

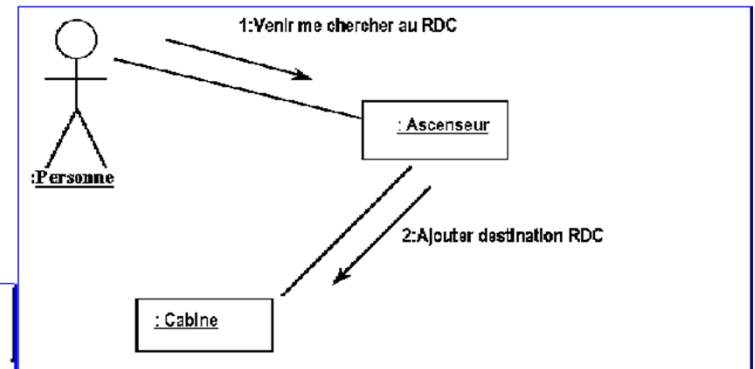
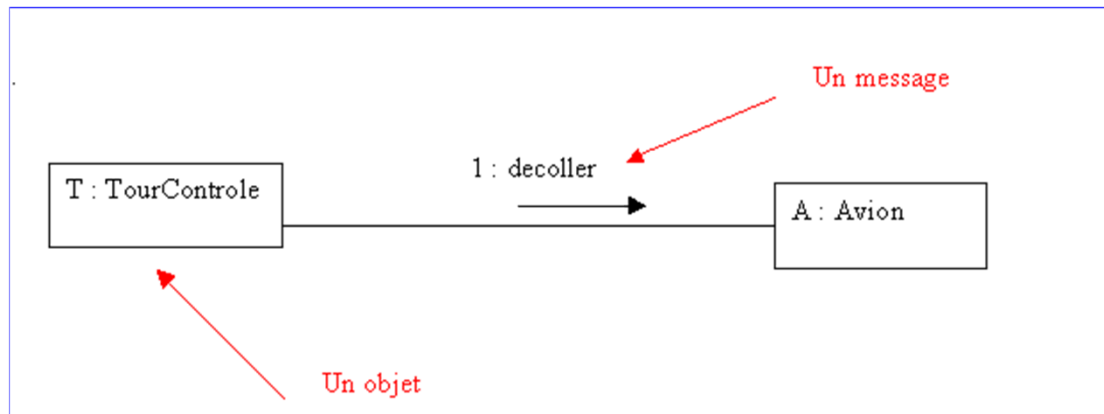
Modélisation des interactions

Diagrammes de séquence

Diagrammes de communication

Définitions

- Une interaction est un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.



Message

- Un message est une communication entre objets. La réception d'un message peut produire un résultat.
- Types de messages:
 - Flux de contrôle
 - Envoi d'un signal: Déclenche une réaction de la part du récepteur de façon asynchrone et sans qu'une réponse ne soit nécessaire.
 - appel d'une opération: applique une opération à un objet de façon synchrone ou asynchrone, et peut requérir une réponse de la part de son récepteur
 - Actions
 - création d'un objet
 - suppression d'un objet existant
- Ces messages correspondent aux stimulus dans le langage UML.
- Un message a un émetteur, un récepteur et une action. L'action est exécutée sur le récepteur. Vous pouvez également créer des messages récursifs, c'est-à-dire des messages dont l'émetteur est en même temps le récepteur.

- UML fournit deux types de représentation des interactions :
 - Les **diagrammes de communication**
 - Les **diagrammes de séquence**
- Les deux types de diagrammes sont équivalents. La différence entre ces deux diagrammes est une différence de centre d'intérêt. Les diagrammes de communication mettent en évidence les liens entre objets et les séquences d'envois de messages sur ces liens. Les diagrammes de séquence mettent en évidence la chronologie des envois de messages.

Spécification

- Le symbole d'un message est une flèche qui montre son sens, et qui peut également afficher les informations suivantes :
 - Un numéro d'ordre qui indique l'ordre dans lequel les messages sont échangés
 - Le nom du message (ou nom de l'opération associée)
 - La condition
 - La valeur de résultat:
 - Les arguments

Exemple de diagramme de communication

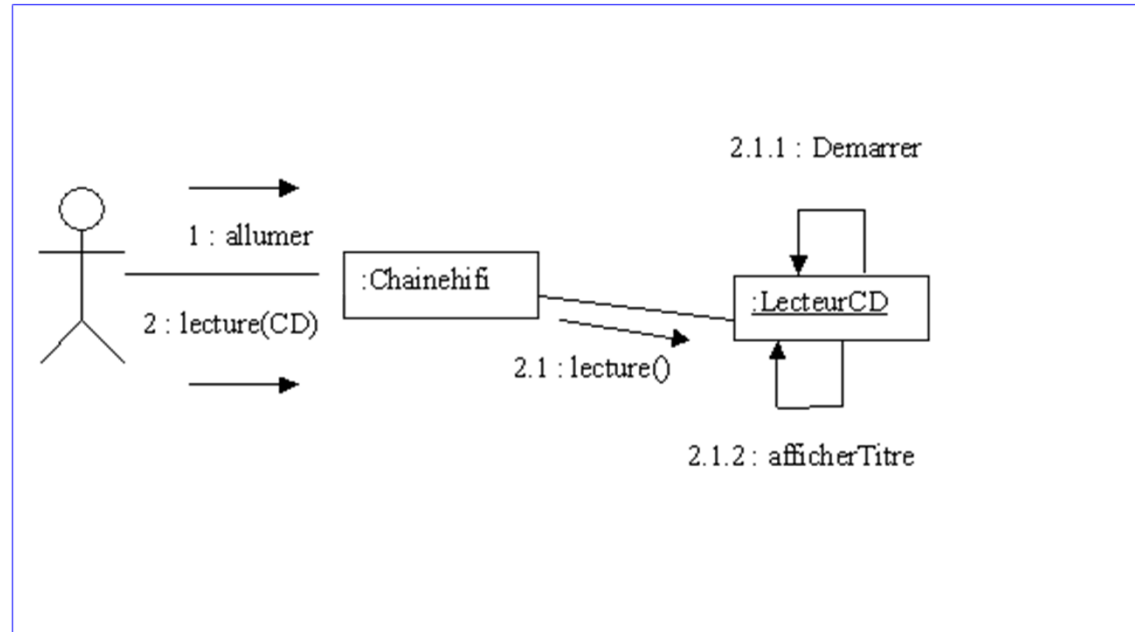


Diagramme de séquence

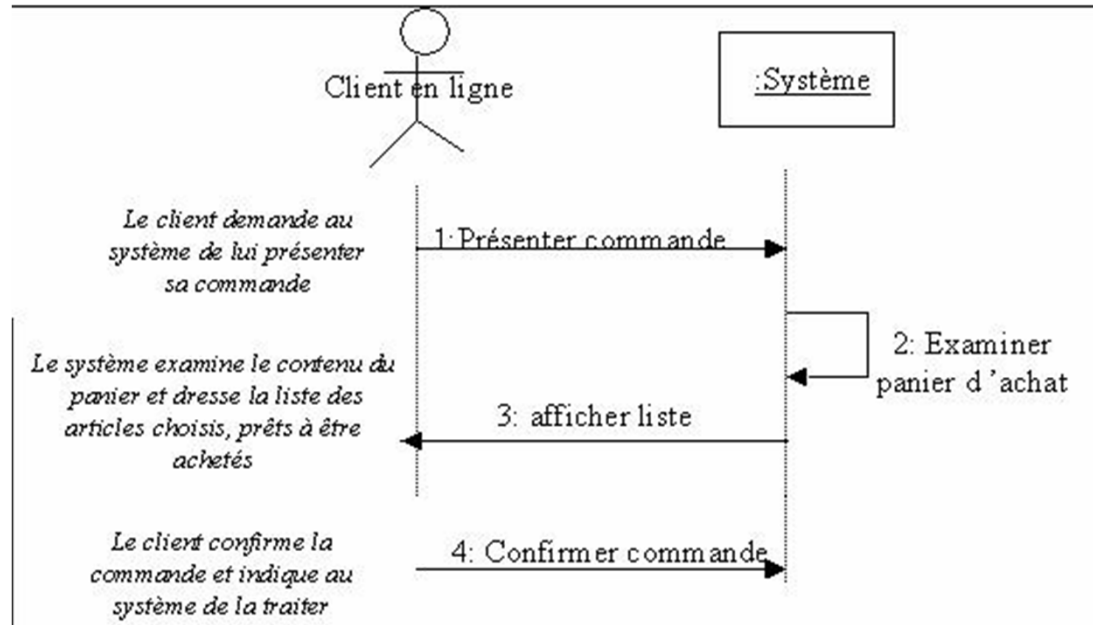
- Le diagramme de séquence est une description graphique des opérations d'un système sous un angle chronologique. C'est une vue dynamique qui contient les symboles d'objets (instances de classe), d'acteurs et de messages qu'ils échangent.
- La dimension verticale est l'axe temporel : les messages y sont représentés par ordre chronologique. La dimension horizontale montre des objets et des acteurs qui échangent des informations.

- Les éléments représentés sur un diagramme de séquence sont:
 - Les **objets** qui participent à l'interaction avec éventuellement leurs états,
 - Les **messages** entre les objets,
 - Une ligne verticale en pointillé est attachée à chaque objet et représente **sa ligne de vie**,
 - Une **période d'activité** correspond au temps pendant lequel un objet effectue une action, soit directement, soit par l'intermédiaire d'un autre objet qui lui sert de sous-traitant. Les périodes d'activité se représentent par des bandes rectangulaires placées sur les lignes de vie. Le début et la fin d'une bande correspondent respectivement au début et à la fin d'une période d'activité,
 - **Les points de contrôle**. Il est possible d'ajouter du pseudo-code sur la partie gauche du diagramme pour représenter des boucles et des branchements,
 - **Les contraintes temporelles** éventuelles

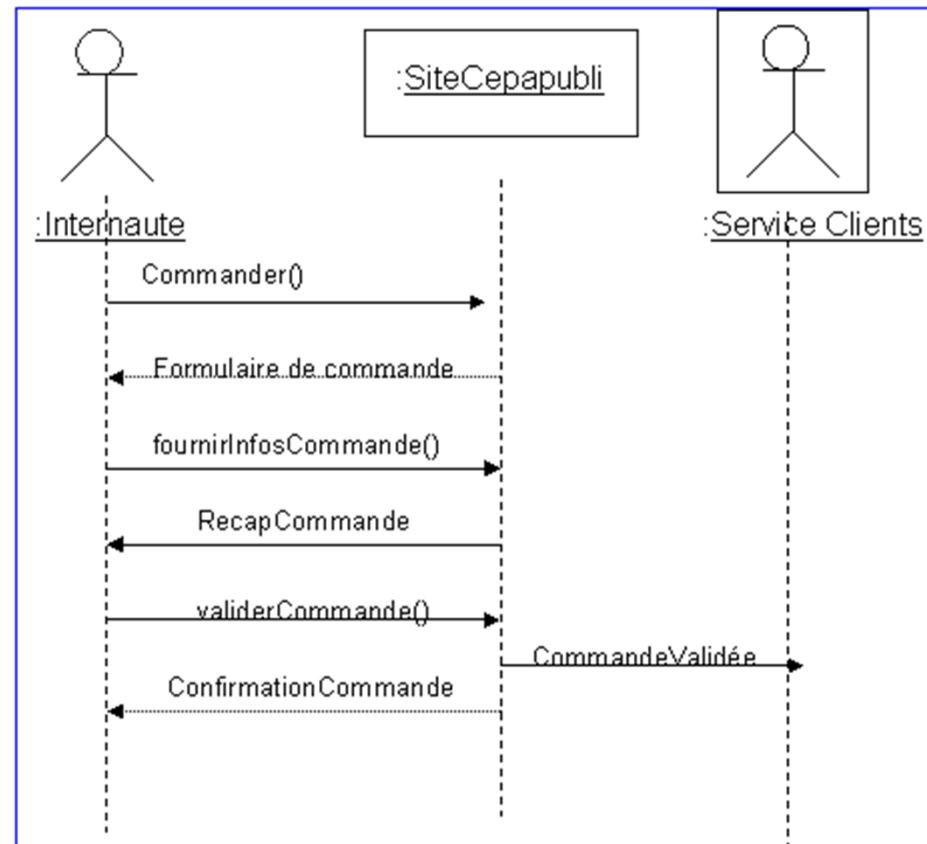
Utilisation

- Flots d'exécution des UC.
- Fournir des informations sur le contrôle de flux.
- Transaction de base de données
- Dégager les paramètres clés qui doivent être partagés entre les élément du système.
- Avoir une vision sur la création et la destruction des objets dans le système.
- Identifier les objets qui envoient trop de messages.
- Avoir des information sur l'aspect performance si un objet reçoit trop de messages.
- Une méthode simple et visuelle afin que les utilisateurs valident les étapes qu'ils entreprennent (ou souhaite entreprendre) et dans quel ordre.

Exemple



Exemple 2



- La syntaxe d'un message est la suivante :
 pré "/" [[["cond"]] [séqu] ["*"["|"]]["["iter"]]] ":" [r "!="] msg("["par"]")

pré : prédécesseurs (liste de numéros de séquence de messages séparés par une virgule).

cond : condition de garde, expression booléenne.

séqu : numéro de séquence du message.

Indique le rang du message, c'est-à-dire son numéro d'ordre par rapport aux autres messages. Les messages sont numérotés à la façon de chapitres dans un document, à l'aide de chiffres séparés par des points. Ainsi, il est possible de représenter le niveau d'emboîtement des messages et leur précédence.

Exemple : l'envoi du message 1.3.5 suit immédiatement celui du message 1.3.4 et ces deux messages font partie du flot (de la famille de messages) 1.3.

Pour représenter l'envoi simultané de deux messages, il suffit de les indexer par une lettre.

Exemple : l'envoi des messages 1.3.a et 1.3.b est simultané.

iter : récurrence du message.

Permet de spécifier en langage naturel l'envoi séquentiel (ou en parallèle, avec "|") de messages. Notez qu'il est aussi possible de spécifier qu'un message est récurrent en omettant la clause d'itération (en n'utilisant que "*" ou "*|").

r : valeur de retour du message.

Permet d'affecter la valeur de retour d'un message, pour par exemple la retransmettre dans un autre message, en tant que paramètre.

msg : nom du message.

par : paramètres (optionnels) du message.

Exemples

Exemple 1

3 : bonjour()

Ce message a pour numéro de séquence "3".

Exemple 2

[heure = midi] 1 : manger()

Ce message n'est envoyé que s'il est midi.

Exemple 3

*1.3.6 * : ouvrir()*

Ce message est envoyé de manière séquentielle un certain nombre de fois.

Exemple 4

*3 / * || [i := 1..5] : fermer()*

Représente l'envoi en parallèle de 5 messages. Ces messages ne seront envoyés qu'après l'envoi du message 3.