



UML

UE Analyse et modélisation
BINV-2160
B. Lehmann
2022-2023

Les diagrammes d'interactions

But des diagrammes d'interactions

- Décrire **l'évolution du système** dans une **utilisation particulière**
 - Évolution du système: comment les objets **interagissent** via des messages.
 - Utilisation particulière: présentation partielle.

Utilisation

- On décrit les **interactions** entre objets en utilisant les **opérations membres de la classe** de ces objets.
- Cependant, les diagrammes peuvent être utilisés avant la conception du diagramme de classes.
 - Exemple : on peut utiliser les diagrammes d'interaction pour **modéliser un scénario**.

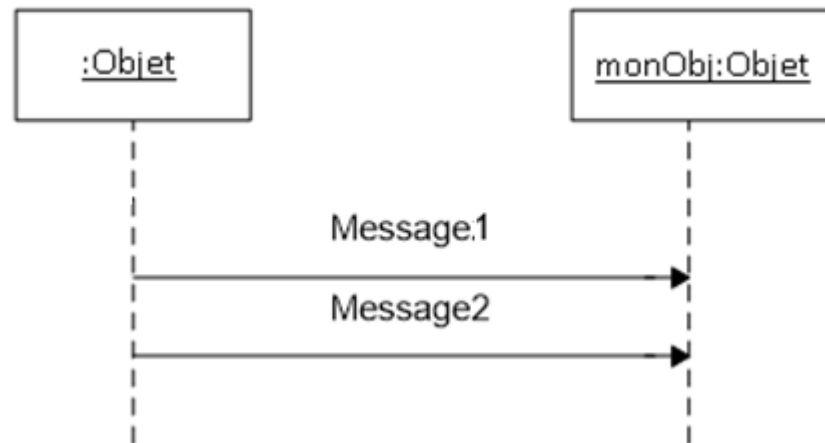
Le diagramme de séquences

Le diagramme de séquences

- Le diagramme de séquences permet de décrire l'évolution du système en présentant les **interactions entre objets de manière chronologique.**
- Il concerne les **objets** et non les classes : ils montrent les **messages** qui sont envoyés d'un objet à un autre.

Représentation graphique

- Axe vertical: temps (*de haut en bas*)
- Axe horizontal: objets concernés

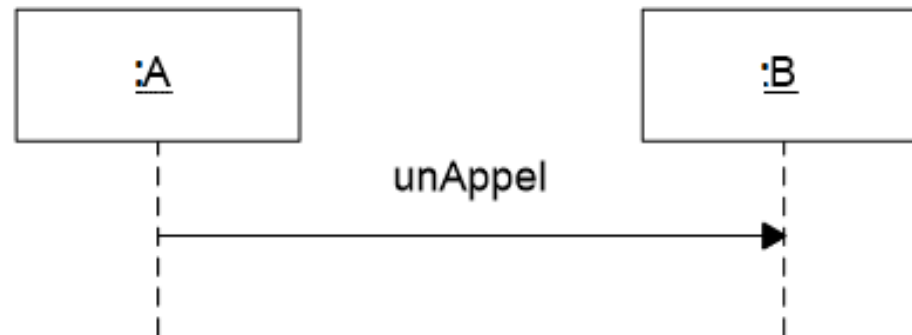


2 types de messages

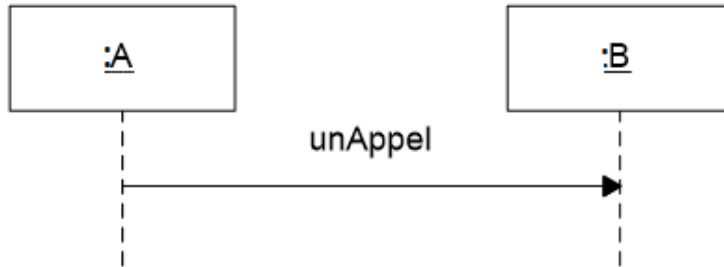
- **L'appel** : invocation synchrone d'une opération (avec un mécanisme pour rendre ensuite la main à l'émetteur)
 - En java, assimilé à un **appel de méthode**.
- **Le signal** : communication asynchrone explicite
 - En java, assimilé au **thread**.

L'appel (1)

- **Synchrone** -> réponse attendue du récepteur; l'émetteur perd le contrôle.
- Représentation:
- Exemple:



L'appel (2)



```
public class B {
    public void unAppel(){//...}
}

public class A {
    private B b ;
    //...
    public A() {
        b.unAppel();}
}
```

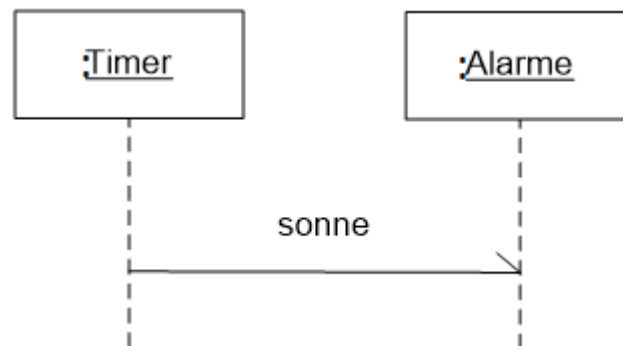


Le signal

- **Asynchrone** -> aucune réponse attendue; l'émetteur garde le contrôle.
- Représentation:
- Exemple:

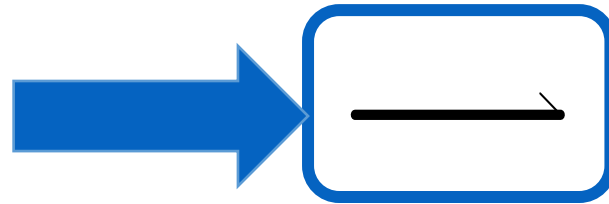


UML2

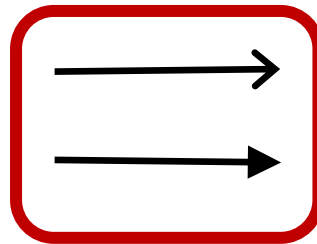


Les messages : représentation dans ce cours

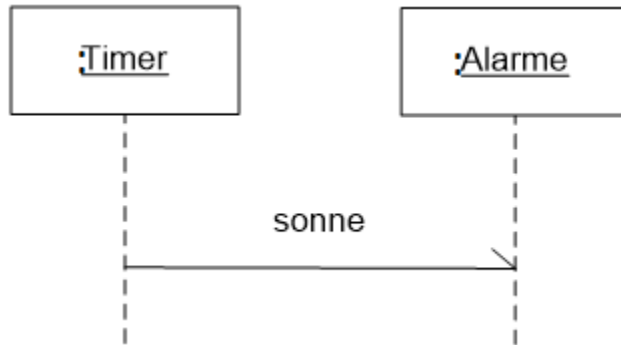
- Le **signal** sera toujours représenté avec une seule ailette :



- L'**appel** peut être indifféremment représenté par :



Le signal



```
public class Alarme {
    public void sonne(){
        // ...
    }
}
```

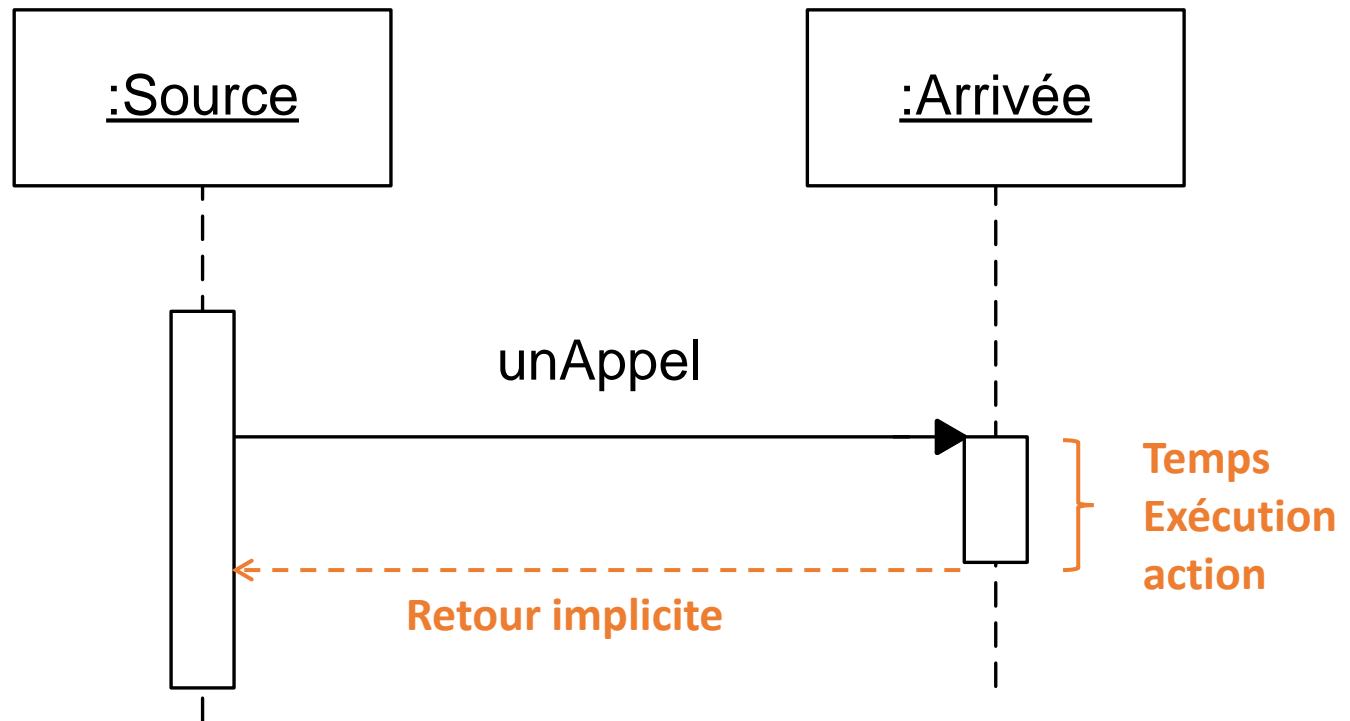


Il faut utiliser des threads pour modéliser en Java un signal asynchrone.

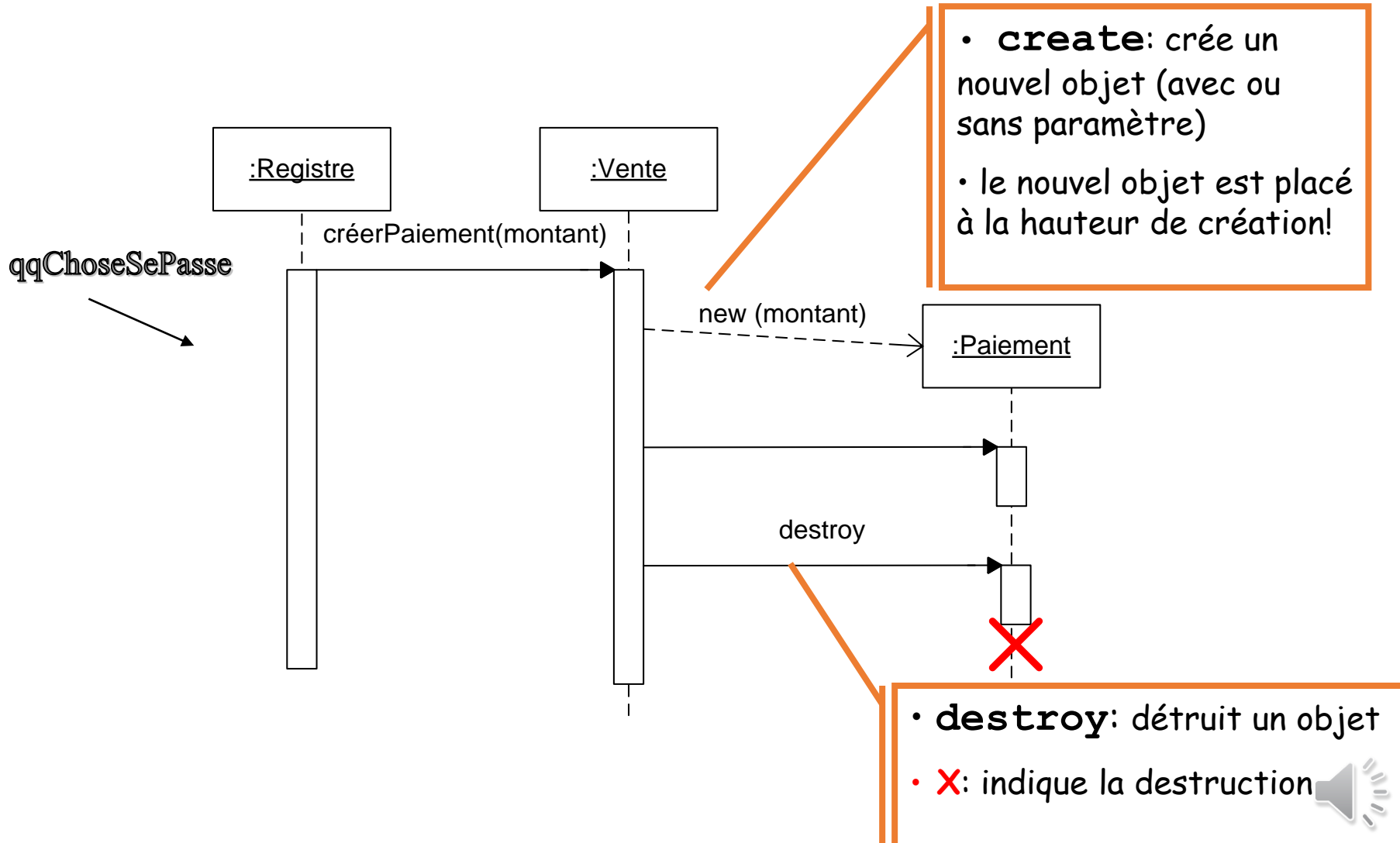
```
public void déclencher(){
    alarme.sonne();
}
```



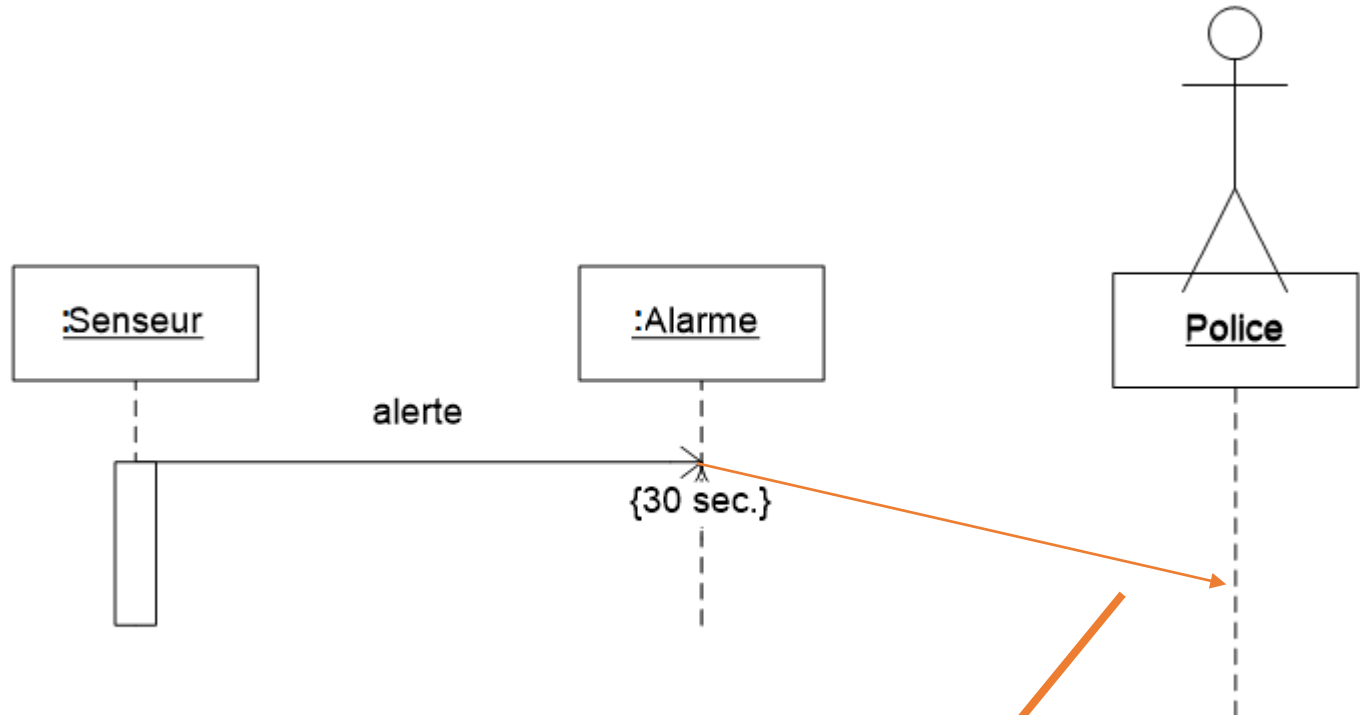
Barre d'activation



Création / *destruction*



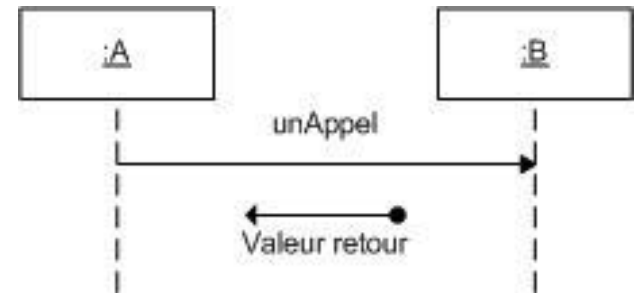
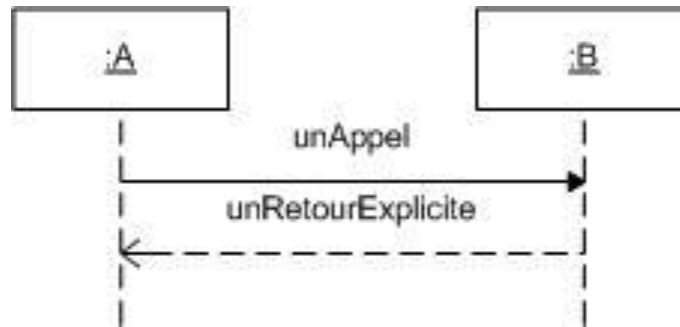
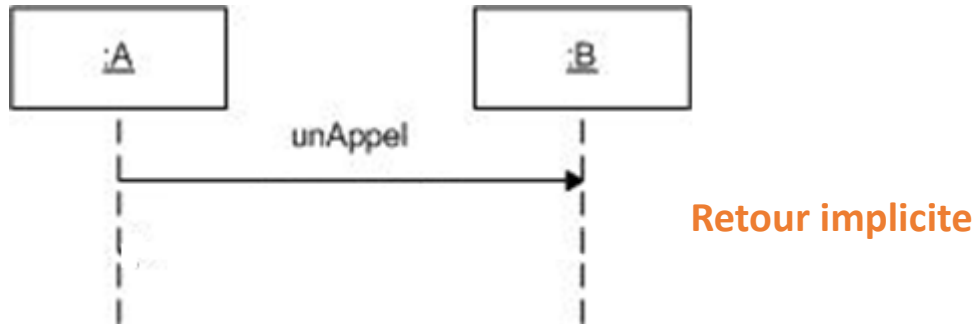
Contraintes de temps



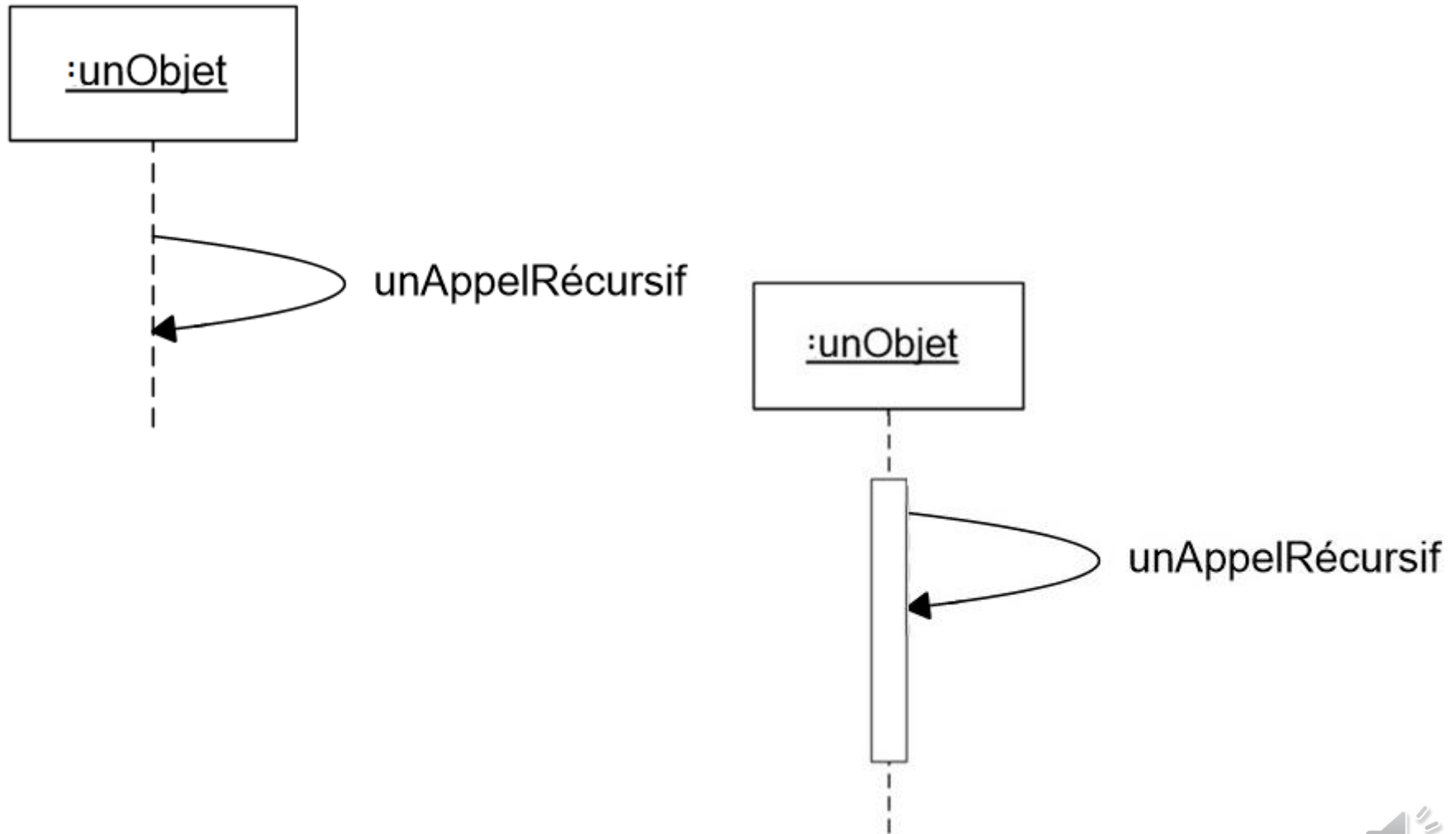
Tracer le message en oblique
permet d'insister sur sa durée



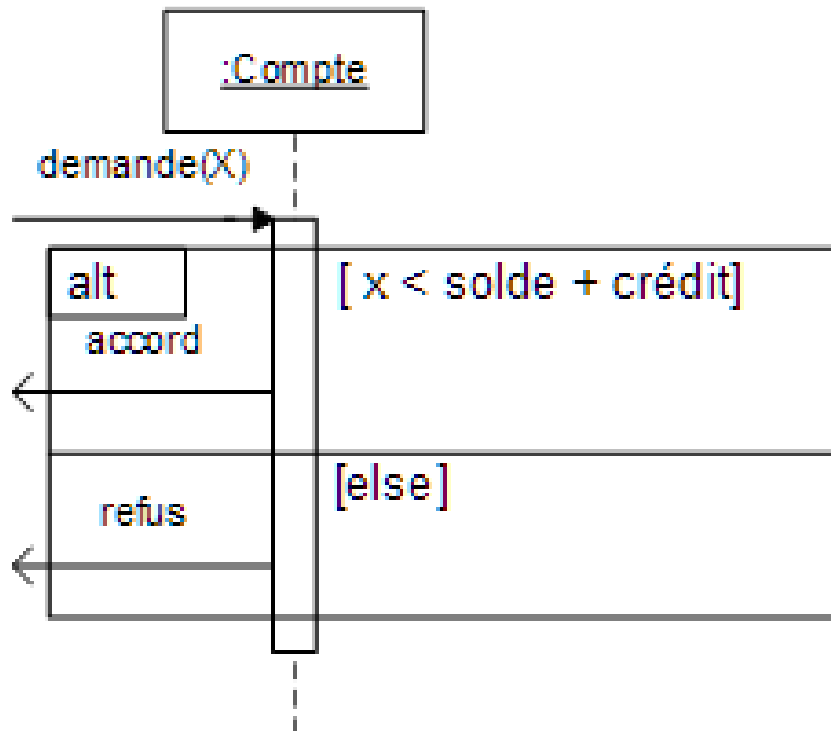
Retour



Appel récursif



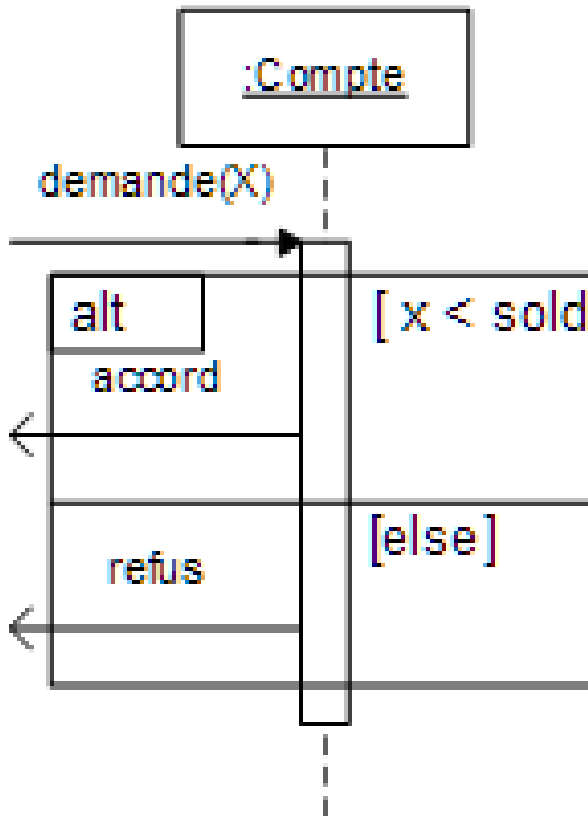
Alternative



- Si ... alors ...
Sinon ...



Alternative

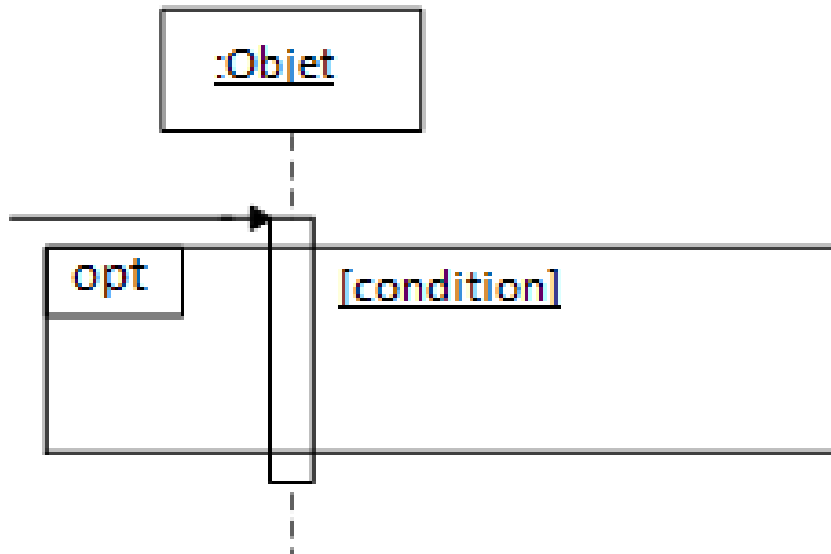


```
public class Compte {
    private int solde;
    private int crédit;
    //...
    public boolean demande(int montant){
        if (montant<solde+crédit)return true;
        return false;
    }
}
```

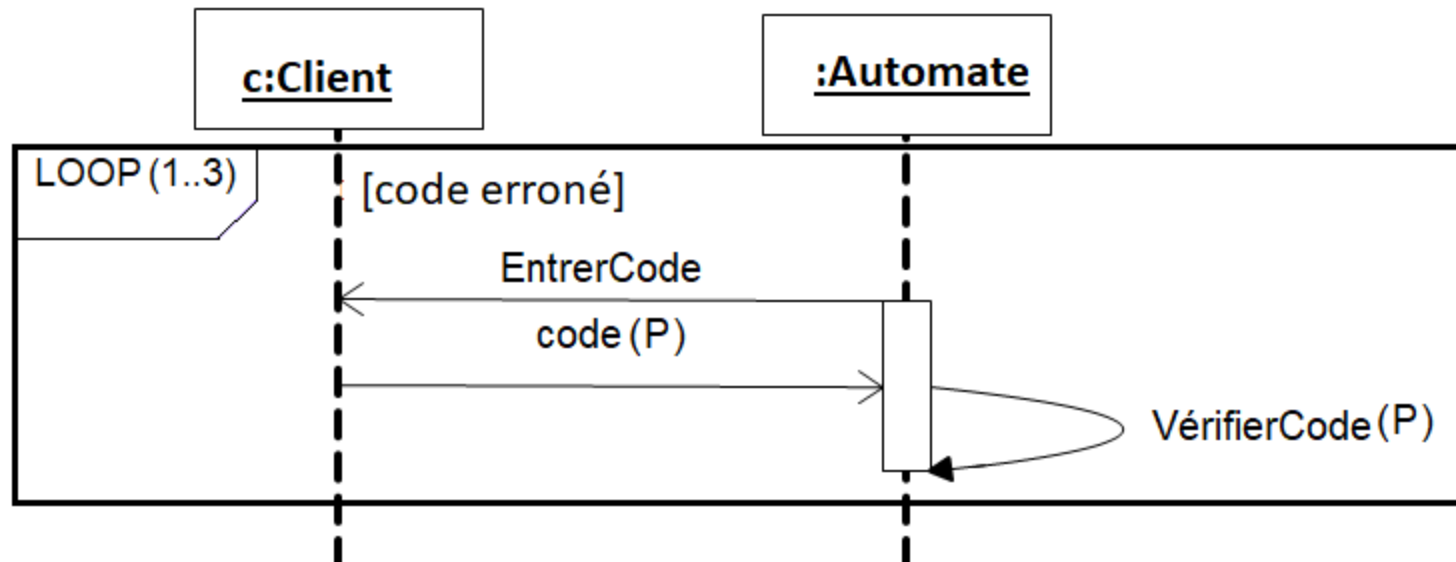


Option

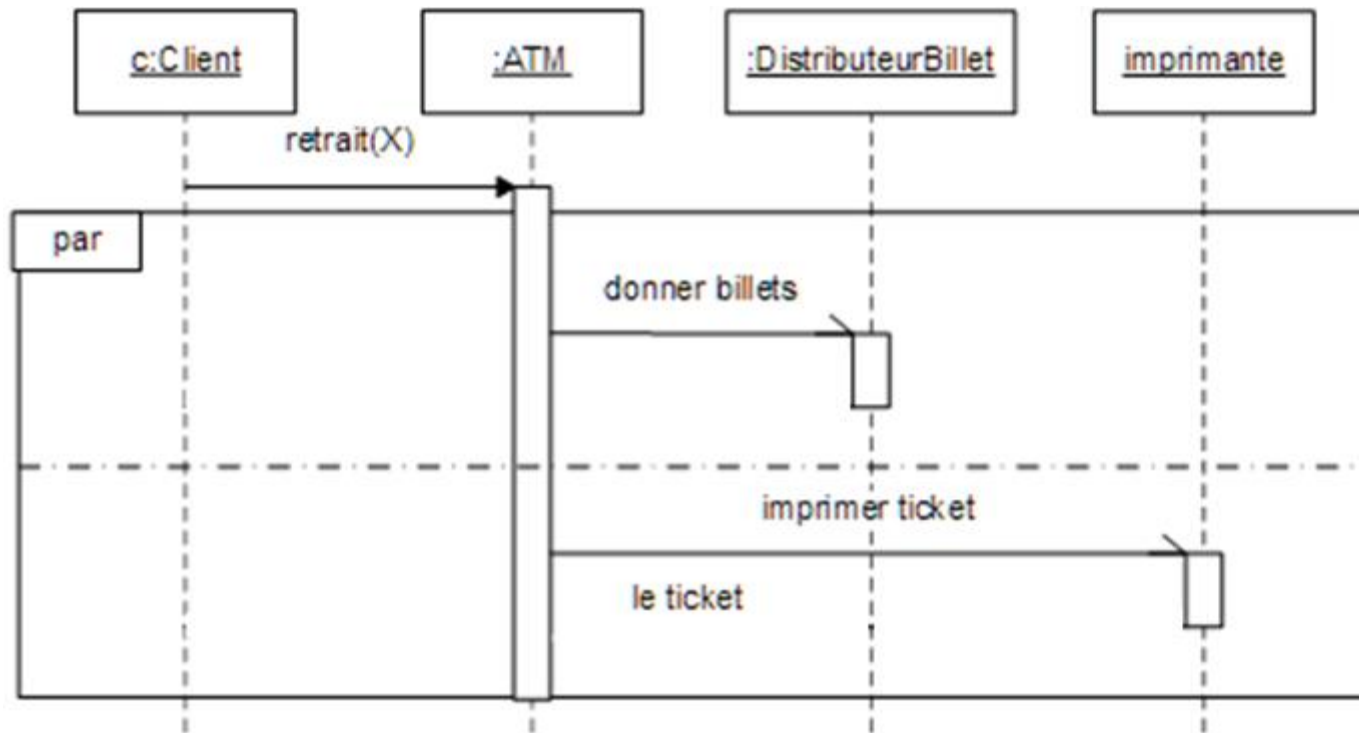
- Si ... alors ...



Itération



Parallélisme



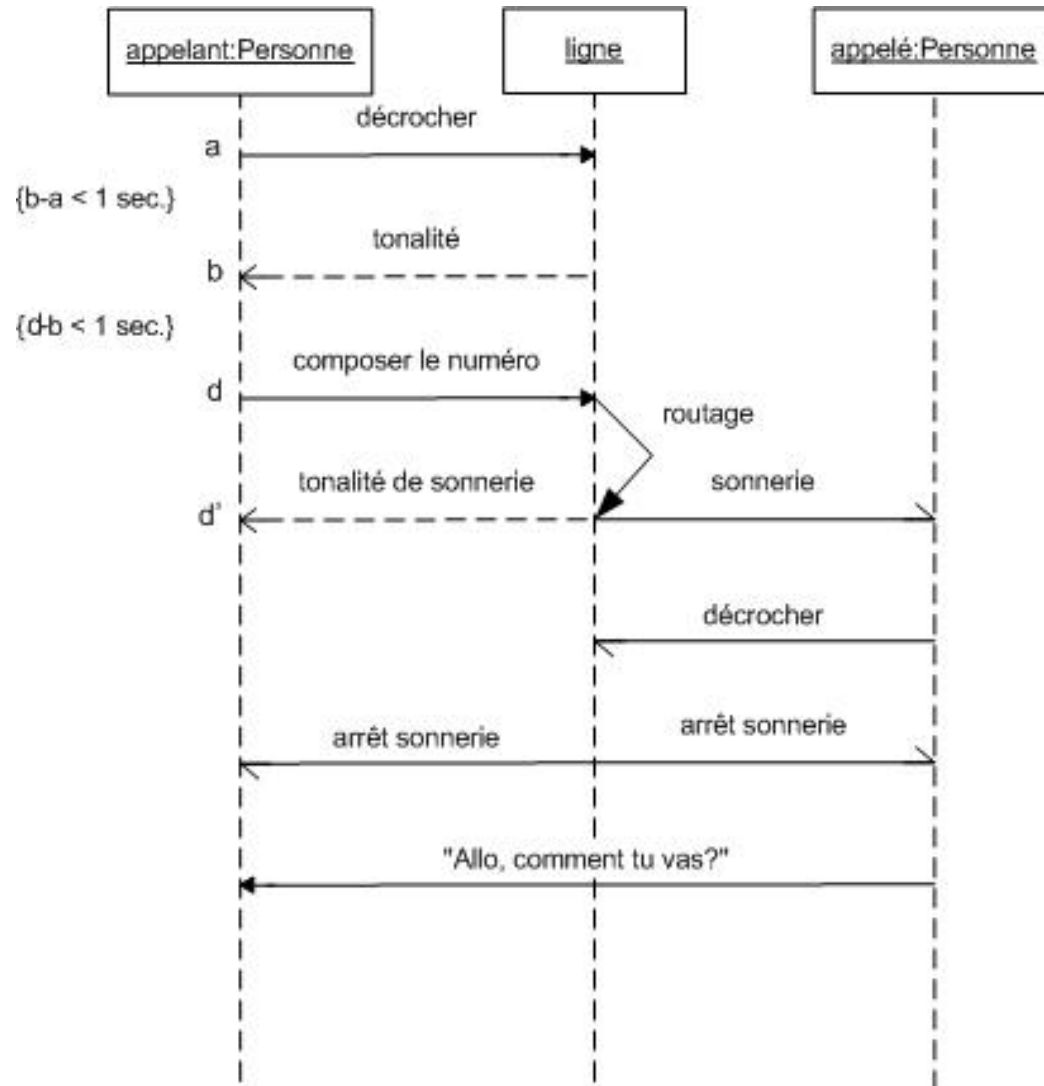
→ L'ordre dans lequel les deux opérations parallèles sont effectuées importe peu.



La marge de gauche

Exemple :
Indication de
timing
Entre { }

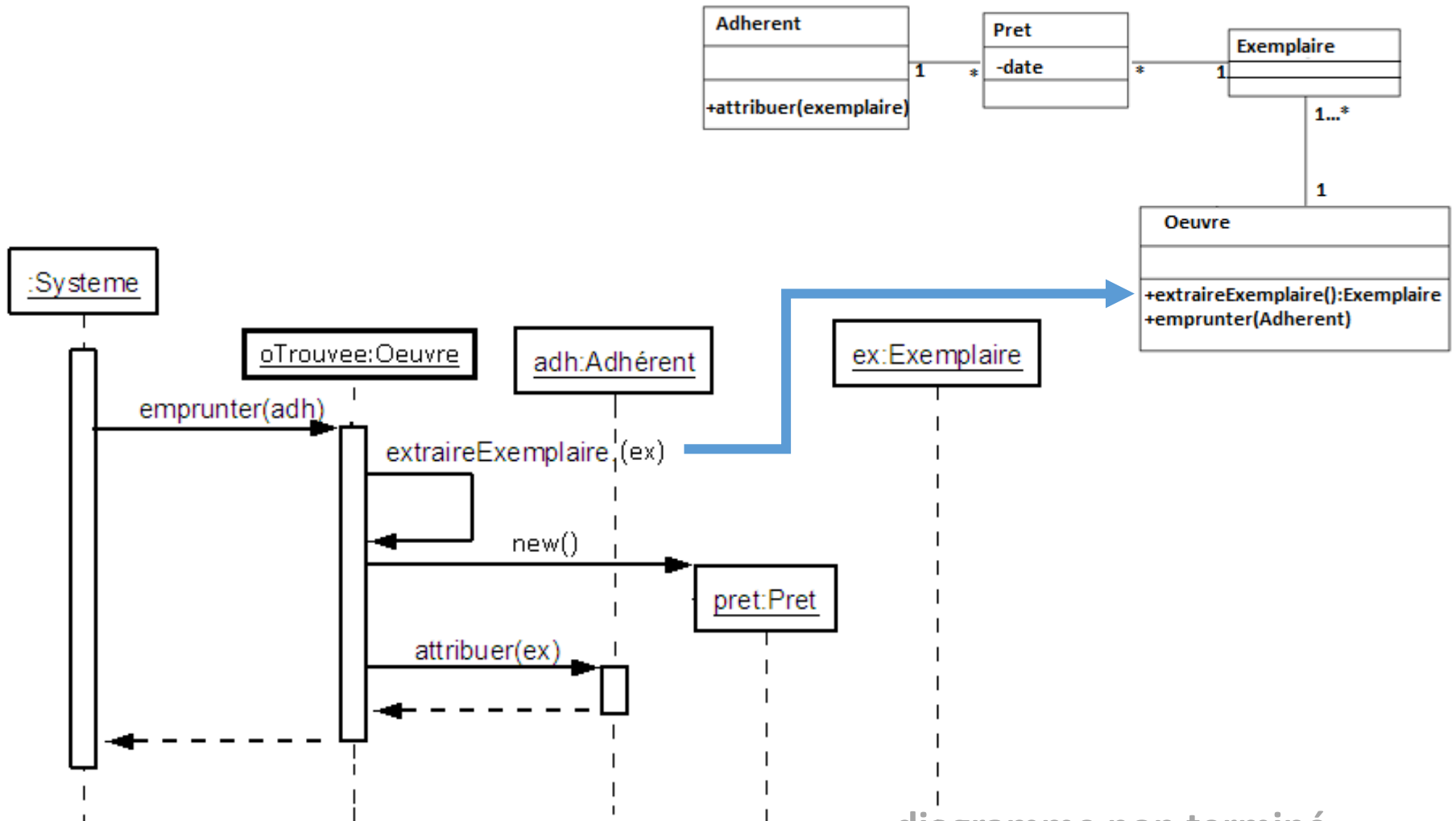
Comme dans
tous les
diagrammes, on
peut ajouter des
commentaires et
des contraintes
OCL (Object
Constraint Language)



Représentation du système

- 2 manières de représenter le système:
 - De façon détaillée :
 - Les différents objets composant le système sont détaillés.
 - Comme une boîte noire:
 - Le système n'est pas décomposé en objets mais est vu comme un tout unique.

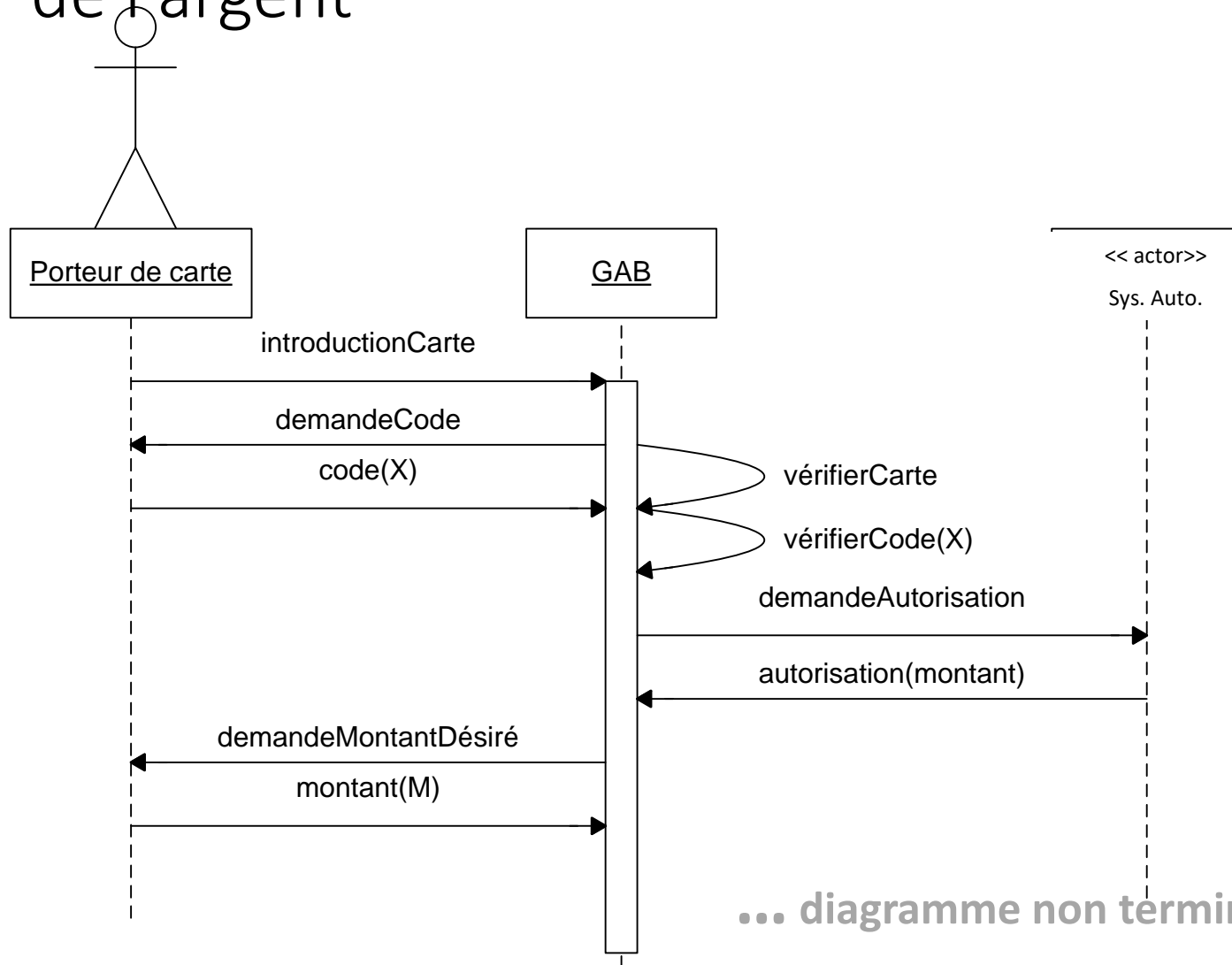
Exemple système détaillé



... diagramme non terminé



Exemple système en boîte noire – retirer de l'argent

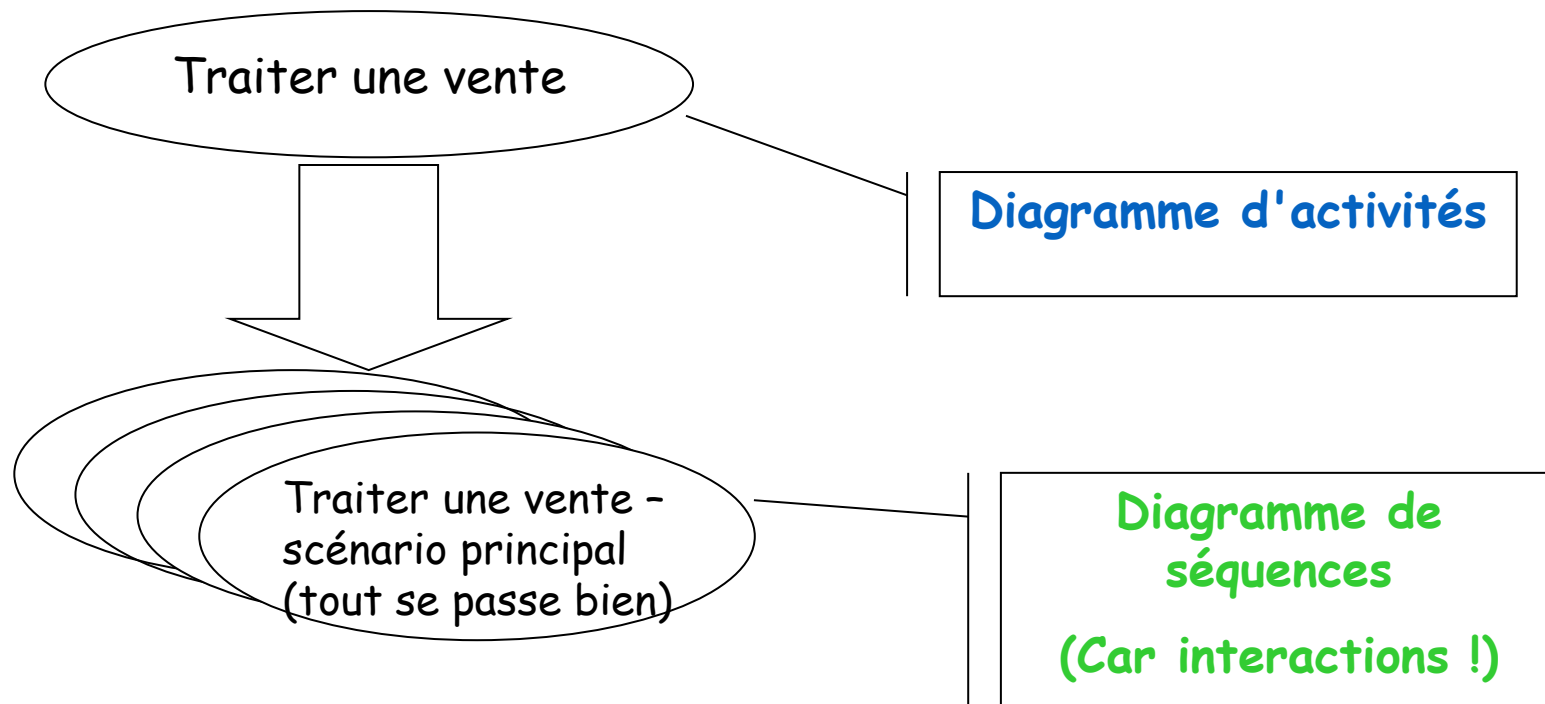


Activités ou séquences?

- Diagramme d'activités:
 - Décrire **l'évolution de toutes les utilisations** d'un système .
 - Pour un cas d'utilisations (ou plusieurs).
- Diagramme de séquences:
 - Décrire **l'évolution du système dans une utilisation particulière.**
 - Pour chaque scénario ou partie de scénario particulier d'un cas d'utilisation.



Activités ou séquences? exemple



Questions sur le
diagramme de
séquences ?

