

Chapitre IV: Diagramme des cas d'utilisation

Nour H. BEN SLIMEN ATTAOUI

UML: DIAGRAMME DE CAS D'UTILISATION

Introduction

- ▶ Les cas d'utilisation ('use cases') servent à exprimer le comportement du système en termes d'actions et de réactions, selon le point de vue de chaque utilisateur.
- ▶ Les cas d'utilisation délimitent le système, ses fonctions (ses cas), et ses relations avec son environnement. Ils constituent un moyen de déterminer les besoins du système.
- ▶ Pour élaborer les cas d'utilisation, il faut se fonder sur des entretiens avec les utilisateurs,

UML: DIAGRAMME DE CAS D'UTILISATION

Introduction

- ▶ Destiné à l'expression du besoin.
- ▶ Complète le modèle objet en offrant une vision « fonctionnelle » du système.
- ▶ Centré sur les utilisateurs.
- ▶ Formalisme très simple.
- ▶ Peut également servir à la conception des tests de validation.

Dans tous les cas, il faut bien garder à l'esprit qu'il n'y a pas de notion temporelle dans un diagramme de cas d'utilisation.

UML: DIAGRAMME DE CAS D'UTILISATION

Modélisation des besoins

- ▶ Avant de développer un système, il faut savoir **précisément** à **QUOI** il devra servir, c'àd à quels besoins il devra répondre.
- ▶ Modéliser les besoins permet de:
 - ▶ Faire l'inventaire des fonctionnalités attendues;
 - ▶ Organiser les besoins entre eux, de manière à faire apparaître des relations (réutilisations possibles, ...).
- ▶ Avec UML, on modélise les besoins au moyen de **diagrammes de cas d'utilisation**.

UML: DIAGRAMME DE CAS D'UTILISATION

Comment recenser les cas d'utilisation?

- ▶ Chaque cas d'utilisation correspond donc à une fonction métier du système, selon le point de vue d'un de ses acteurs.
- ▶ Aussi, pour identifier les cas d'utilisation, il faut se placer du point de vue de chaque acteur et déterminer **comment** et surtout **pourquoi** il se sert du système.
- ▶ Nommez les cas d'utilisation avec un verbe à l'infinitif suivi d'un complément en vous plaçant du point de vue de l'acteur et non pas de celui du système. Par exemple, un distributeur de billets aura probablement un cas d'utilisation Retirer de l'argent et non pas Distribuer de l'argent.

UML: DIAGRAMME DE CAS D'UTILISATION

démarche d'utilisation des cas d'utilisation

1. Identifier les acteurs et les cas.
2. Décrire les cas en écrivant des scénarios sous forme textuelle.
3. Dessiner les diagrammes de cas.
4. Récapituler les cas, les structurer et s'assurer de la cohérence d'ensemble

UML: DIAGRAMME DE CAS D'UTILISATION

Éléments des diagrammes de cas d'utilisation

I. Acteur

- ▶ Un acteur est une personne ou un système qui interagit avec le système étudié, en échangeant de l'information.
- ▶ On trouve les acteurs en observant les utilisateurs directs du système, les responsables de sa maintenance, ainsi que les autres systèmes qui interagissent avec lui.
- ▶ Un acteur représente un rôle joué par un utilisateur qui interagit avec le système. La même personne physique peut jouer le rôle de plusieurs acteurs. D'autre part, plusieurs personnes peuvent jouer le même rôle, et donc agir comme un même acteur.

UML: DIAGRAMME DE CAS D'UTILISATION

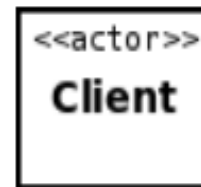
Éléments des diagrammes de cas d'utilisation

- Un acteur se représente par un petit bonhomme avec son nom (i.e. son rôle) inscrit dessous
- Il est également possible de représenter un acteur sous la forme d'un classeur stéréotypé << actor >>



Client

Exemple de représentation d'un acteur.

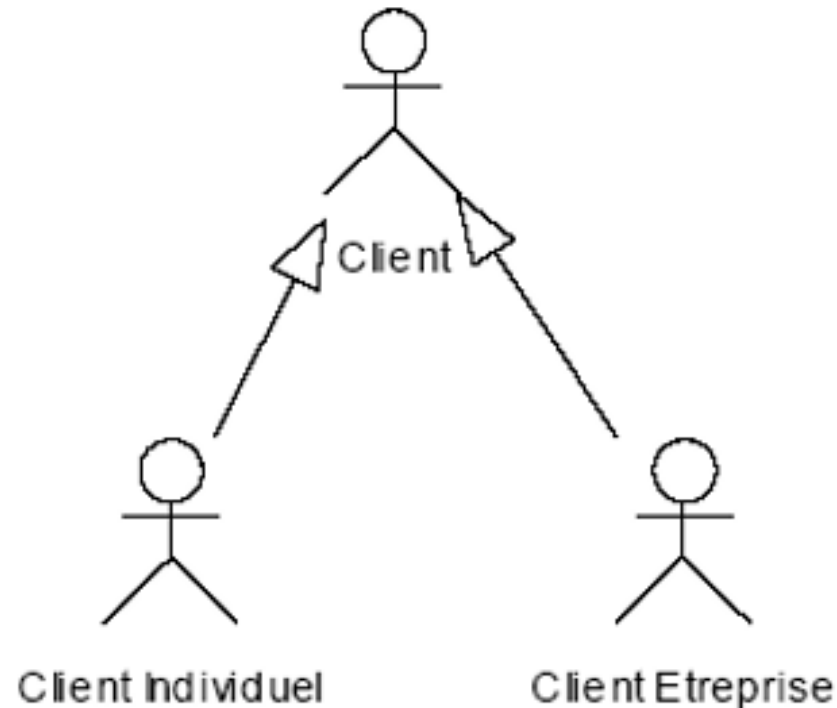


Exemple de représentation d'un acteur sous la forme d'un classeur.

UML: DIAGRAMME DE CAS D'UTILISATION

Éléments des diagrammes de cas d'utilisation

- On peut utiliser des relations de généralisation entre les acteurs pour les définir de manière plus précise:



UML: DIAGRAMME DE CAS D'UTILISATION

Éléments des diagrammes de cas d'utilisation

Acteurs **principaux** et **secondaires** :

- ▶ Un acteur est qualifié de principal pour un cas d'utilisation lorsque ce cas rend service à cet acteur. Les autres acteurs sont alors qualifiés de secondaires.
- ▶ Un acteur principal obtient un résultat observable du système tandis qu'un acteur secondaire est sollicité pour des informations complémentaires.

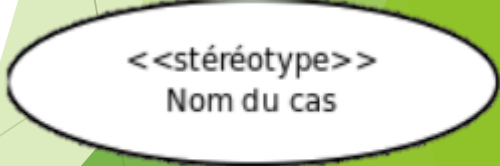
UML: DIAGRAMME DE CAS D'UTILISATION

Éléments des diagrammes de cas d'utilisation

II. Cas d'utilisation

Un cas d'utilisation “raconte” comment on doit utiliser le système pour atteindre un but particulier :

- ▶ Il correspond à une fonction du système (sans entrer dans les détails de chaque scénario).
- ▶ Il décrit les interactions entre le système et les utilisateurs.
- ▶ Il se représente par une ellipse contenant le nom du cas (un verbe à l'infinitif), et optionnellement, au-dessus du nom, un stéréotype.
- ▶ Il est connecté à un acteur à travers une association représentée par une ligne continue (avec ou sans flèche).

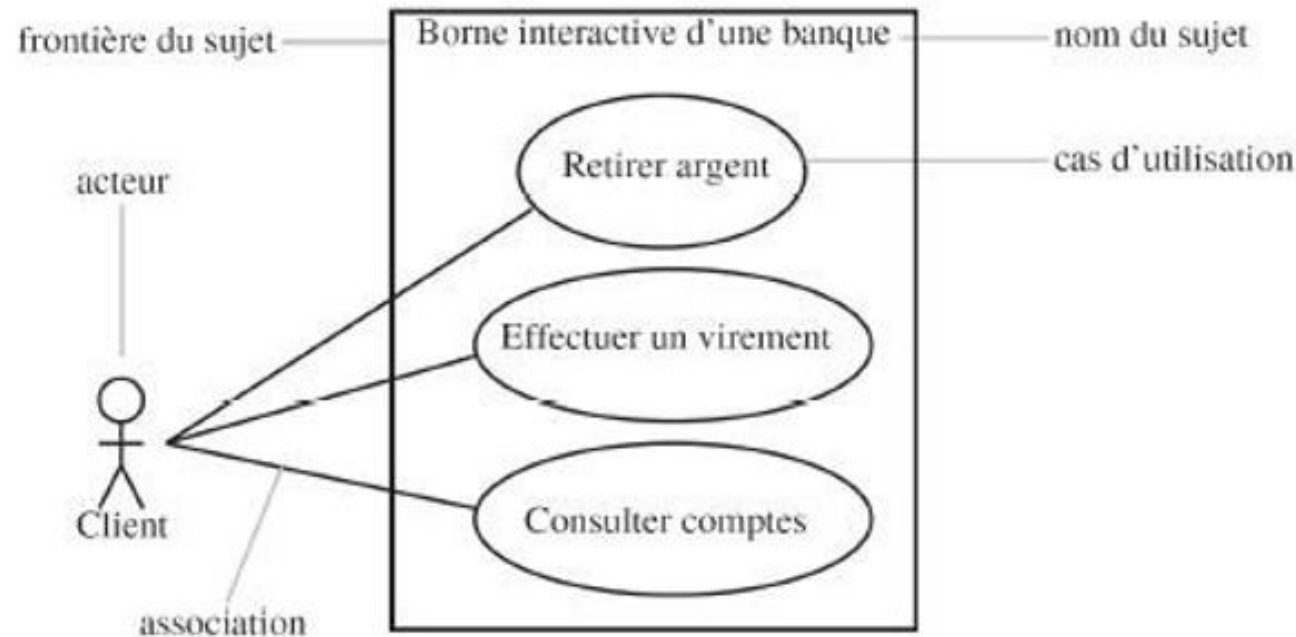


<<stéréotype>>
Nom du cas

UML: DIAGRAMME DE CAS D'UTILISATION

Représentation d'un diagramme de cas d'utilisation

- La frontière du système est représentée par un cadre. Le nom du système figure à l'intérieur du cadre, en haut. Les acteurs sont à l'extérieur et les cas d'utilisation à l'intérieur,

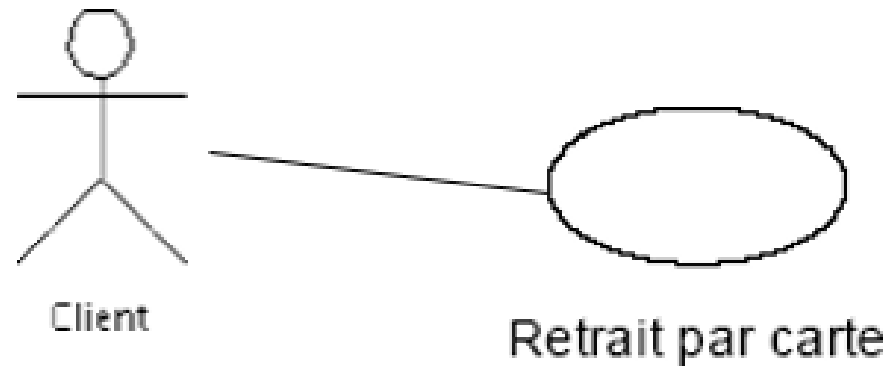


UML: DIAGRAMME DE CAS D'UTILISATION

Relations dans le diagramme des cas d'utilisation

Relations entre acteurs et cas d'utilisation

- Relation d'association est une communication entre un acteur et un cas d'utilisation et est représenté un trait continu.



UML: DIAGRAMME DE CAS D'UTILISATION

Relations dans le diagramme des cas d'utilisation

Relations entre les cas d'utilisation

- ▶ **Généralisation:**
- ▶ **Inclusion**
- ▶ **Extension**

UML: DIAGRAMME DE CAS D'UTILISATION

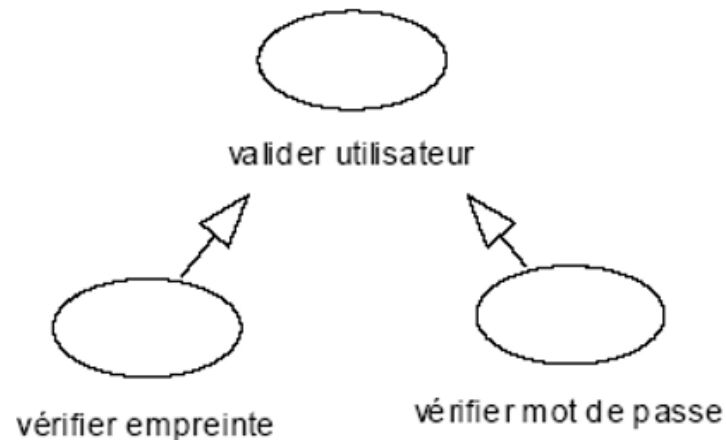
Relations dans le diagramme des cas d'utilisation

Relations entre les cas d'utilisation

► Généralisation:

- Elle est identique à celle des classes.
- Un cas d'utilisation hérite du comportement et de la signification du cas d'utilisation parent.
- L'enfant peut remplacer ou compléter le cas d'utilisation parent.

Exemple:



UML: DIAGRAMME DE CAS D'UTILISATION

Concepts

Relations entre les cas d'utilisation

► L'inclusion:

Une relation “**include**” du cas d'utilisation **A** vers le cas d'utilisation **B** signifie que le flot d'événements de **A** contient une séquence d'événements qui correspond à **B**.

Le cas d'utilisation inclus n'est jamais tout seul, il fait toujours partie d'un cas d'utilisation qui l'englobe. L'inclusion permet de :

- factoriser des cas d'utilisation, qui peuvent servir à d'autres cas d'utilisation: éviter de décrire le même comportement plusieurs fois
- simplifier un cas d'utilisation, en le décomposant en plusieurs cas d'utilisation. La relation d'inclusion est représentée par une dépendance à laquelle on attribue le stéréotype «include»,

UML: DIAGRAMME DE CAS D'UTILISATION

Relations dans le diagramme des cas d'utilisation

Relations entre les cas d'utilisation

► L'inclusion:

Un cas 'A' inclut un cas 'B' si le comportement décrit par le cas 'A' inclut le comportement du cas 'B' : le cas 'A' dépend de 'B'. Lorsque 'A' est sollicité, 'B' l'est obligatoirement, comme une partie de 'A'.

Cette dépendance est symbolisée par le stéréotype << include >>. Par exemple, l'accès aux informations d'un compte bancaire inclut nécessairement une phase d'authentification avec un identifiant et un mot de passe.

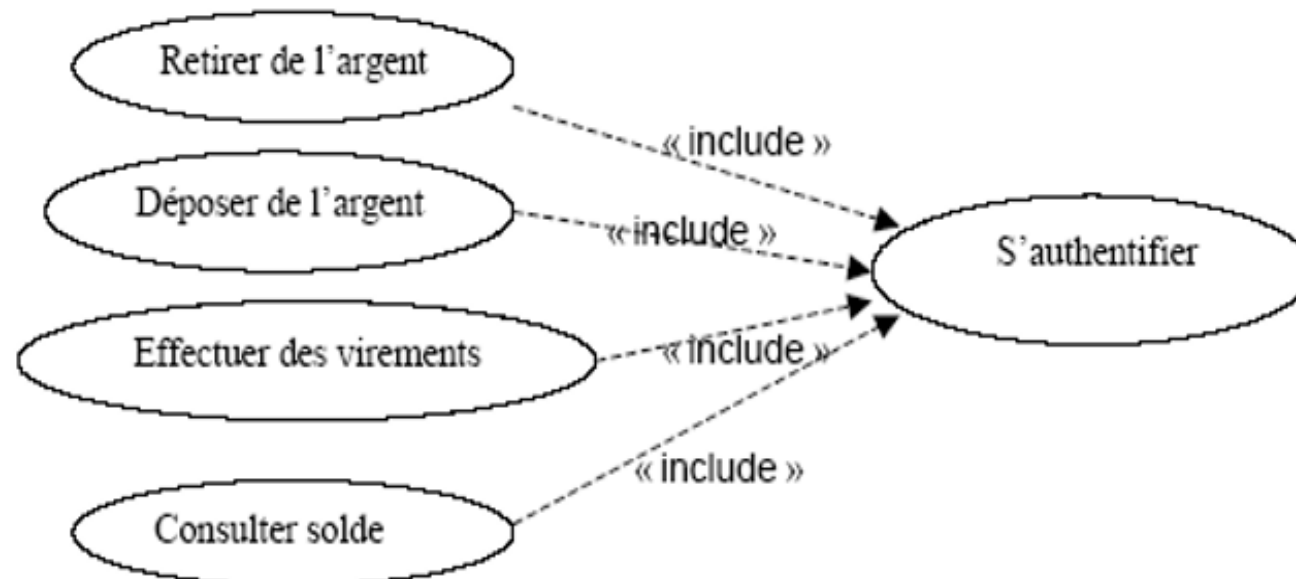
UML: DIAGRAMME DE CAS D'UTILISATION

Relations dans le diagramme des cas d'utilisation

Relations entre les cas d'utilisation

► L'inclusion:

Exemple:



UML: DIAGRAMME DE CAS D'UTILISATION

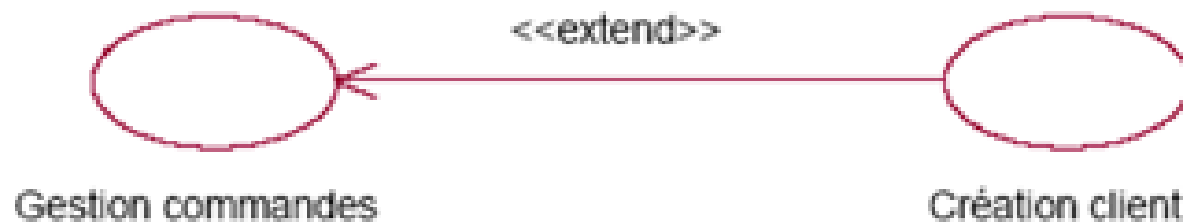
Relations dans le diagramme des cas d'utilisation

Relations entre les cas d'utilisation

► L'extension:

Une relation « extend » du cas d'utilisation 'A' vers le cas d'utilisation 'B' signifie que le flot d'événements de 'A' peut intervenir, de façon facultative, pendant le déroulement de 'B'. 'B' spécifie un comportement facultatif.

Exemple:



UML: DIAGRAMME DE CAS D'UTILISATION

Description textuelle des cas d'utilisation

- ▶ Le diagramme de cas d'utilisation décrit les grandes fonctions d'un système du point de vue des acteurs, mais n'expose pas de façon détaillée le dialogue entre les acteurs et les cas d'utilisation.
- ▶ Bien que de nombreux diagrammes d'UML permettent de décrire un cas, il est recommandé de rédiger une description textuelle, car c'est une forme souple qui convient dans bien des situations.
- ▶ Une description textuelle couramment utilisée se compose de deux parties.

UML: DIAGRAMME DE CAS D'UTILISATION

Description textuelle des cas d'utilisation

I. La première partie permet d'identifier le cas d'utilisation, elle doit contenir les informations qui suivent:

- ▶ **Nom** : utiliser une tournure à l'infinitif (ex. : Réceptionner un colis).
- ▶ **Objectif** : une description résumée permettant de comprendre l'intention principale du cas d'utilisation. Cette partie est souvent renseignée au début du projet dans la phase de découverte des cas d'utilisation.
- ▶ **Acteurs principaux** : ceux qui vont réaliser le cas d'utilisation (la relation avec le cas d'utilisation est illustrée par le trait liant le cas d'utilisation et l'acteur dans un diagramme de cas d'utilisation).
- ▶ **Acteurs secondaires** : ceux qui ne font que recevoir des informations à l'issue de la réalisation du cas d'utilisation.
- ▶ **Dates** : les dates de création et de mise à jour de la description courante.
- ▶ **Responsable** : le nom des responsables.

UML: DIAGRAMME DE CAS D'UTILISATION

Description textuelle des cas d'utilisation

- II. La deuxième partie contient la description du fonctionnement du cas sous la forme d'une séquence de messages échangés entre les acteurs et le système. Elle contient toujours une séquence nominale qui décrit le déroulement normal du cas. À la séquence nominale s'ajoutent fréquemment des séquences alternatives (des embranchements dans la séquence nominale) et des séquences d'exceptions (qui interviennent quand une erreur se produit).
 - ▶ **Les préconditions** : elles décrivent dans quel état doit être le système (l'application) avant que ce cas d'utilisation puisse être déclenché.
 - ▶ **Des scénarios** : ces scénarios sont décrits sous la forme d'échanges d'événements entre l'acteur et le système. On distingue le scénario nominal, qui se déroule quand il n'y a pas d'erreur, des scénarii alternatifs qui sont les variantes du scénario nominal et enfin les scénario d'exception qui décrivent les cas d'erreurs.
 - ▶ **Des postconditions** : elles décrivent l'état du système à l'issue des différents scénarios.

UML: DIAGRAMME DE CAS D'UTILISATION

Exemple: Vente de tickets de cinéma

Pour l'achat des tickets de cinéma, on dispose d'un distributeur relié au système permettant aux clients d'acheter eux-mêmes leurs tickets pour les réservations individuelles.

Le guichetier est une personne qui vend aussi des tickets pour les réservations individuelles, mais aussi des tickets pour les abonnements. Le guichetier est relié au système à travers un terminal. Il peut faire des échanges de tickets et aussi vérifier leurs disponibilités. Le gérant supervise les ventes en interrogeant à tout moment le système. La vente des tickets peut se faire avec carte de crédits en utilisant une connexion avec le système de la banque.