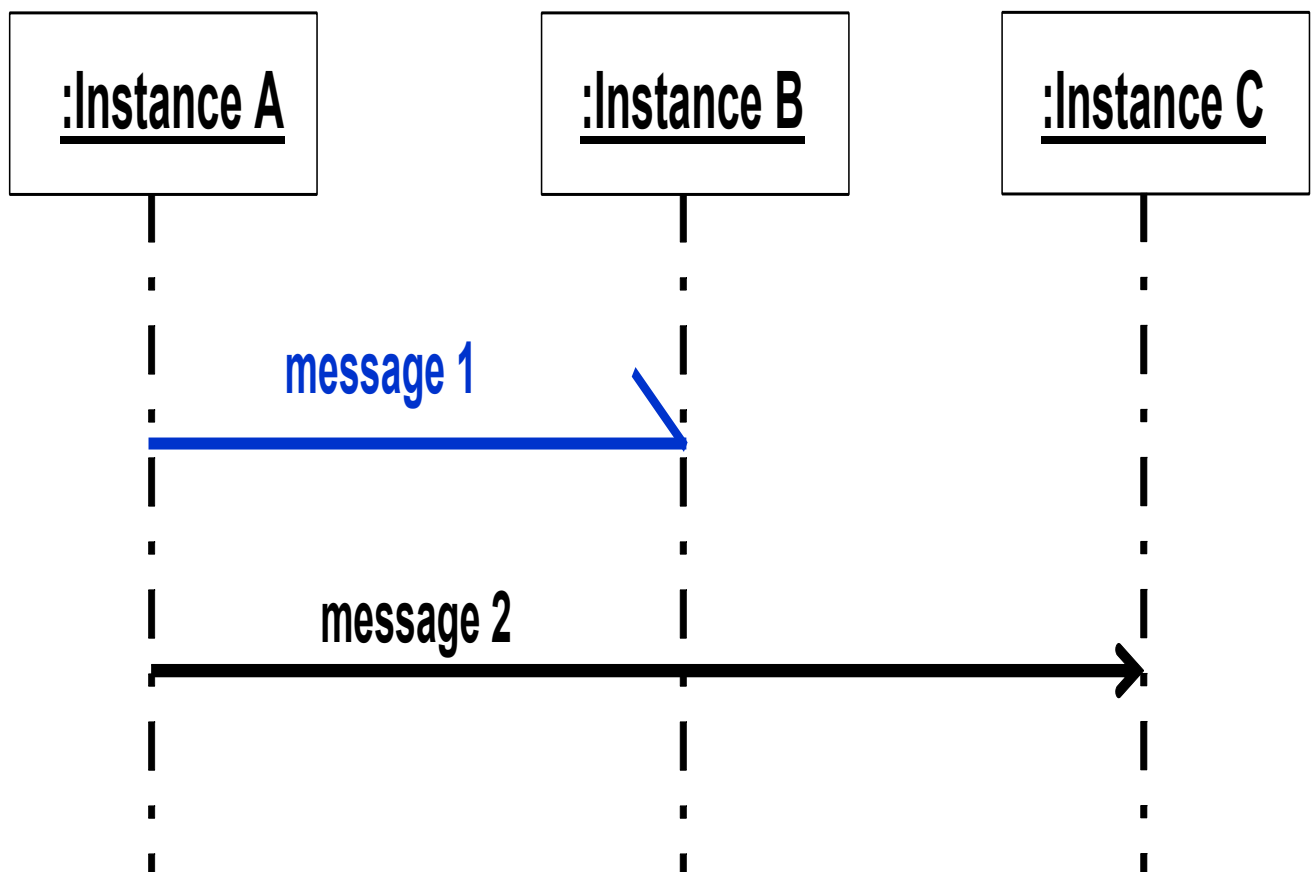
C'est un message qui bloque l'expéditeur jusqu'à prise en compte du message par le destinataire.



Types de messages (3/5)

● Message asynchrone:

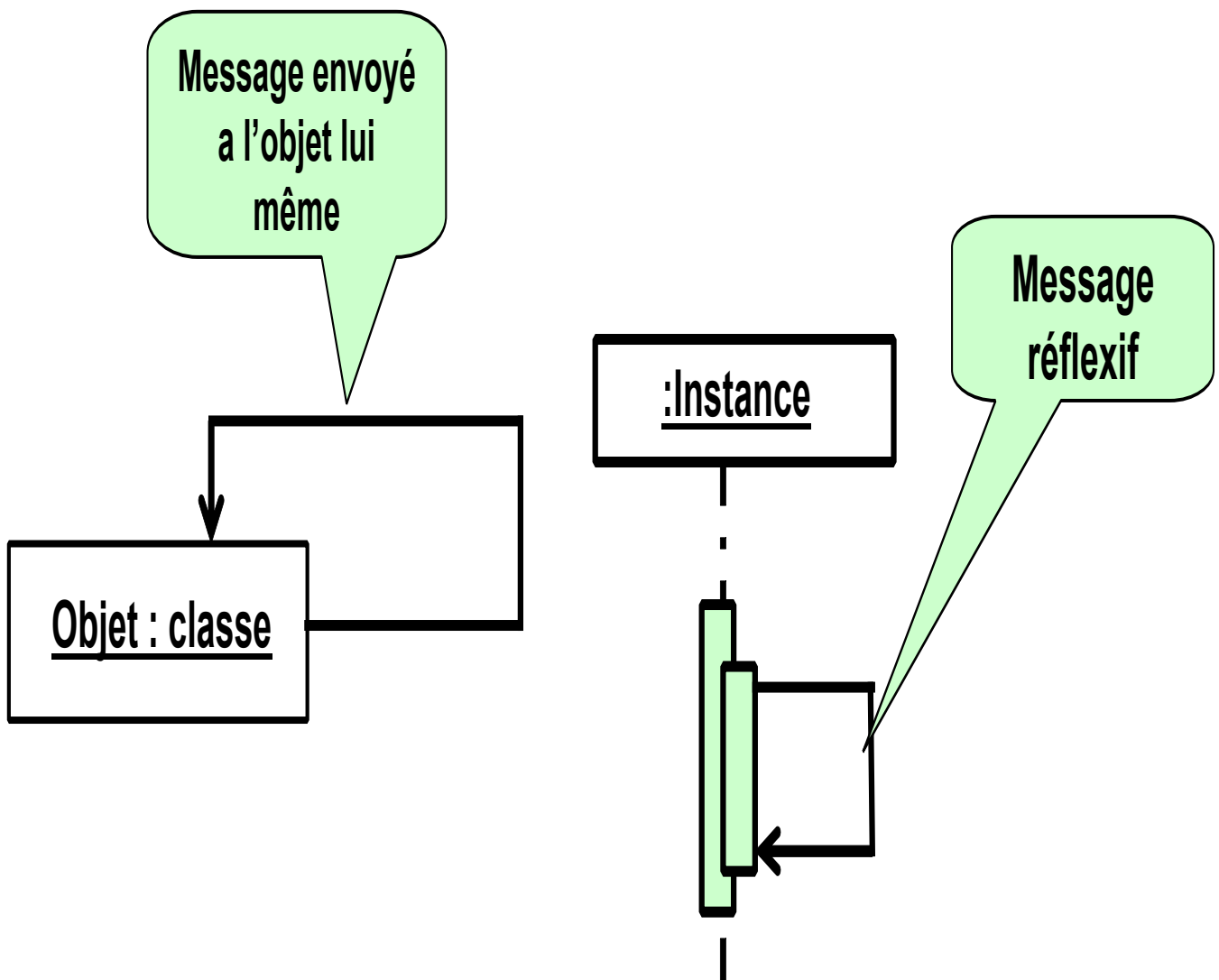
C'est un message qui n'interrompt pas l'exécution de l'expéditeur.
Le message envoyé peut être pris en compte par le récepteur à tout moment ou ignoré.



Types de messages (4/5)

● Message réflexif:

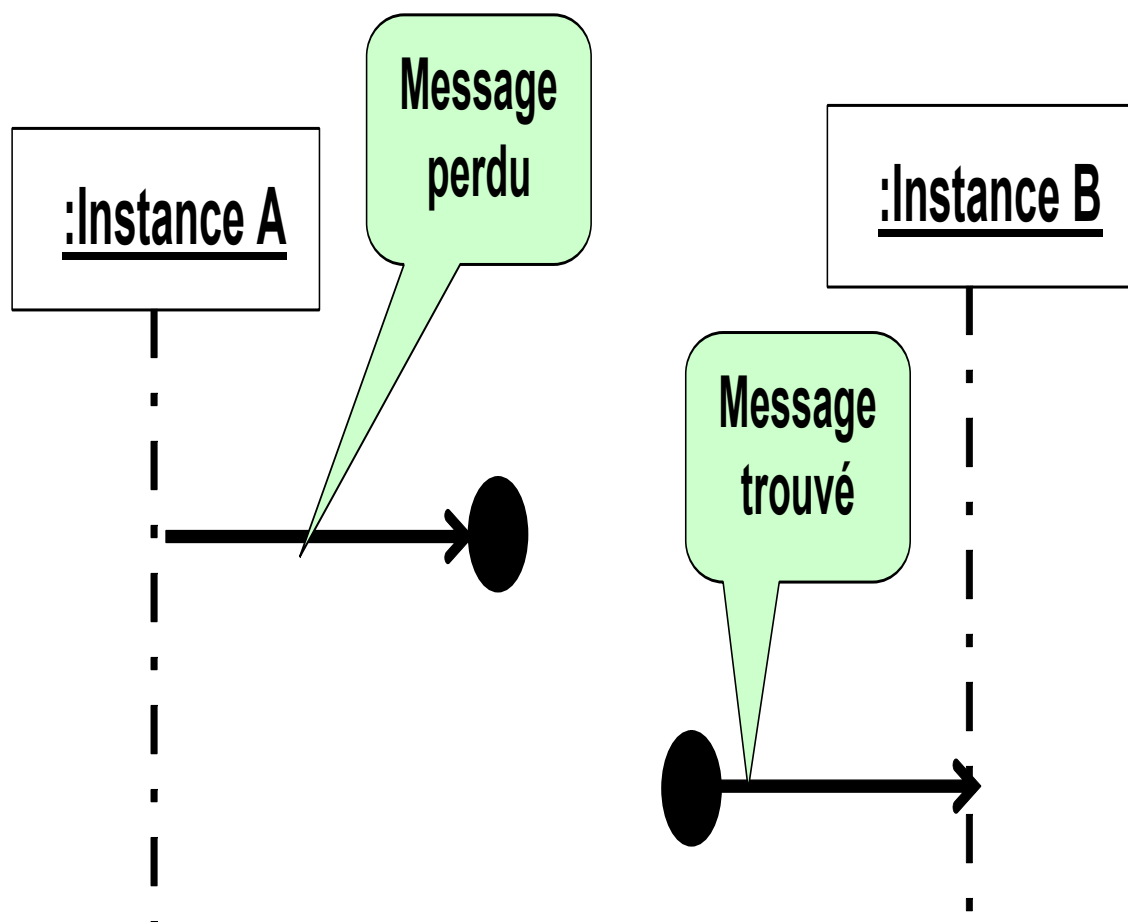
Il représente un message qu'envoie un objet à une instance de cet objet.



Types de messages (5/5)

● Messages perdus / trouvés

Ils permettent de représenter respectivement la perte d'un message (destinataire non contacté) et l'arrivée d'un message (expéditeur non identifié).

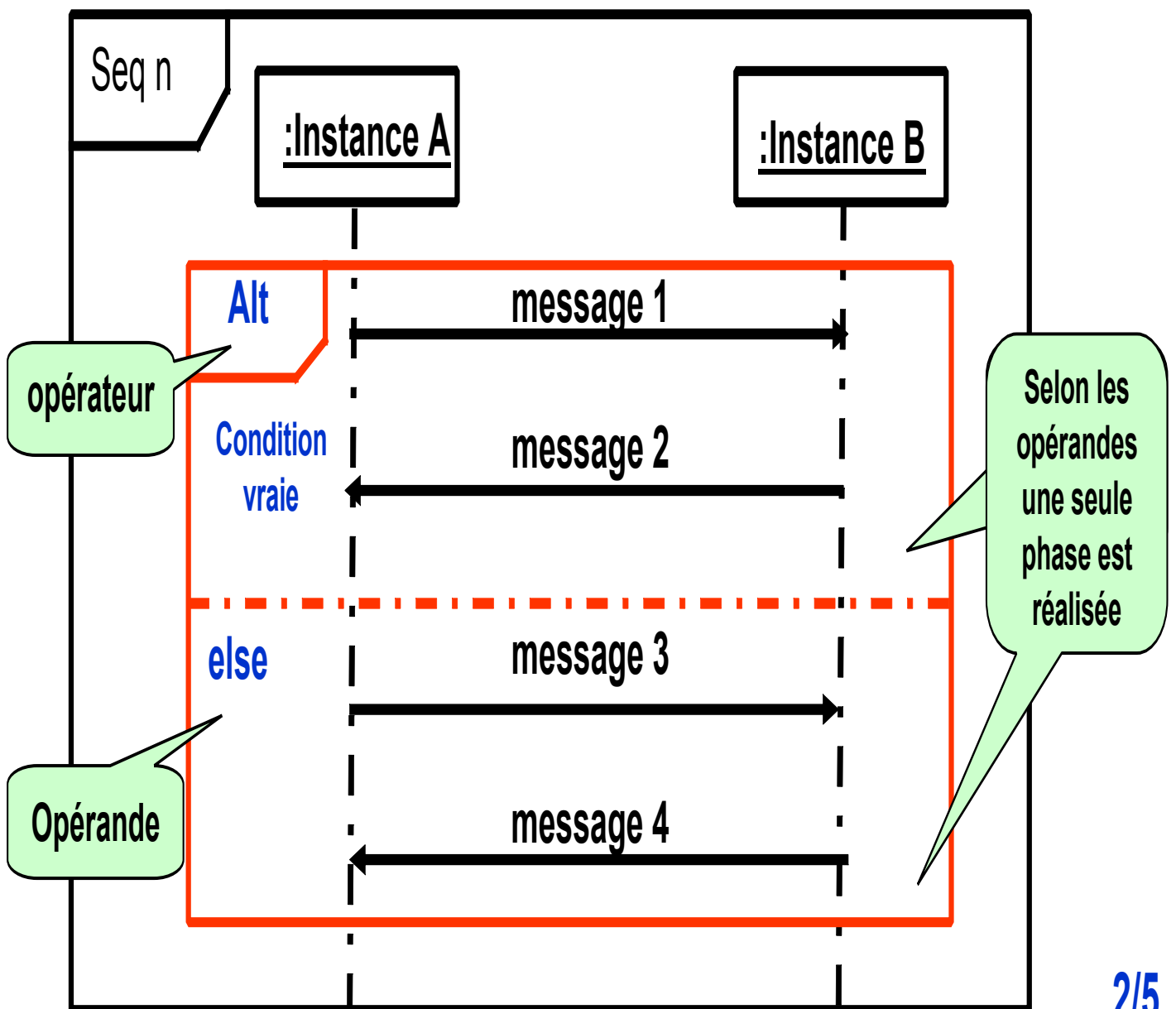


Les fragments combinés

- Un fragment combiné définit une expression de fragments d'interaction.
- L'utilisation des fragments combinés permet de décrire de manière concise plusieurs traces d'exécution d'actions.
- Un fragment combiné est défini par un **opérateur** d'interaction et les **opérandes** d'interactions correspondants
- Il est représenté dans un rectangle dont le coin supérieur gauche contient un pentagone où figure le type de la combinaison, appelé ***opérateur d'interaction***.

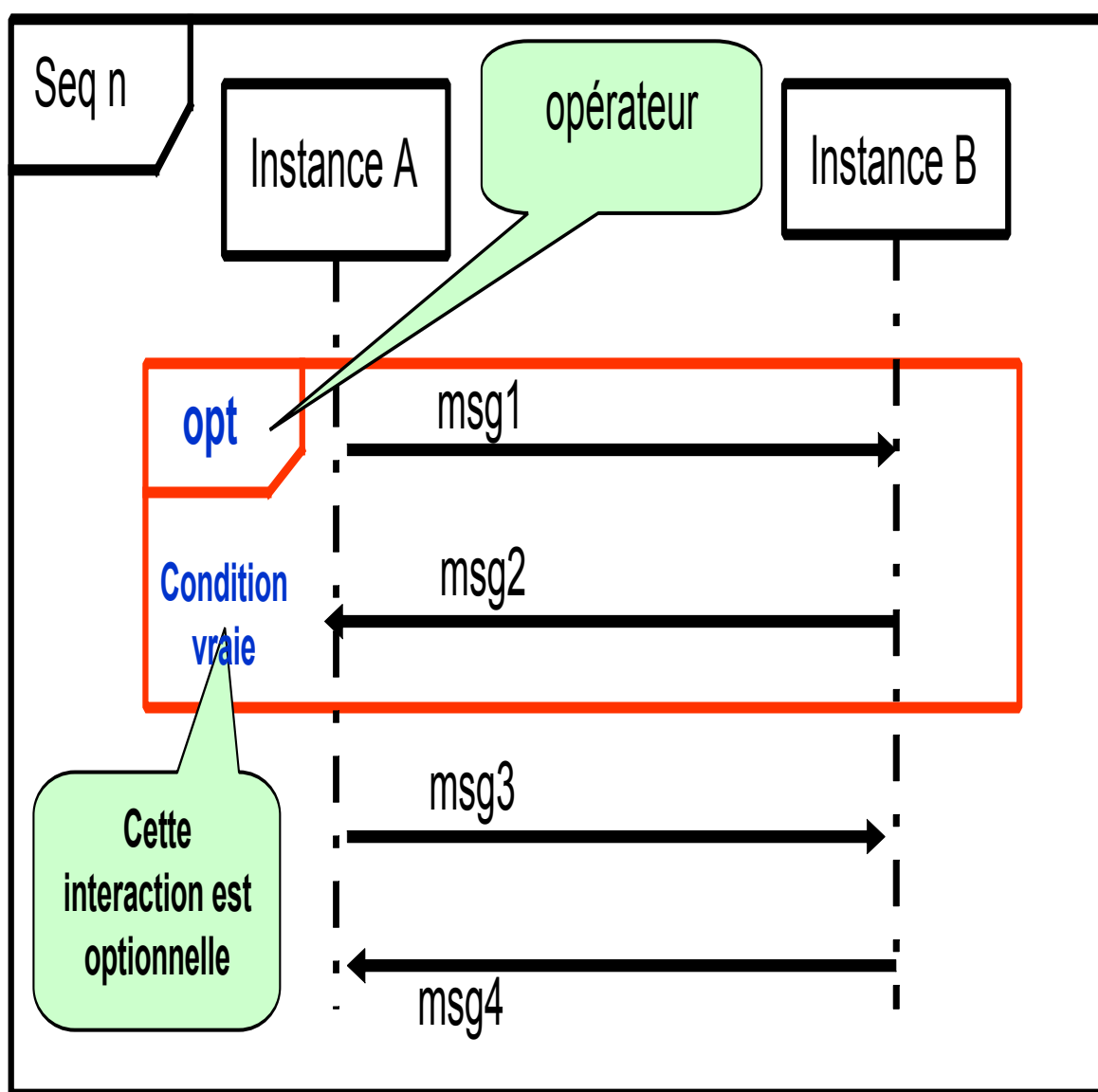
Opérateur « alt »

- L'opérateur "alt" désigne un choix, une alternative.
- C'est l'équivalent du **SI...ALORS...SINON**.



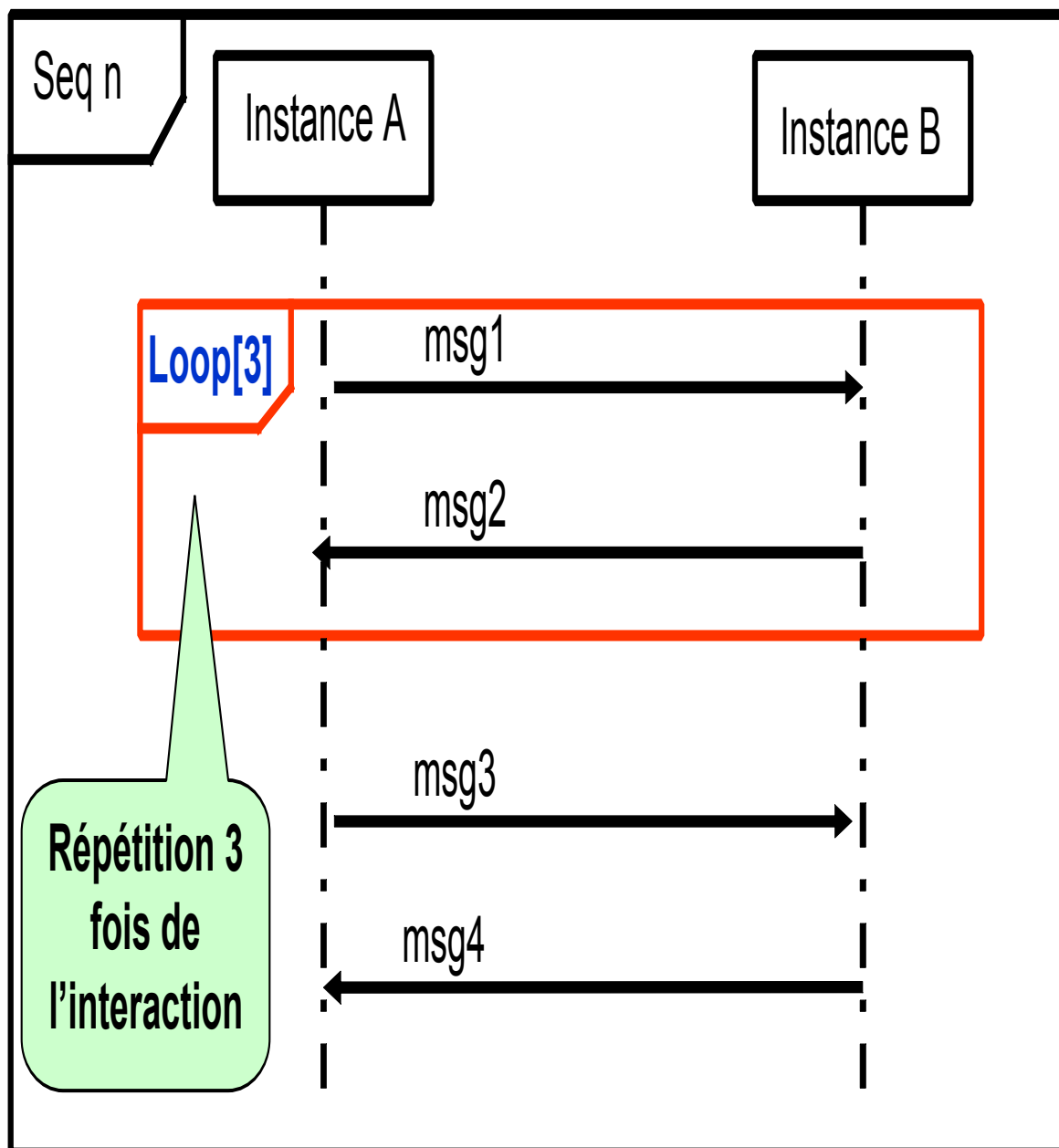
Opérateur « opt »

- Il désigne un fragment combiné optionnel.
- Un fragment optionnel est équivalent à un fragment "alt" qui ne posséderait pas d'opérande else.



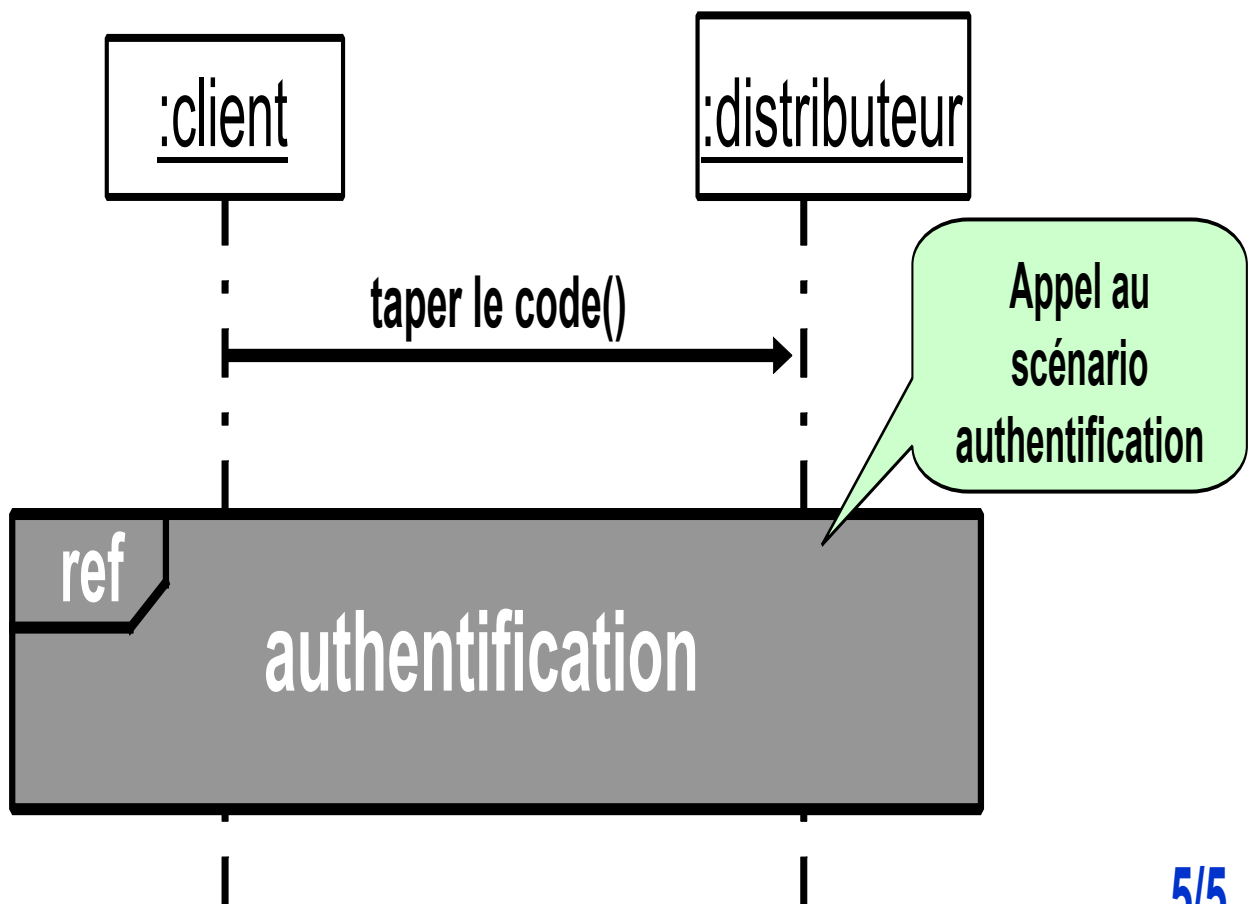
Opérateur « loop »

- C'est un opérateur utilisé pour décrire un ensemble d'interactions qui s'exécutent **en boucle**.



La référence

- C'est un raccourci ou renvoi vers un autre diagramme de séquence existant.
- Ce fragment permet de factoriser des parties de comportement utilisées dans plusieurs scénarios.



Les étiquettes des messages (1/3)

- Les étiquettes décrivent les messages auxquels elles sont attachées.
- La syntaxe d'une étiquette de message est la suivante :

`[[pré "/"] [[cond] [sérq] ["*"|"||"|"iter"] :] [r :=] msg ([par])`

- **pré** : liste de numéros de séquence de messages qui vont précéder l'envoi du message.
- **cond** : **garde**, expression booléenne. Permet de conditionner l'envoi du message, à l'aide d'une clause exprimée en langage naturel.

Les étiquettes des messages (2/3)

- **séq**: numéro de séquence du message. Indique le rang du message
- **iter** : récurrence du message. Permet de spécifier en langage naturel l'envoi séquentiel de messages.
|| indique l'envoi parallèle de ces messages.
- **r** : valeur de retour du message.
- **msg** : nom du message.
- **par** : paramètres (optionnels) du message.

Exercice d'application (3/3)

● 5 : tapercode()

Ce message sans paramètre (de nom tapercode) a pour numéro de séquence 5

● [heure = 17h] 1 : quitteralledecours()

Ce message (ayant 1 comme numéro de séquence) n'est envoyé que s'il est 17h.

● 2 / || [i := 1..3] : remercier()

Représente l'envoi en parallèle de 3 messages qui ne seront envoyés qu'après l'envoi du message 2.

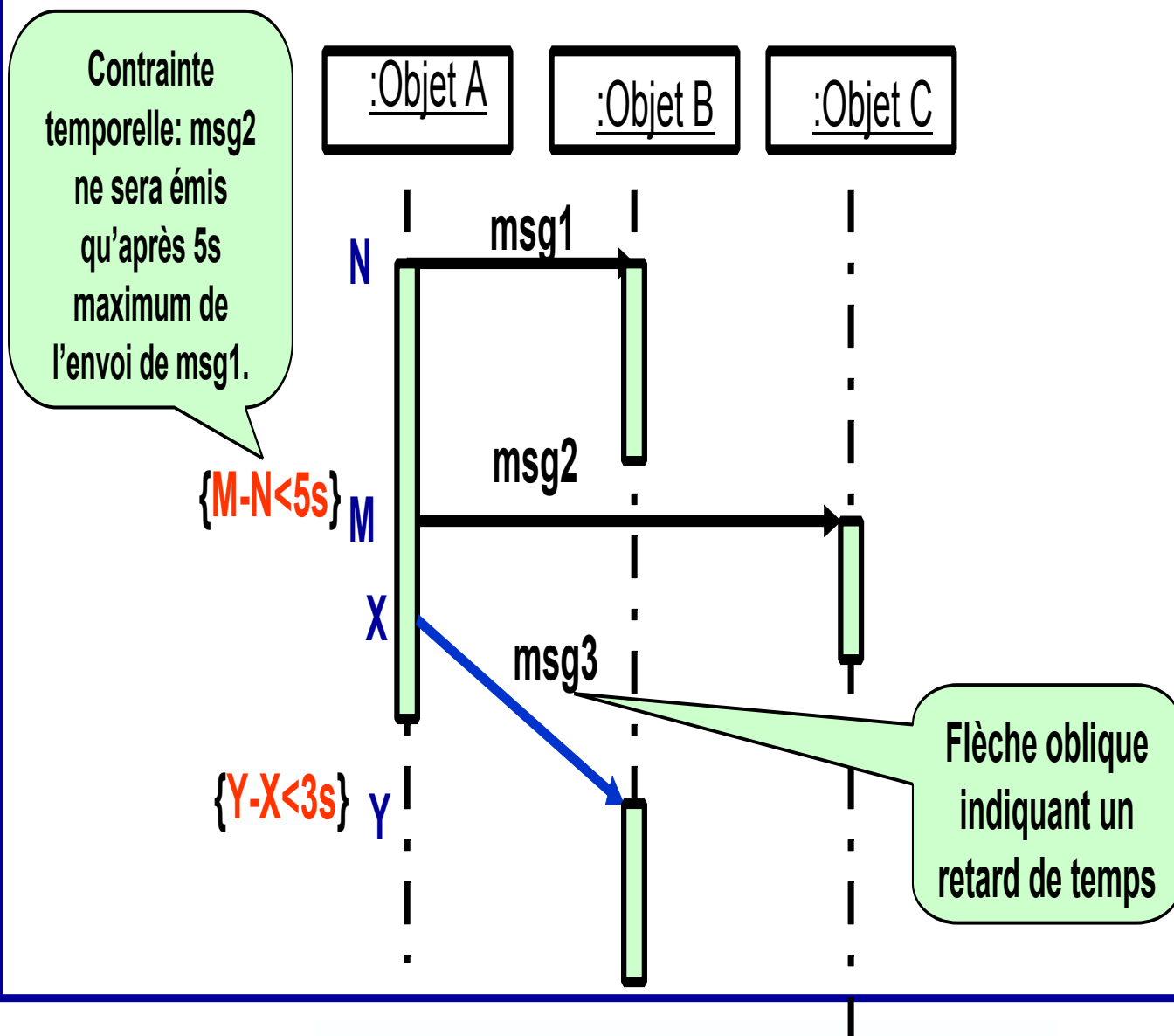
● 1.3 , 2.1 / [t < 10s] 2.5 : age := demanderAge()

Ce message (numéro 2.5) ne sera envoyé qu'après envoi des messages 1.3 et 2.1 et que si "t < 10s". Il aura age comme type de retour.

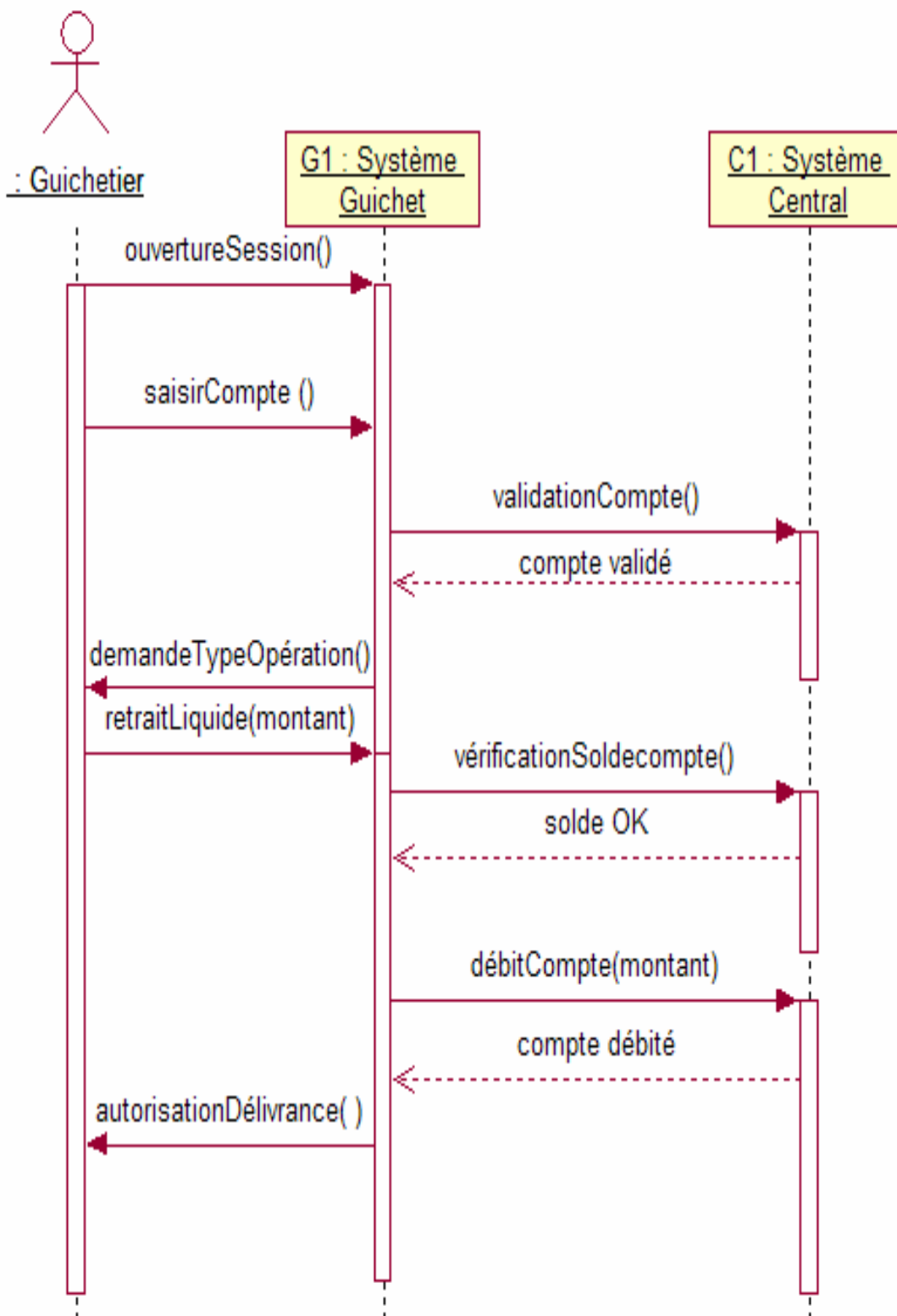
Contraintes temporelles

● Pour modéliser les délais de transmission non négligeables, on utilise les deux notations suivantes:

- une flèche oblique,
- ou des notations temporelles dans la marge.



Exemple complet



Exemple : appel téléphonique

