

Chapitre 3

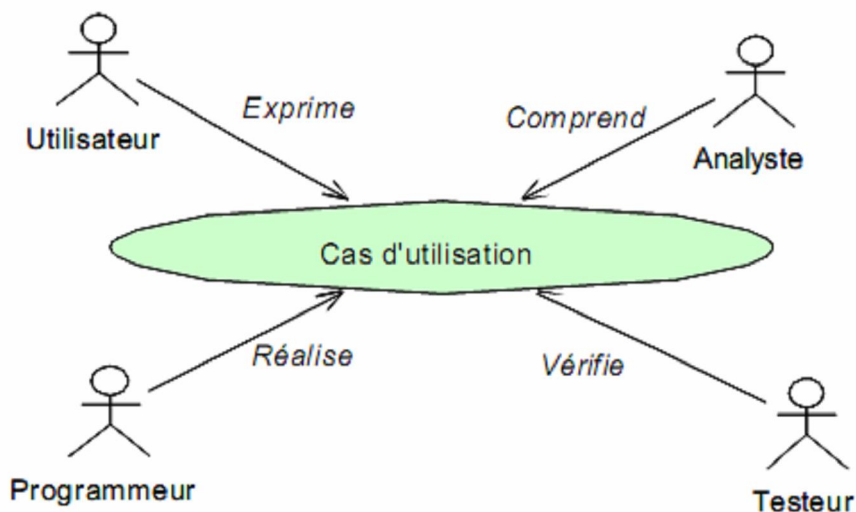
Diagrammes UML Vue Statique

1. Diagramme de cas d'utilisation

1.1 Introduction

Les cas d'utilisation ont été formalisés par Ivar Jacobson en 1992.

Un diagramme de cas d'utilisation scinde la fonctionnalité du système en unités cohérentes, **les cas d'utilisation**, ayant un sens pour **les acteurs**. Les cas d'utilisation permettent d'exprimer le besoin des utilisateurs d'un système, ils sont donc une vision orientée utilisateur de ce besoin au contraire d'une vision informatique.



Pour élaborer les cas d'utilisation, il faut se fonder sur des entretiens avec les utilisateurs.

Diagramme de cas d'utilisation = acteurs + cas d'utilisation + relations

Démarche:

- 1 Partitionner l'ensemble des besoins pour réduire la complexité de leur détermination.
- 2 Focaliser sur les vrais besoins

Terminologie : Langage naturel

1.2 Intérêt

Résolution des problèmes :

- 1 De non compréhension des besoins exprimés de manière non structurée et sans cohérence,
- 2 De besoins contradictoires,
- 3 D'oublis et d'imprécisions
- 4 De difficultés d'application et

5 Du coût d'une application.

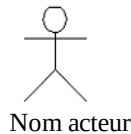
1.3 Eléments des diagrammes des cas d'utilisation

1.3.1 Acteur

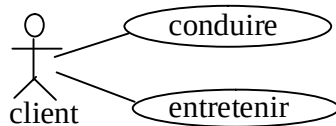
Définition :

Un acteur représente un rôle joué par une personne externe, un processus ou une chose qui interagit avec un système (pour l'exploiter ou le maintenir).

Représentation :



Une même personne physique peut jouer le rôle de plusieurs acteurs. Exemple :



Catégories d'acteurs :

1. *Acteurs principaux* : utilisent les fonctions principales du système. Exemple : *Client* dans le cas d'un *système de distribution de billets*.
2. *Acteurs secondaires* : effectuent des tâches administratives ou de maintenance. Exemple : *Rechargeur* de la caisse du distributeur dans le cas d'un *système de distribution de billets*.

Chaque acteur identifié doit être décrit clairement.

Remarque : Si le nombre d'acteurs est grand, il est conseillé de les regrouper par paquetages pour simplifier le modèle de cas d'utilisation.

1.3.2 Cas d'utilisation

Définition :

Un cas d'utilisation est une unité cohérente d'une fonctionnalité visible de l'extérieur. Il réalise un service de bout en bout, avec un déclenchement, un déroulement et une fin, pour l'acteur qui l'initie. Un cas d'utilisation modélise donc un service rendu par le système, sans imposer le mode de réalisation de ce service.

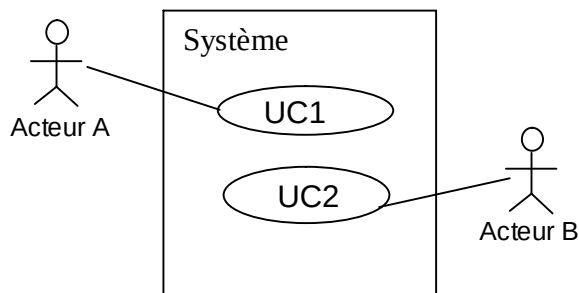
Représentation :



Le nom d'un cas d'utilisation peut être précédé d'un stéréotype et suivi de propriétés.

Représentation d'un diagramme de cas d'utilisation :

Les cas peuvent être contenus dans un rectangle limitant le système.



1.3.3 Relations dans un diagramme d'utilisation

Relation entre acteur et cas d'utilisation

Relation d'association : Une relation d'association est chemin de communication entre un acteur et un cas d'utilisation et est représentée par un trait continu.

Lorsqu'un acteur peut interagir plusieurs fois avec un cas d'utilisation, il est possible d'ajouter une multiplicité sur l'association du côté du cas d'utilisation.

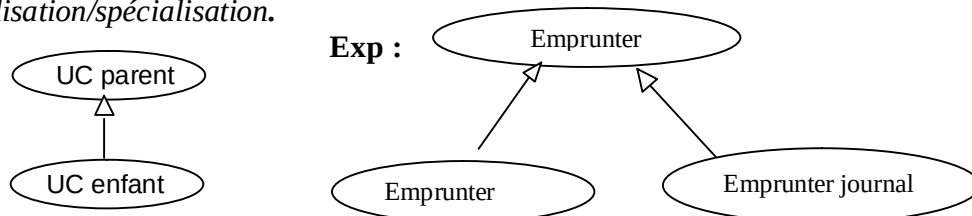
- Le symbole * signifie plusieurs ;
- exactement n s'écrit tout simplement n,
- n..m signifie entre n et m, etc.

Cas d'utilisation interne : Quand un cas n'est pas directement relié à un acteur, il est qualifié de cas d'utilisation interne.

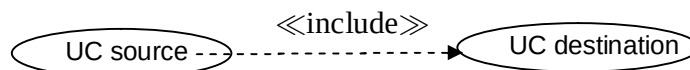
Relation entre cas d'utilisation

UML définit 3 types de relations entre cas d'utilisation :

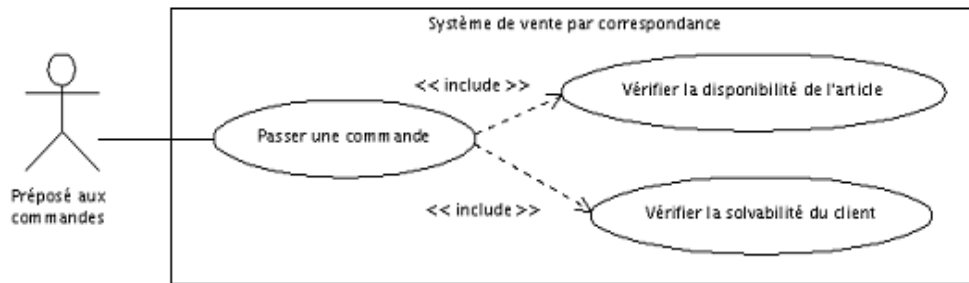
- *La généralisation/spécialisation.*



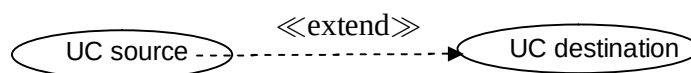
- *L'inclusion* : Un cas A inclut un cas B si le comportement décrit par le cas A inclut le comportement du cas B. Le cas A dépend de B. Lorsque A est sollicité, B l'est obligatoirement, comme une partie de A.



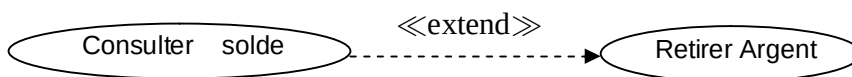
Exp :



- *L'extension* : On dit qu'un cas d'utilisation A étend un cas d'utilisation B lorsque le cas d'utilisation A peut être appelé au cours de l'exécution du cas d'utilisation B. Exécuter B peut éventuellement entraîner l'exécution de A : contrairement à l'inclusion, l'extension est optionnelle. Cette dépendance est symbolisée par le stéréotype « extend ».



Exp :



Relation entre acteurs

La seule relation possible entre deux acteurs est la généralisation : un acteur A est une généralisation d'un acteur B si l'acteur A peut être substitué par l'acteur B. Dans ce cas, tous les cas d'utilisation accessibles à A le sont aussi à B, mais l'inverse n'est pas vrai.

