

UML

Diagrammes de séquence

Delphine Longuet
delphine.longuet@lri.fr

Diagrammes de séquence (conception)

: Représenter les communications avec et au sein du logiciel

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

En conception : Décrire la **réalisation des cas d'utilisation** sur le système représenté par le diagramme de classes

- Point de vue **global** sur le fonctionnement du système
- Description au niveau de **l'état** (état du système à un instant)
- Description de **cas d'utilisation** particuliers
- Représentation des **échanges de messages**
 - entre les acteurs et le système, entre les objets du système
 - de façon chronologique

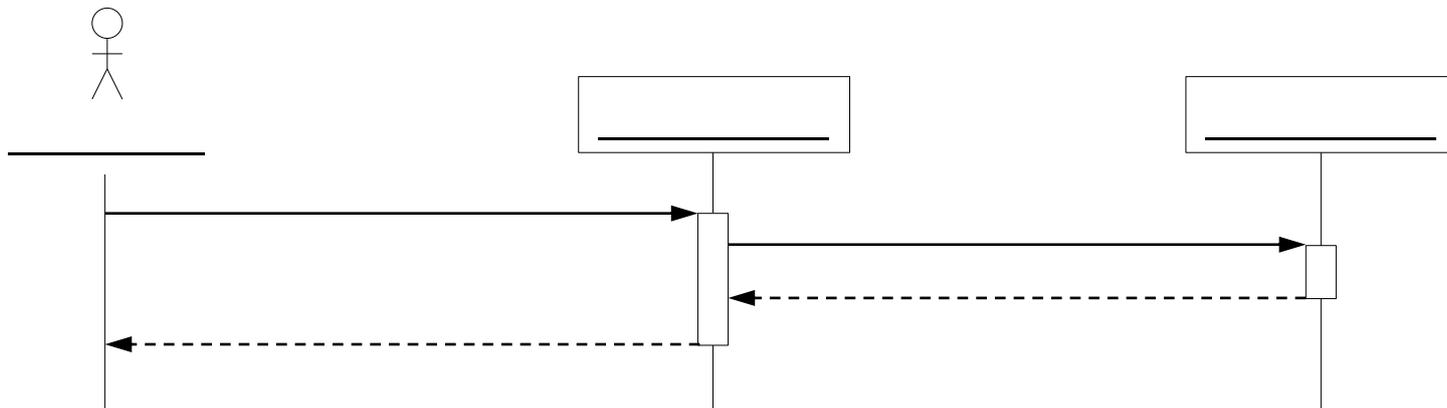
Diagrammes de séquence (conception)

Éléments du diagramme de séquence

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

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

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



Utilisation en phase de conception

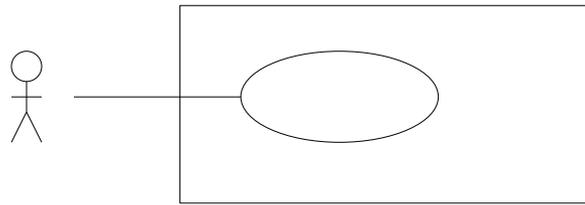


Diagramme de cas d'utilisation

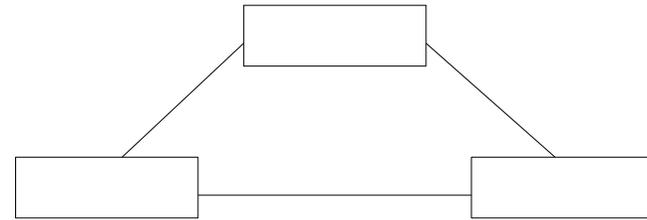


Diagramme de classes du système

: Description de la réalisation d'un cas d'utilisation système décrit par le diagramme de classes

Communication entre les acteurs et le système vu comme un ensemble d'objets

Utilisation en phase de conception

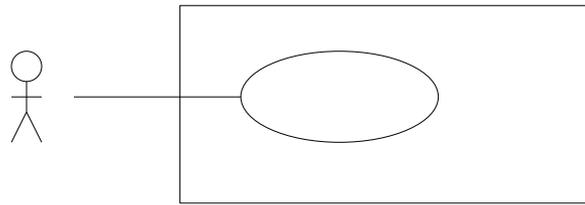


Diagramme de cas d'utilisation

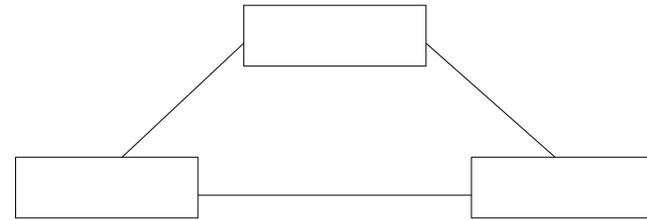
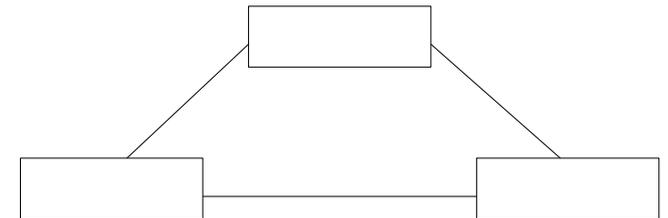


Diagramme de classes du système



Communication entre acteurs et système via une
physique...)

(texte, web,

Utilisation en phase de conception

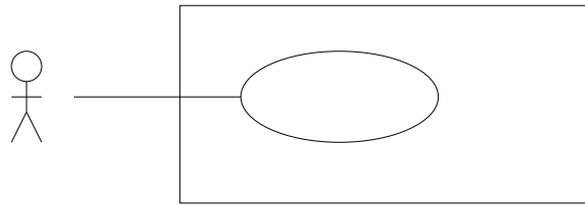


Diagramme de cas d'utilisation

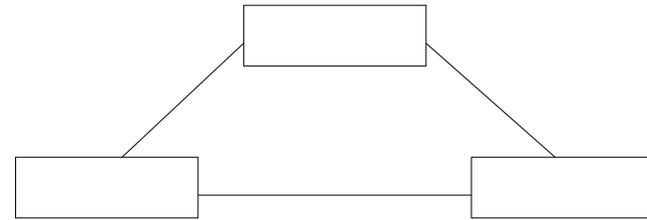
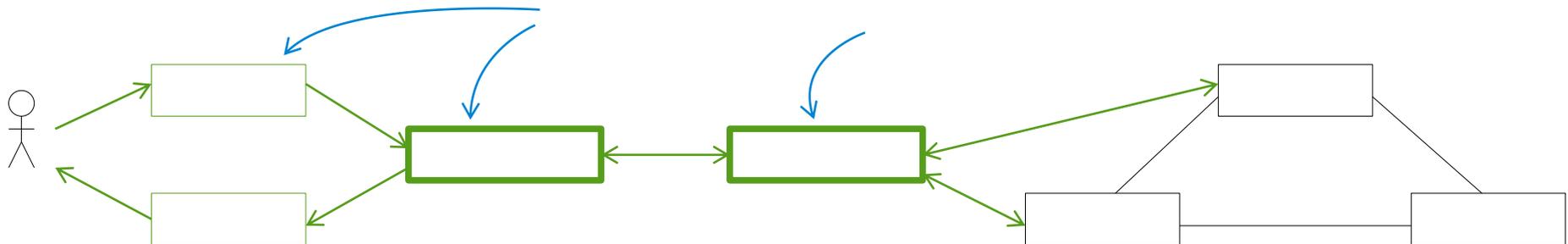


Diagramme de classes du système



: Création de classes de contrôle

classes d'interface

- gèrent les interactions avec les acteurs
- encapsulent le résultat des opérations

Utilisation en phase de conception

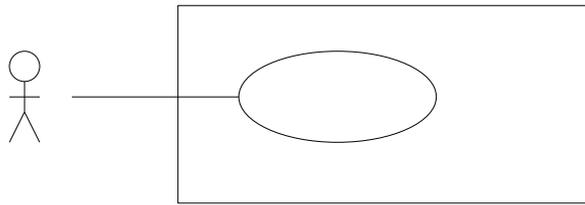


Diagramme de cas d'utilisation

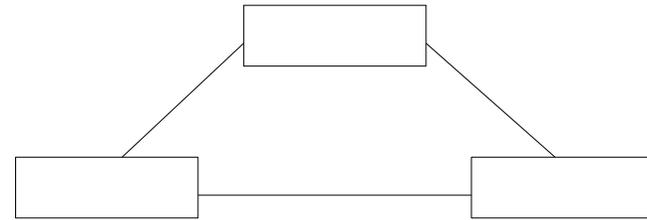


Diagramme de classes

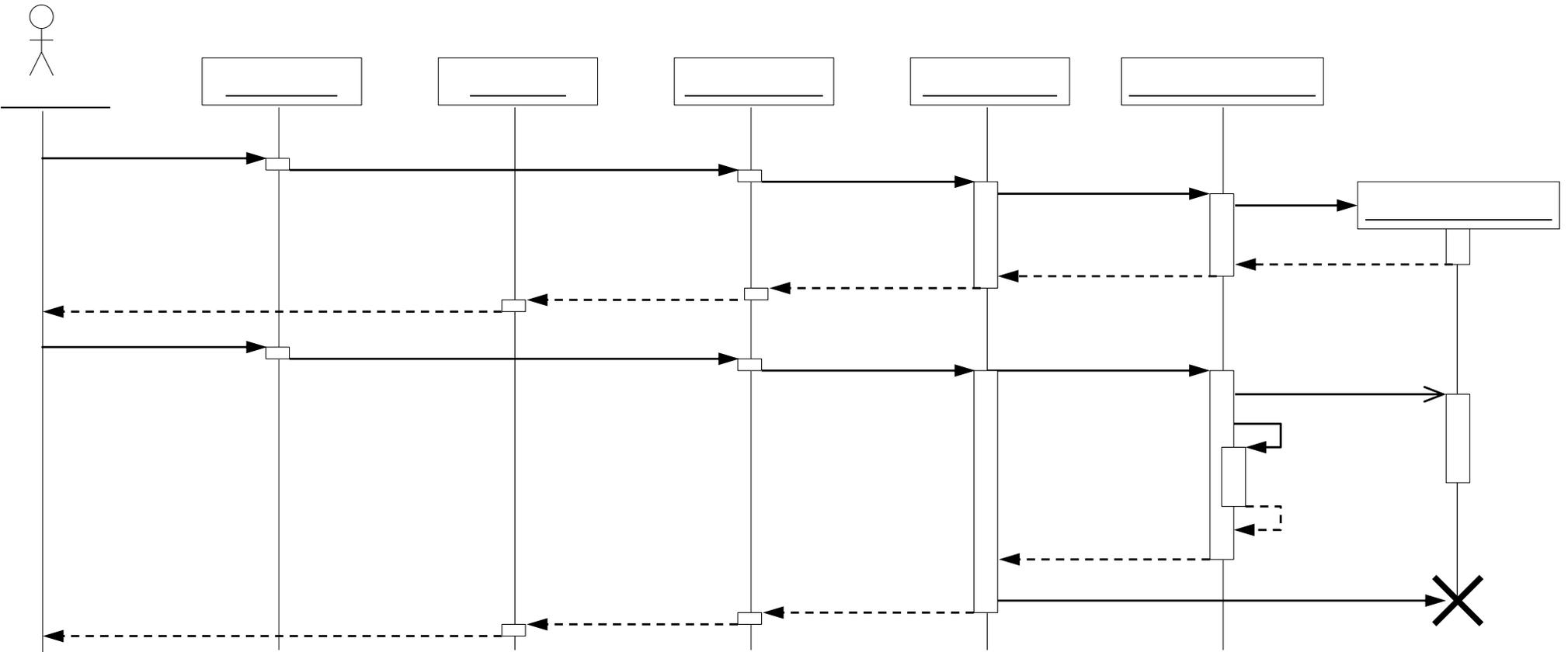
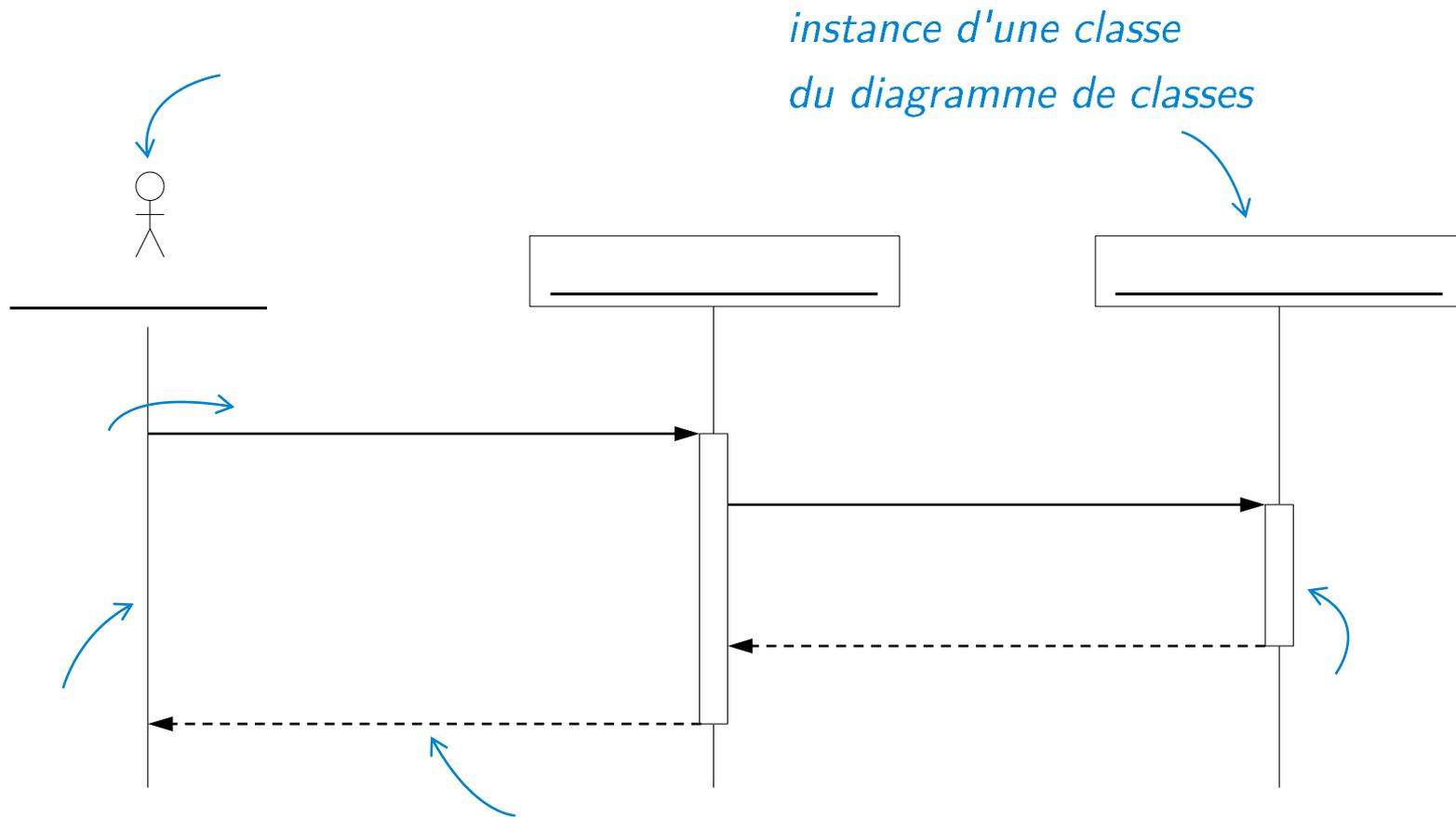


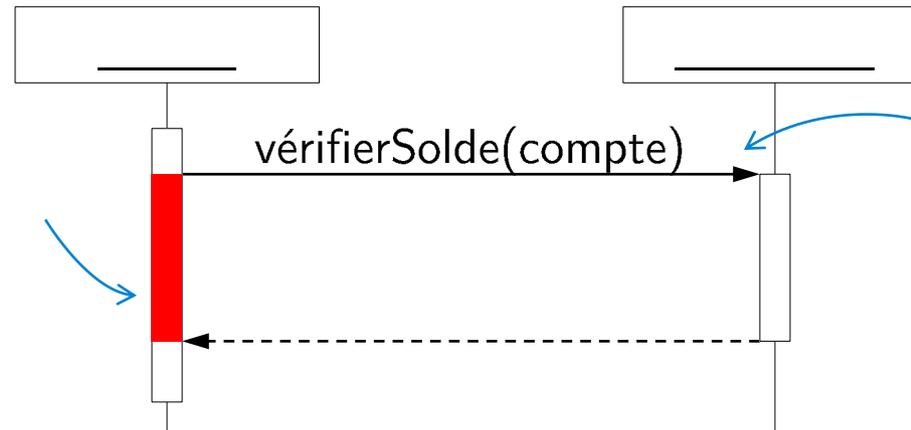
Diagramme de séquence du cas d'utilisation Utiliser

Éléments de base

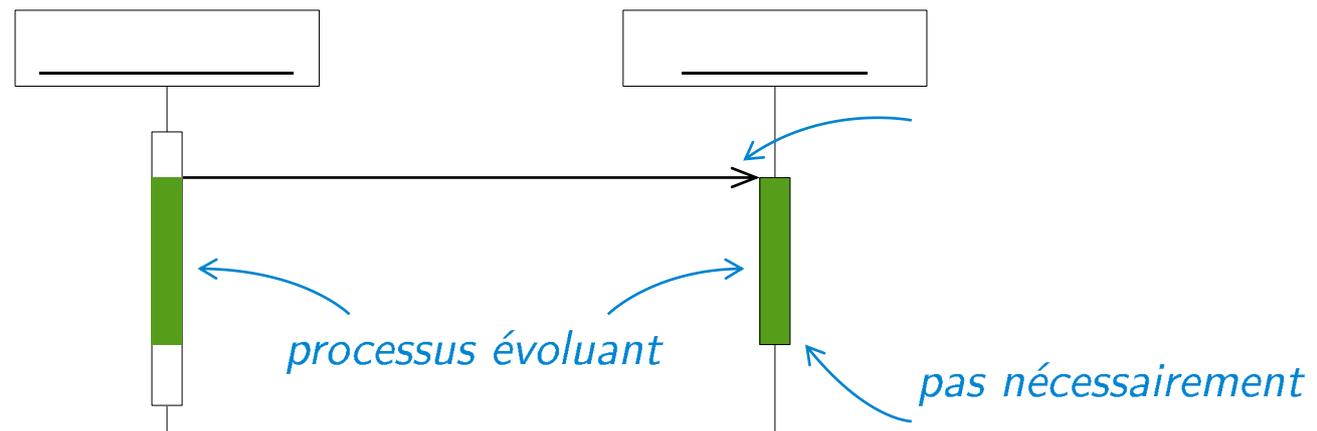


Types de messages

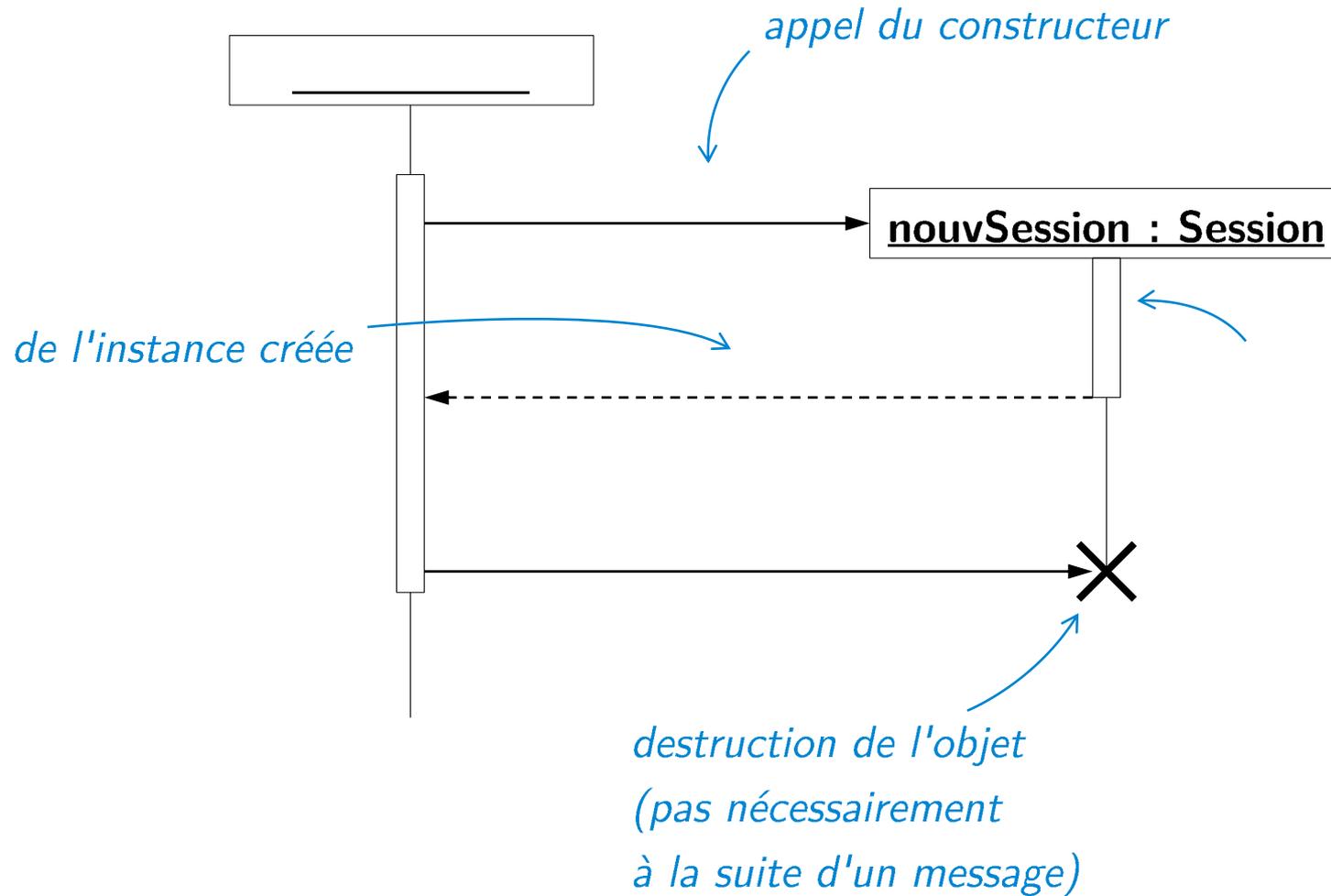
Message synchrone : Émetteur en attente du retour



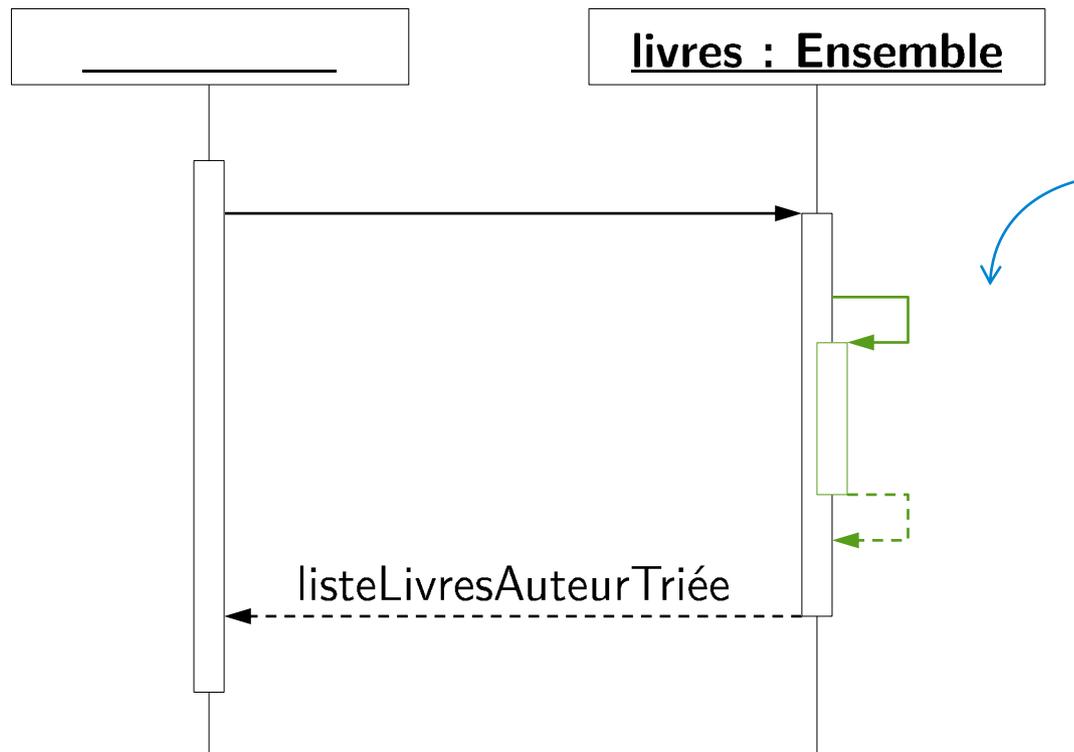
Message asynchrone : Émetteur non bloqué, continue son exécution



Création et destruction d'objet



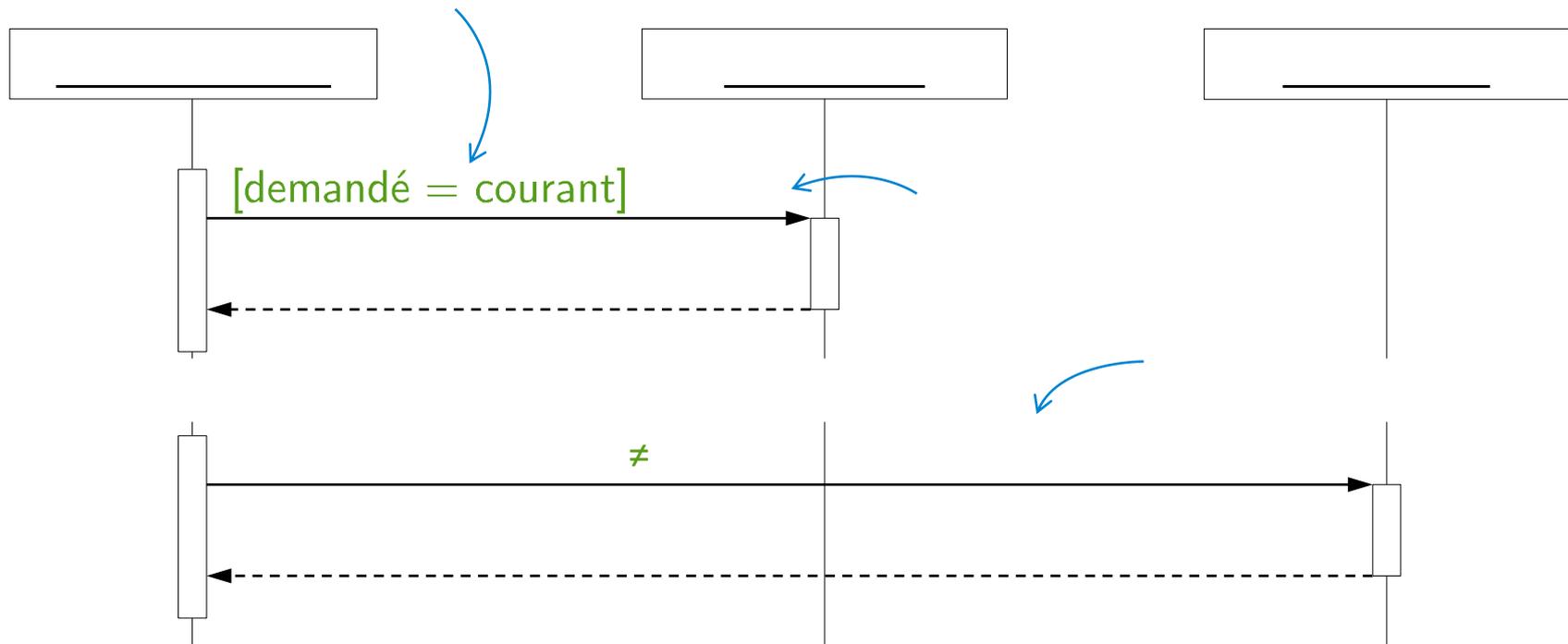
Message réflexif



Alternative

à l'envoi d'un message

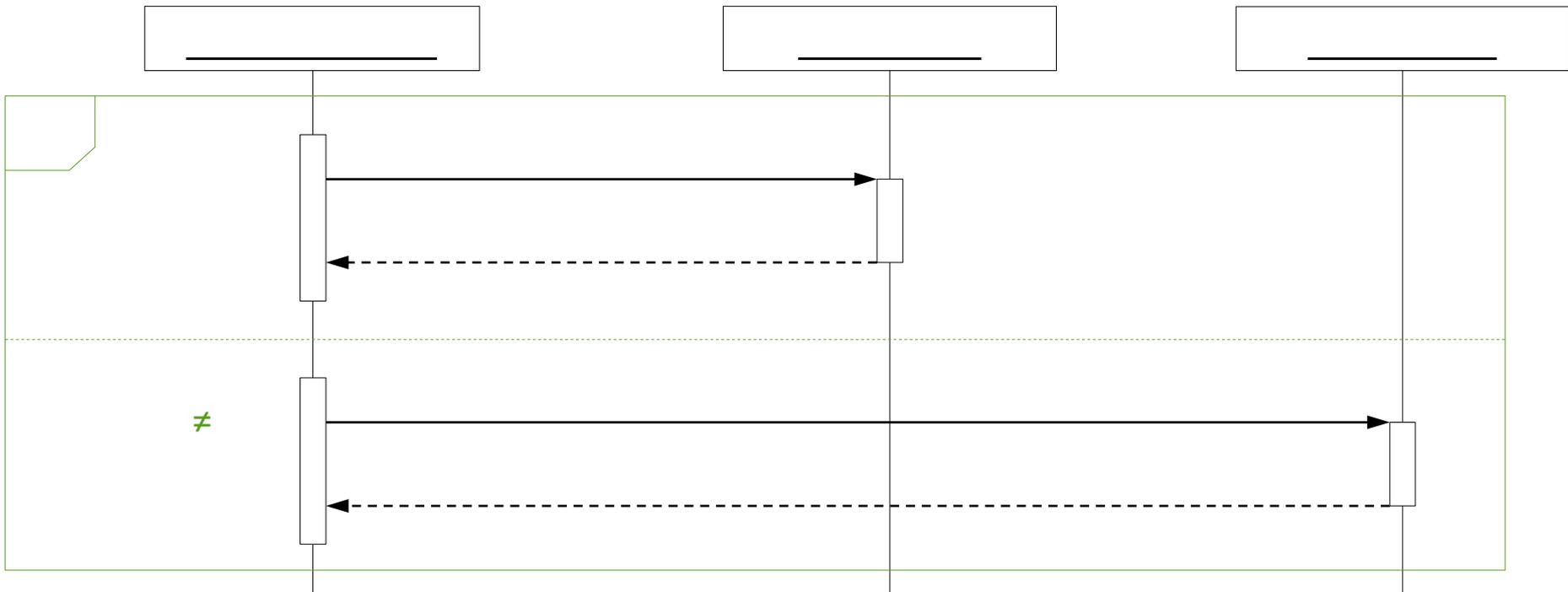
- Deux diagrammes



Alternative

à l'envoi d'un message

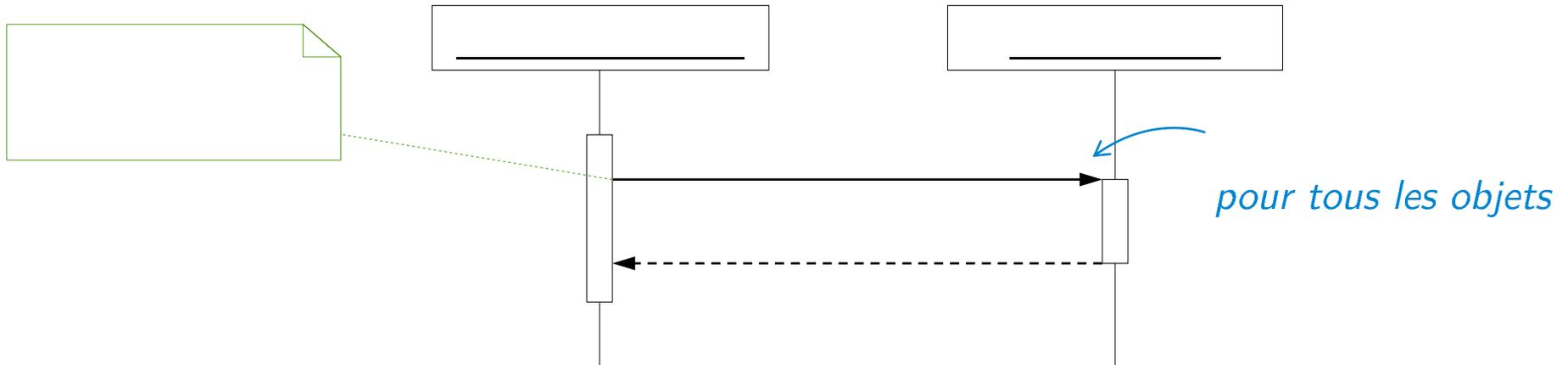
- Deux diagrammes
- Bloc d'alternative



Boucle

un enchaînement de messages

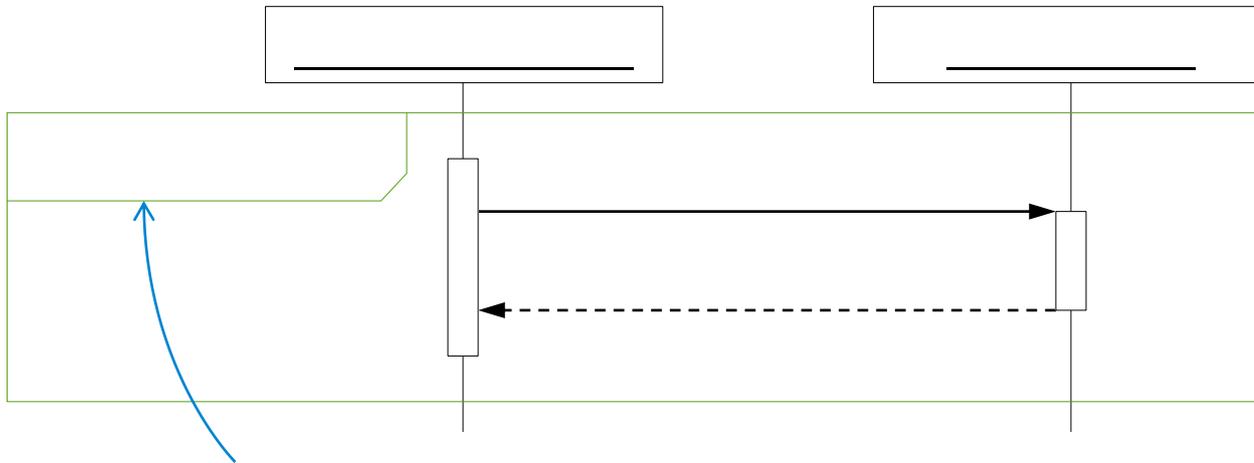
-



Boucle

un enchaînement de messages

-
- Bloc de boucle



Référence à un autre diagramme

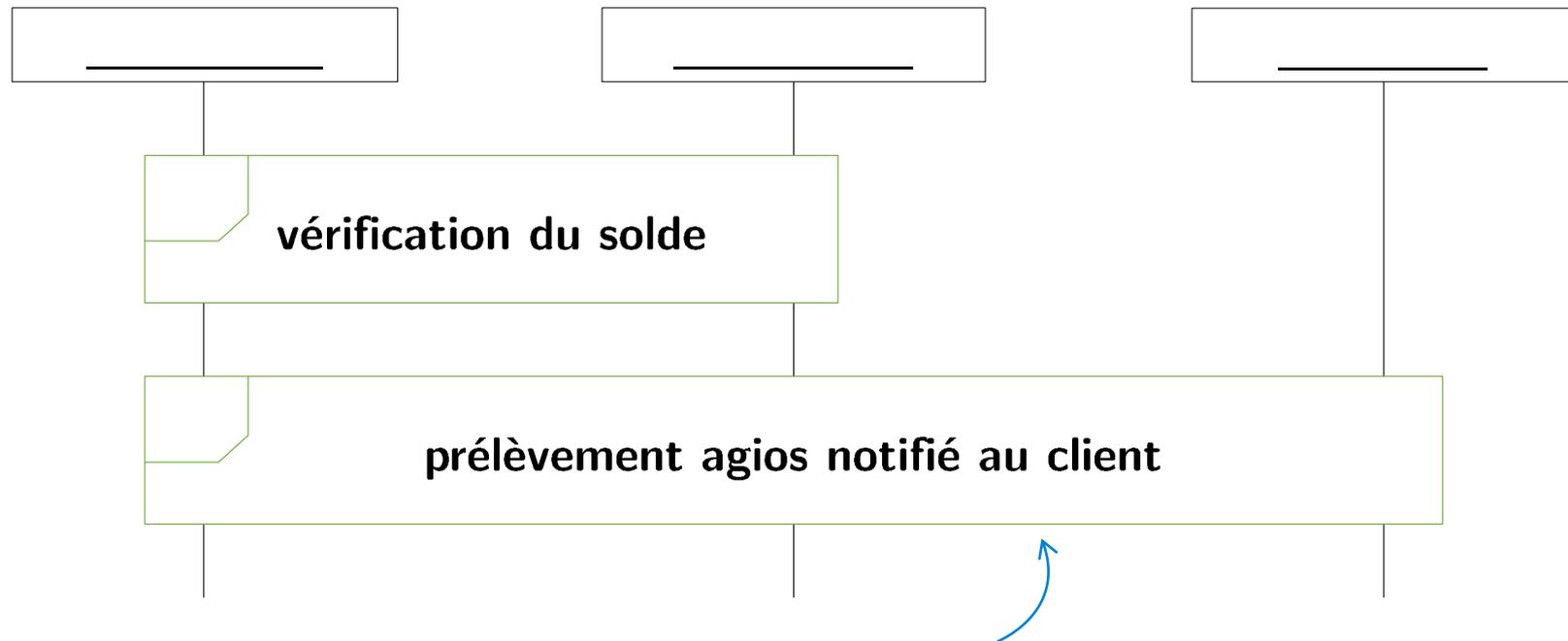
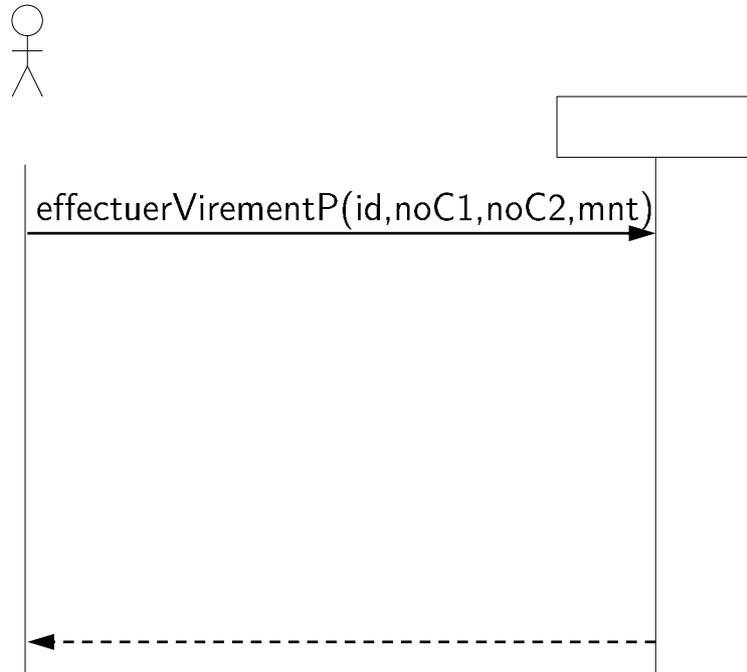
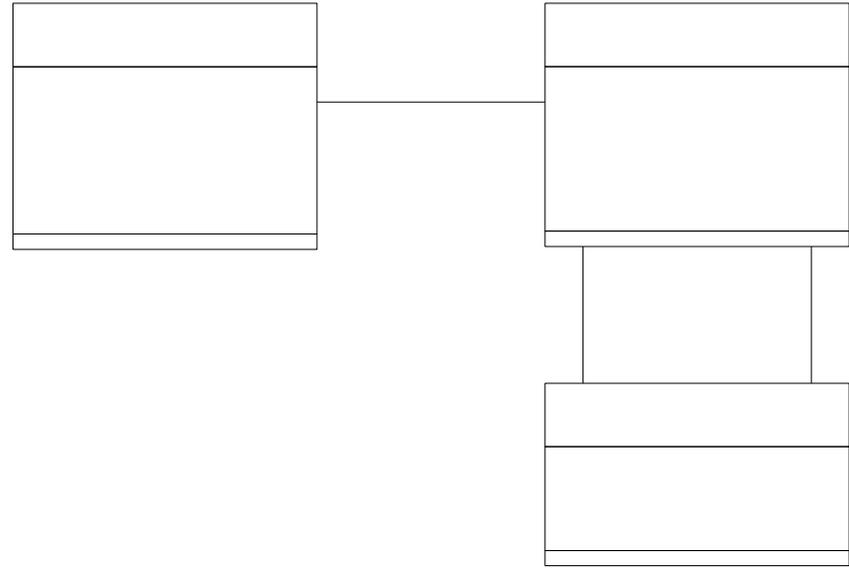
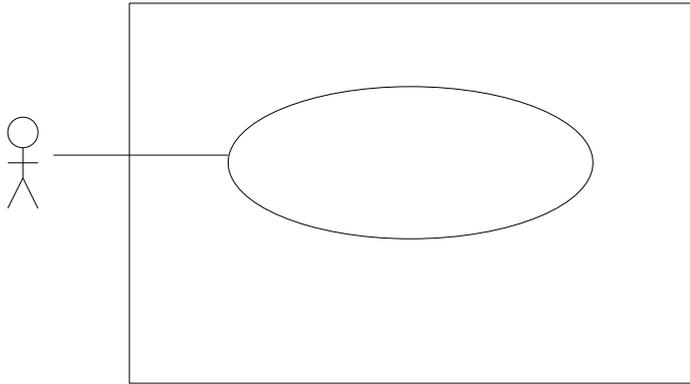


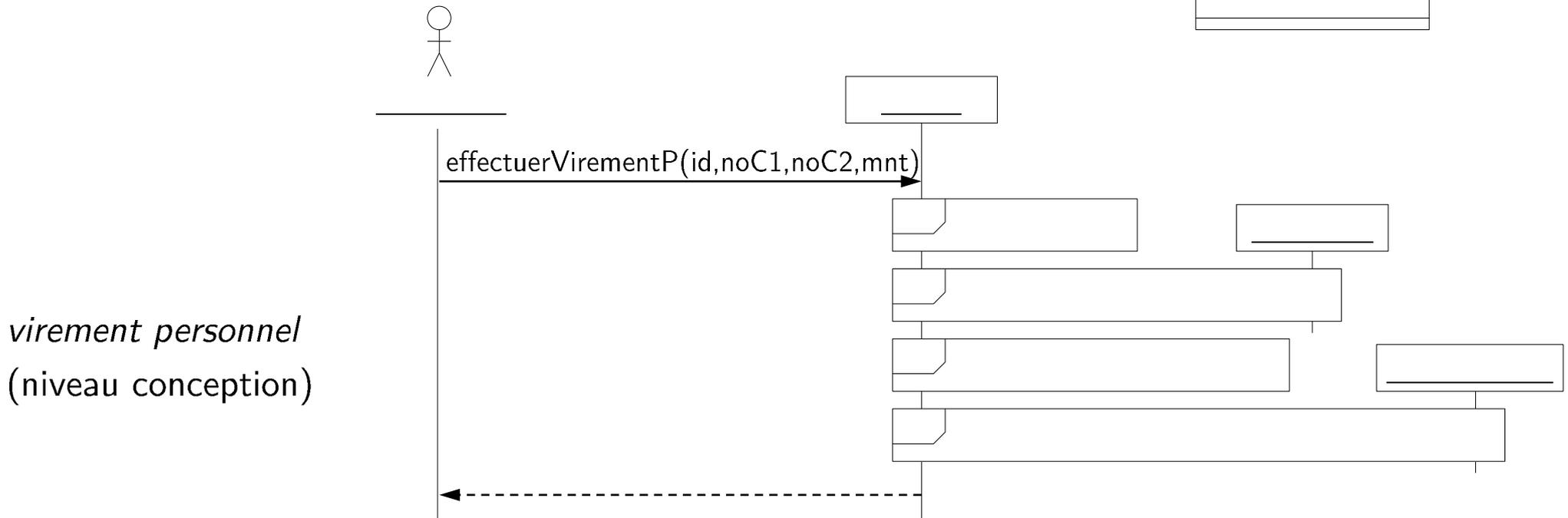
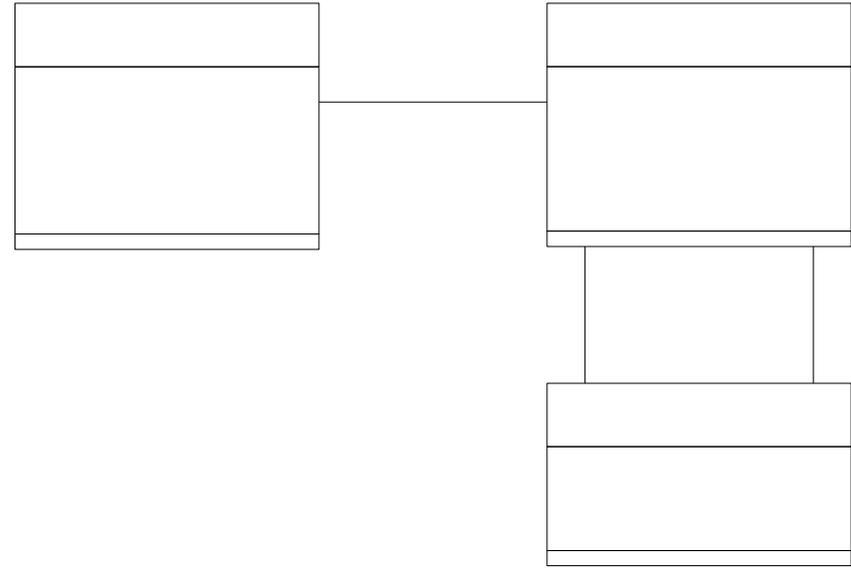
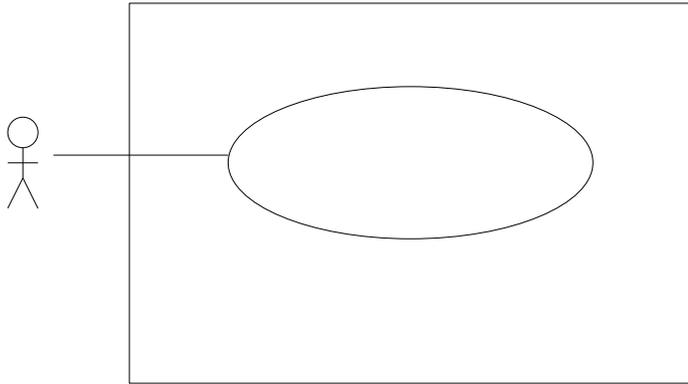
diagramme décrit ailleurs

Exemple Analyse

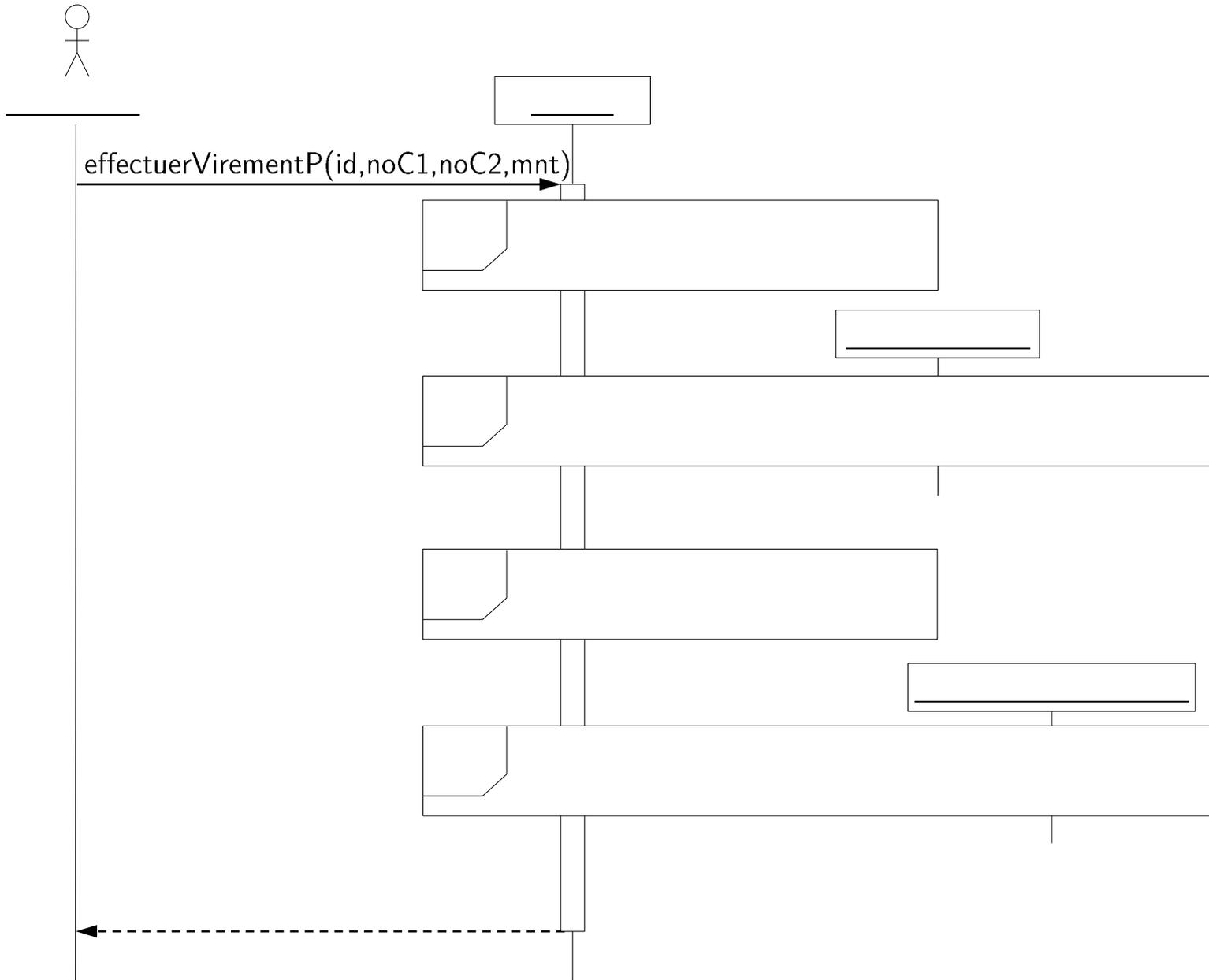


virement personnel

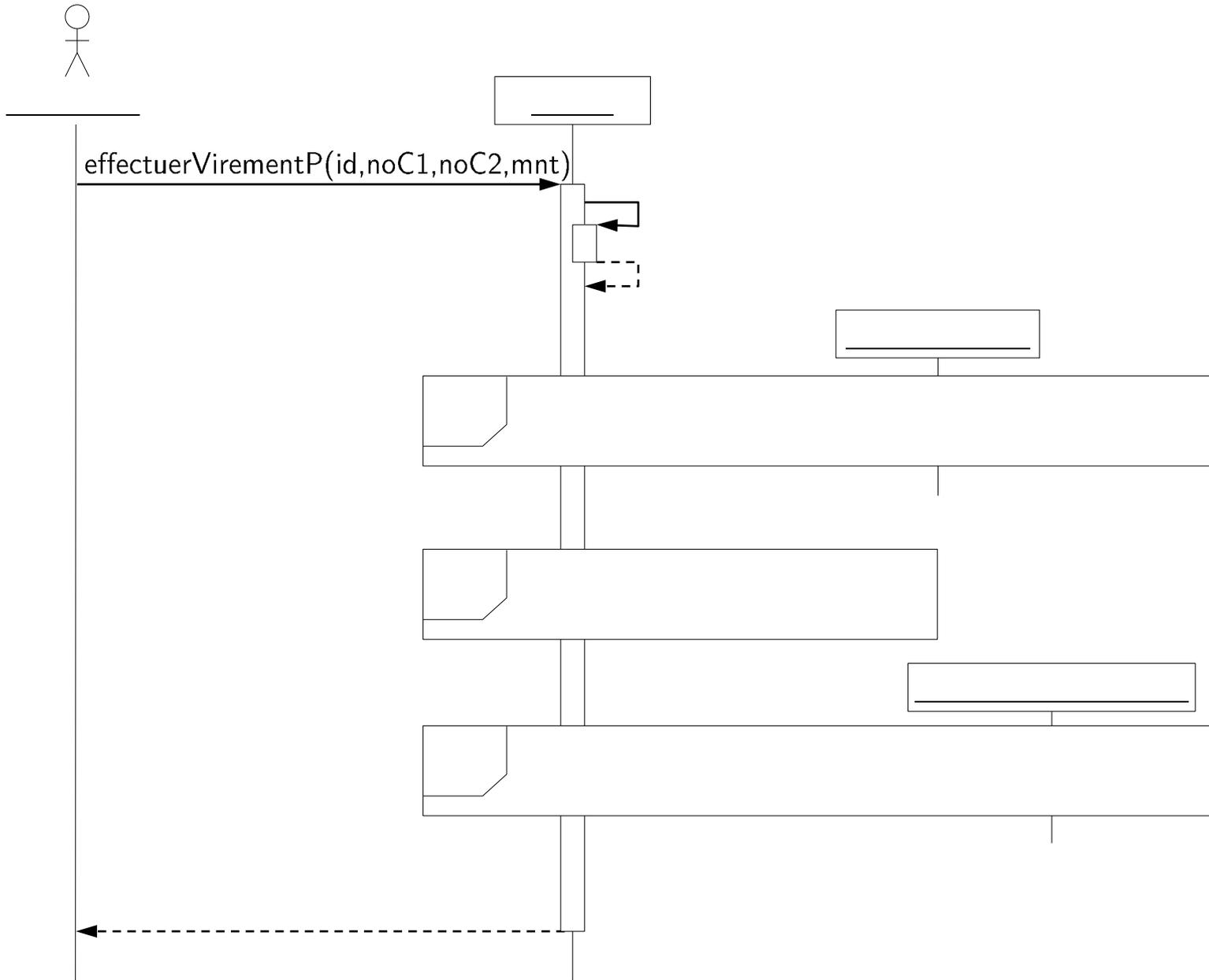
Exemple Conception



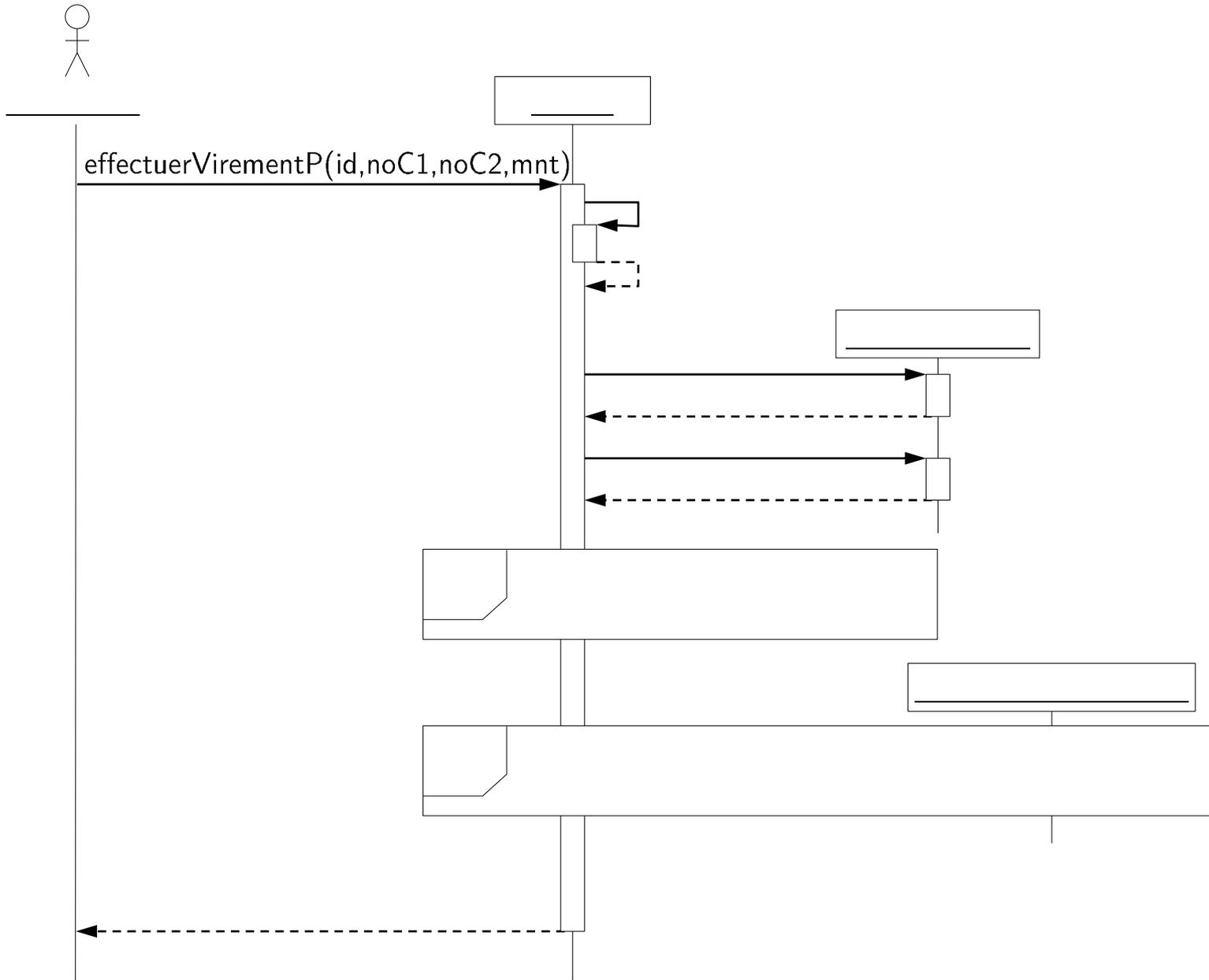
Exemple Conception



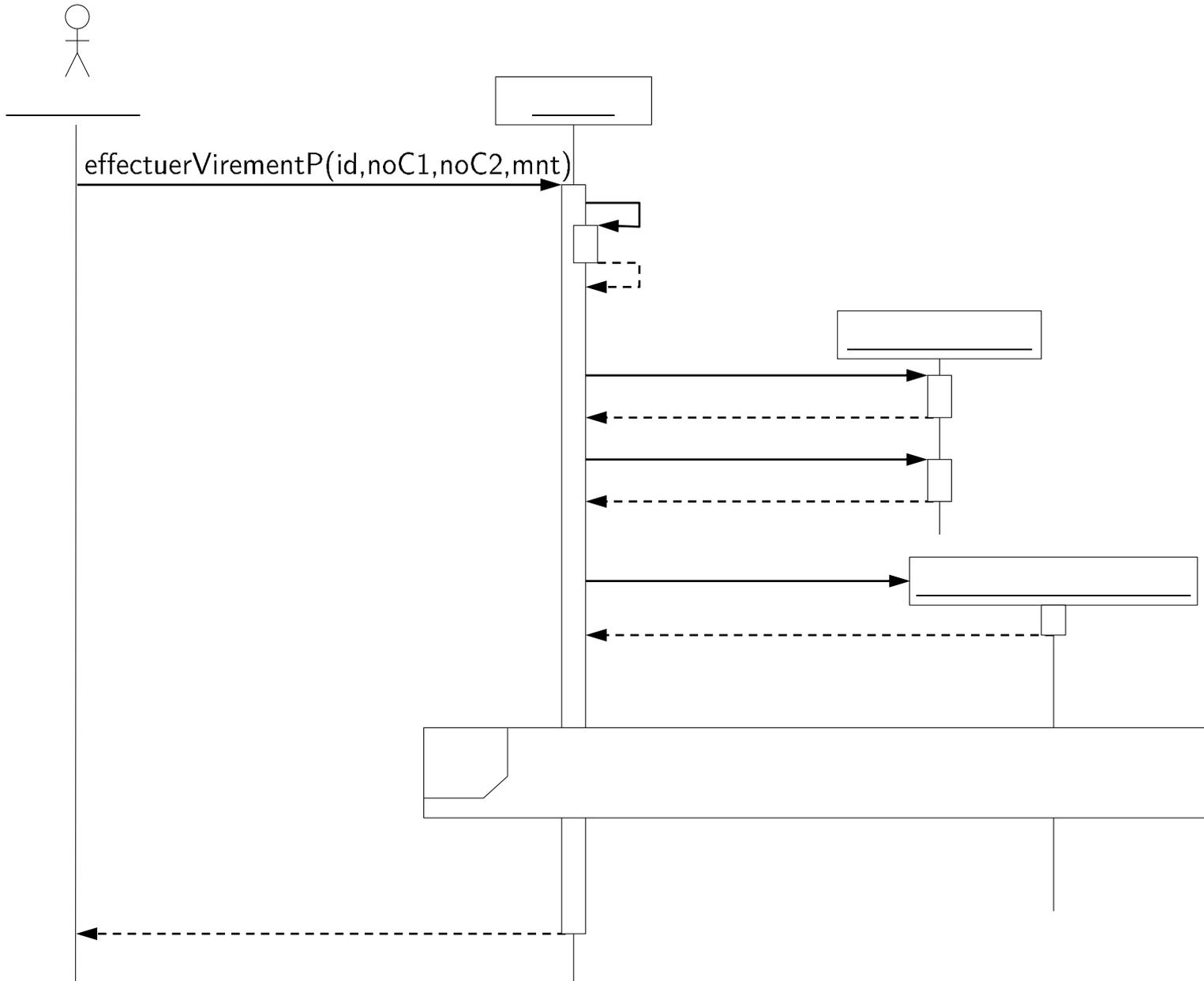
Exemple Conception



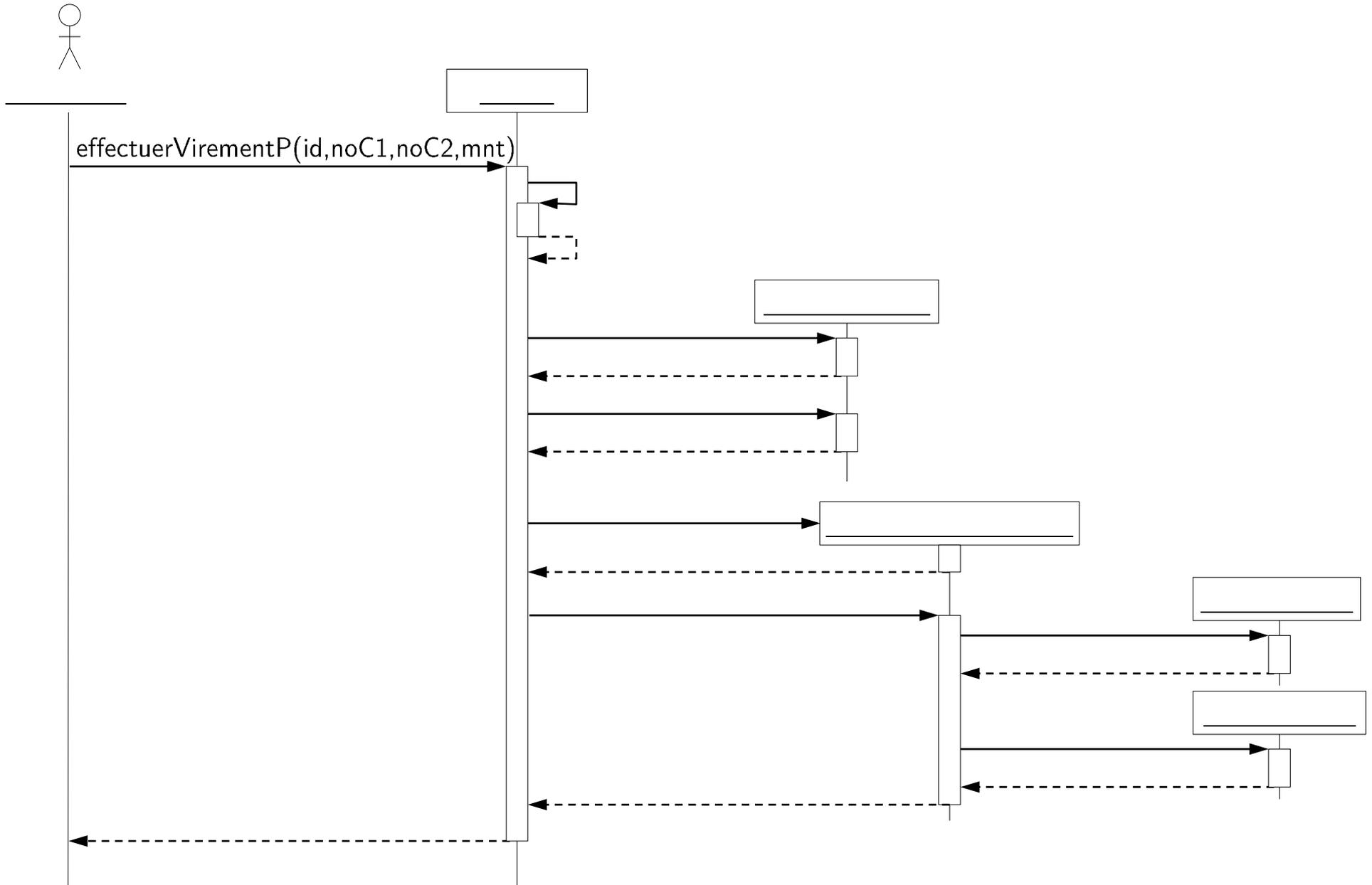
Exemple Conception



Exemple Conception



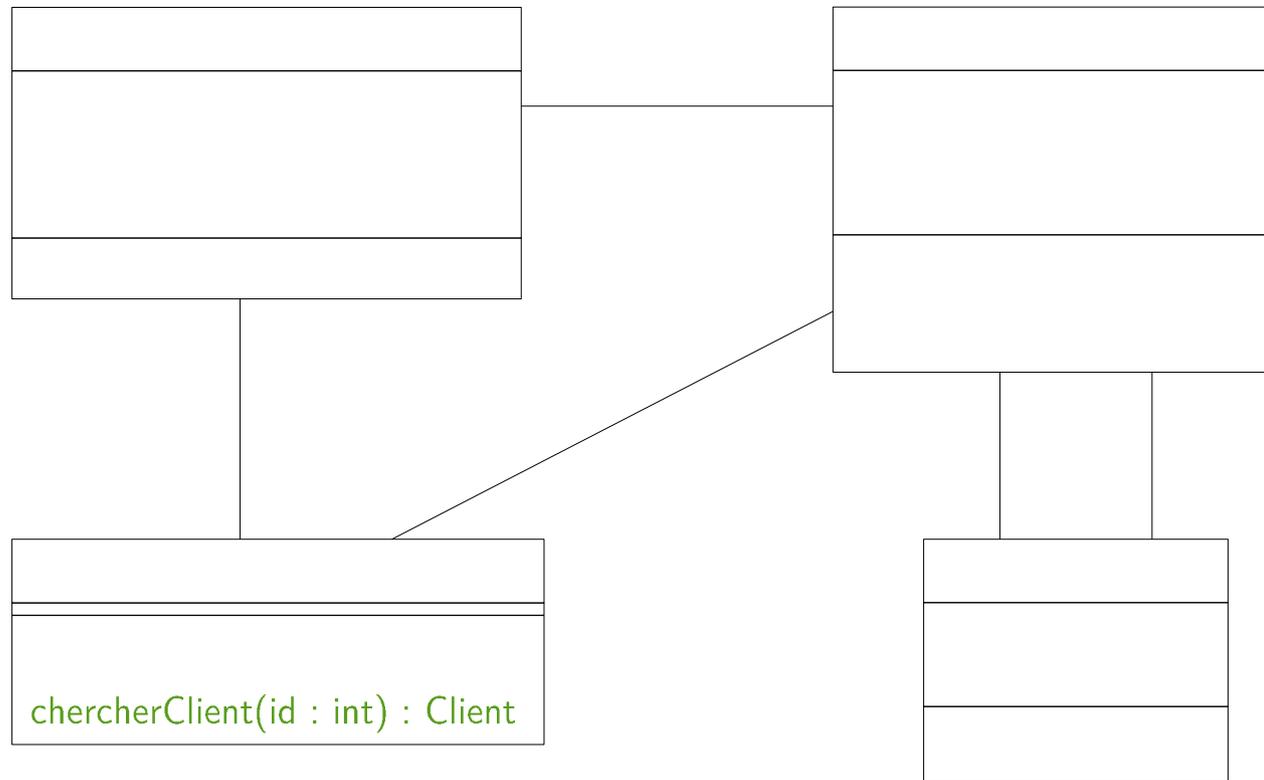
Exemple Conception



Exemple Conception

Diagramme de classes
opérations nécessaires

avec les classes techniques et les



Quelques règles

Messages entre acteurs et interface

- « » opérations liées au **cas d'utilisation** (même nom)
- Arguments (saisis) et valeurs de retour (affichées)

Messages au sein du système

- **Opérations** du diagramme de classes
- Si message de **objA : ClasseA** **objB : ClasseB**
opération du message dans ClasseB