

# MODÉLISATION UML

Mme Sande MAZOUZ



# Diagramme de cas d'utilisation

# Diagrammes de cas d'utilisation

3

- *Que doit faire mon logiciel ?*



C'est la question  
primordiale !

# DCU (Suite)

4

- Une réflexion sur les fonctionnalités attendues du futur système avant la conception ;
- Avoir une idée sur les grands modules du système ;
- Les fonctionnalités que doit fournir chaque composant ;
- Ces fonctionnalités vont aider les utilisateurs à effectuer leur « mission ».

# DCU (Suite)

5

- La détermination et la compréhension des besoins sont souvent difficiles ;
- Il faut clarifier et organiser les besoins des clients (les modéliser).
- Les use cases permettent de structurer les besoins des utilisateurs et les objectifs correspondants au système ;

# DCU (Suite)

6

- Ils centrent l'expression des exigences du système sur ses utilisateurs : ils partent du principe que les objectifs du système sont tous motivés ;
- Ils se limitent aux préoccupations "*réelles*" des utilisateurs ;
- ils ne présentent pas de solutions d'implémentation et ne forment pas un inventaire fonctionnel du système ;

# DCU (Suite)

7

- Ils identifient les utilisateurs du système (acteurs) et leur interactions avec le système ;
- Ils permettent de classer les acteurs et structurer les objectifs du système;
- Ils servent de base à la traçabilité des exigences d'un système dans un processus de développement intégrant UML;
- **Les cas d'utilisation représentent le premier modèle à concevoir .**

# DCU (Suite)

8

- Le DCU a été formalisé par **Ivar Jacobson** . Il décrit sous la forme *d'actions et de réactions le comportement d'un système du point de vue d'un utilisateur ;*
- Il permet de définir les frontières du système et les relations entre le système et l'environnement ;
- Un DCU comprend les acteurs, le système et les cas d'utilisation ;
- En UML, un **cas d'utilisation** modélise un service (ou une fonctionnalité) rendu par le système.



# Les acteurs

9

## □ Qu'est ce qu'un acteur ?

- Un acteur représente un ensemble cohérent de rôles joué par **quelque chose** ou **quelqu'un** se trouvant dans l'environnement du système étudié ;
- Un acteur est en relation avec le métier de l'entreprise et interagit avec le système dans différents cas d'utilisation ;
- Il peut être un élément de la structure de l'entreprise tel qu'une direction, un service ou un poste de travail.

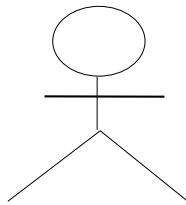
**Acteur != utilisateur**

# Les acteurs (Suite)

10

## Notation :

Les acteurs sont représentés par un *stick man* ou une classe « **actor** ».



Acteur



# Les acteurs (Suite)

11

- Il ne faut pas confondre la notion d'acteur avec la notion d'utilisateur.
  - ▣ Un acteur décrit le **rôle** qu'un utilisateur joue par rapport au système.
  - ▣ Un utilisateur c'est une **personne** utilisant le système.
- Une même personne physique peut être modélisée par plusieurs acteurs.

# Les acteurs (Suite)

12

## □ Comment déterminer les acteurs ?

Les acteurs se déterminent en observant les utilisateurs directs du système, ceux qui sont responsables de son exploitation ou de sa maintenance, ainsi que les autres systèmes qui interagissent avec le système en question.

# Les acteurs (Suite)

13

## Types d'acteurs :

Quartres catégories d'acteurs :

- Acteurs principaux, acteurs secondaires, matériel externe et autres systèmes.
- Par défaut, le rôle d'un acteur est « **principal** ». Si ce n'est pas le cas on peut indiquer le rôle « secondaire » sur l'acteur.

# Les acteurs (Suite)

14

- Si un acteur a pour rôle unique de consommer des informations du système sans modifier l'état de celui-ci au niveau métier alors son rôle est **secondaire ;**
- Un acteur peut aussi être un périphérique externe ou un système externe.

# Les acteurs (Suite)

15

- **Les acteurs principaux** : les personnes qui **utilisent le système** pour réaliser leur but.
- **Les acteurs secondaires** ou bien **auxiliaires** : Ils sont sollicités pour des informations complémentaires.
- **Le matériel externe** : les dispositifs matériels incontournables qui **font partie du domaine** de l'application et qui doivent être **utilisés** (capteurs, imprimantes, périphériques...).
- **Les autres systèmes** : les systèmes avec lesquels le système doit **interagir** ( **service central, organisation...**).

# Les acteurs (Suite)

16



Exemple d'acteurs



# Les cas d'utilisations

17

## □ *Que veut faire chaque acteur ?*

Il s'agit ici de définir les fonctionnalités du logiciel pour chacun des acteurs.

## □ **Un cas d'utilisation (use case) :**

Un cas d'utilisation est une manière spécifique d'utiliser un système. **C'est l'image d'une fonctionnalité du système, déclenchée en réponse à la stimulation d'un acteur.**

# Les cas d'utilisations (Suite)

18

- Un cas d'utilisation est représenté par une ellipse appelée : **stéréotype**.
- Les cas d'utilisation sont contenus dans un rectangle appelé **classeur** qui représente les limites du système.
- Les acteurs sont alors forcément à l'extérieur du rectangle puisqu'ils ne font pas partie du système.

# Les cas d'utilisations (Suite)

19

- La fonction réalisée par le CU est représentée sous forme **d'un verbe à l'infinitif**.
- Les fonctionnalités du système exprimées textuellement seront modélisées à l'aide des cas d'utilisations.

# Les cas d'utilisations (Suite)

20

## Exemple:

- Ajouter un produit au panier pour un site e-commerce.



Ajouter produit à  
son panier

A UML Use Case diagram consisting of an oval shape. Inside the oval, the text "Ajouter produit à son panier" is written in a bold, black, sans-serif font, centered horizontally and vertically.

- Créer une nouvelle réservation dans un cabinet médical.



Créer réservation

A UML Use Case diagram consisting of an oval shape. Inside the oval, the text "Créer réservation" is written in a bold, black, sans-serif font, centered horizontally and vertically.

# Le système

21

- Le classeur contient **le libellé** qui correspond au nom du système.
- Le classeur sert à délimiter le système par rapport à son environnement et aux autres systèmes.
- Le système est un ensemble de cas d'utilisation. De cette manière, UML modélise le système et les services rendus par le système.

# Le système (Suite)

22

**Borne interactive du système**



**Cas d'utilisation 1**

**Cas d'utilisation 2**

**Exemple Rectangle**

# Les scénarios

23

- Les cas d'utilisation se déterminent en observant et en précisant, acteur par acteur, **les séquences d'interaction -les scénarios-** du point de vue de l'utilisateur.
- Un cas d'utilisation regroupe **une famille de scénarios** d'utilisation selon un critère fonctionnel.
- Les cas d'utilisation sont des **abstractions** du dialogue entre les acteurs et le système: **ils décrivent des interactions potentielles**, sans entrer dans le détail de chaque scénario.

# Les scénarios (Suite)

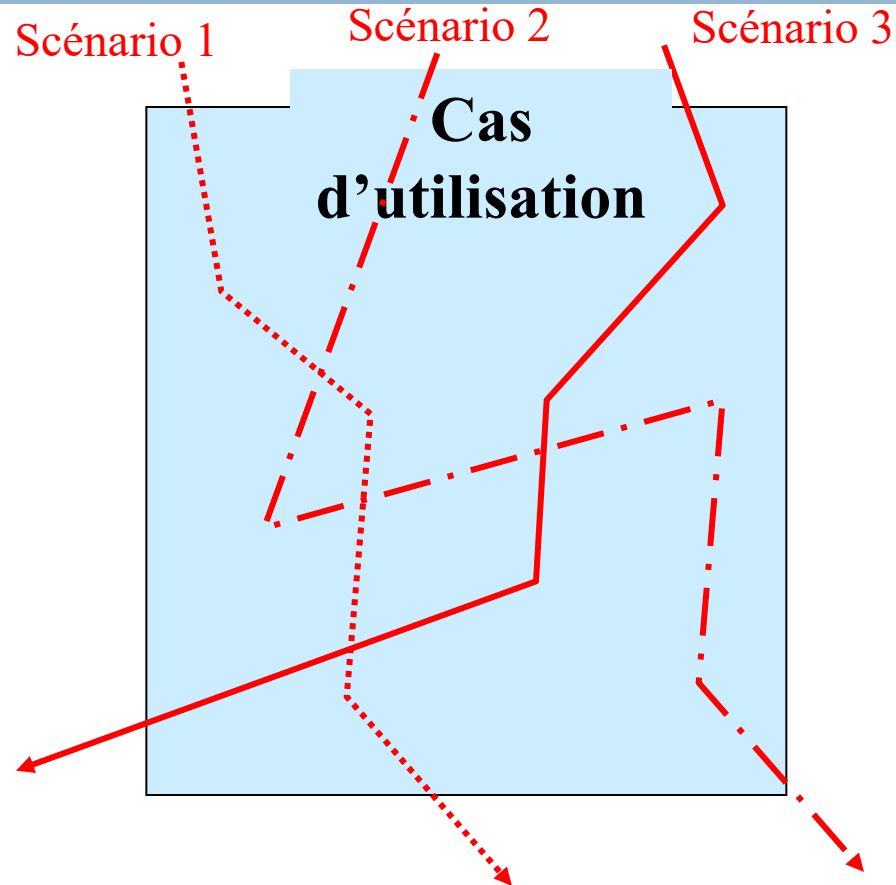
24

- Les cas d'utilisation doivent être vus comme des classes dont les instances sont les scénarios.
- Chaque fois qu'un acteur interagit avec le système, le cas d'utilisation instancie un scénario; ce scénario correspond au flot de messages échangés par les objets durant l'interaction particulière qui correspond au scénario.



# Cas d'utilisation et scénarios

25

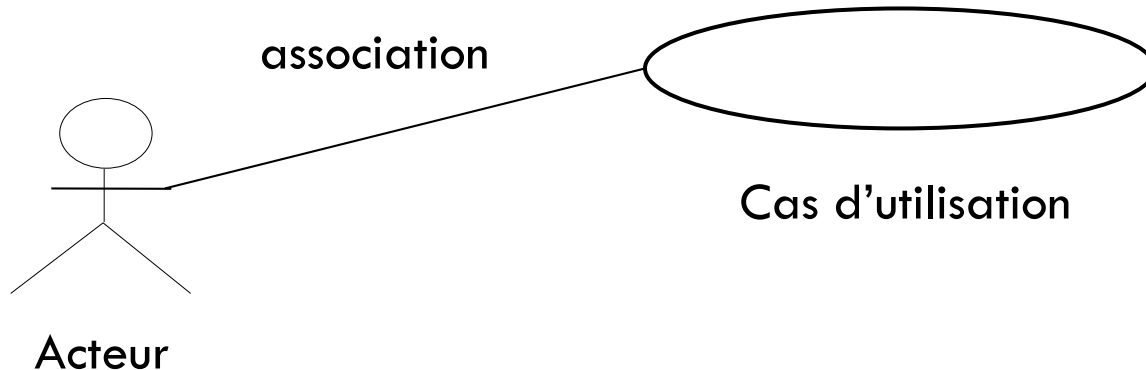


Un cas d'utilisation décrit, de manière abstraite, une famille de scénarios.

# Les associations

26

- Les acteurs impliqués dans un cas d'utilisation lui sont liés par une association.
- Cette association est représentée par une ligne.

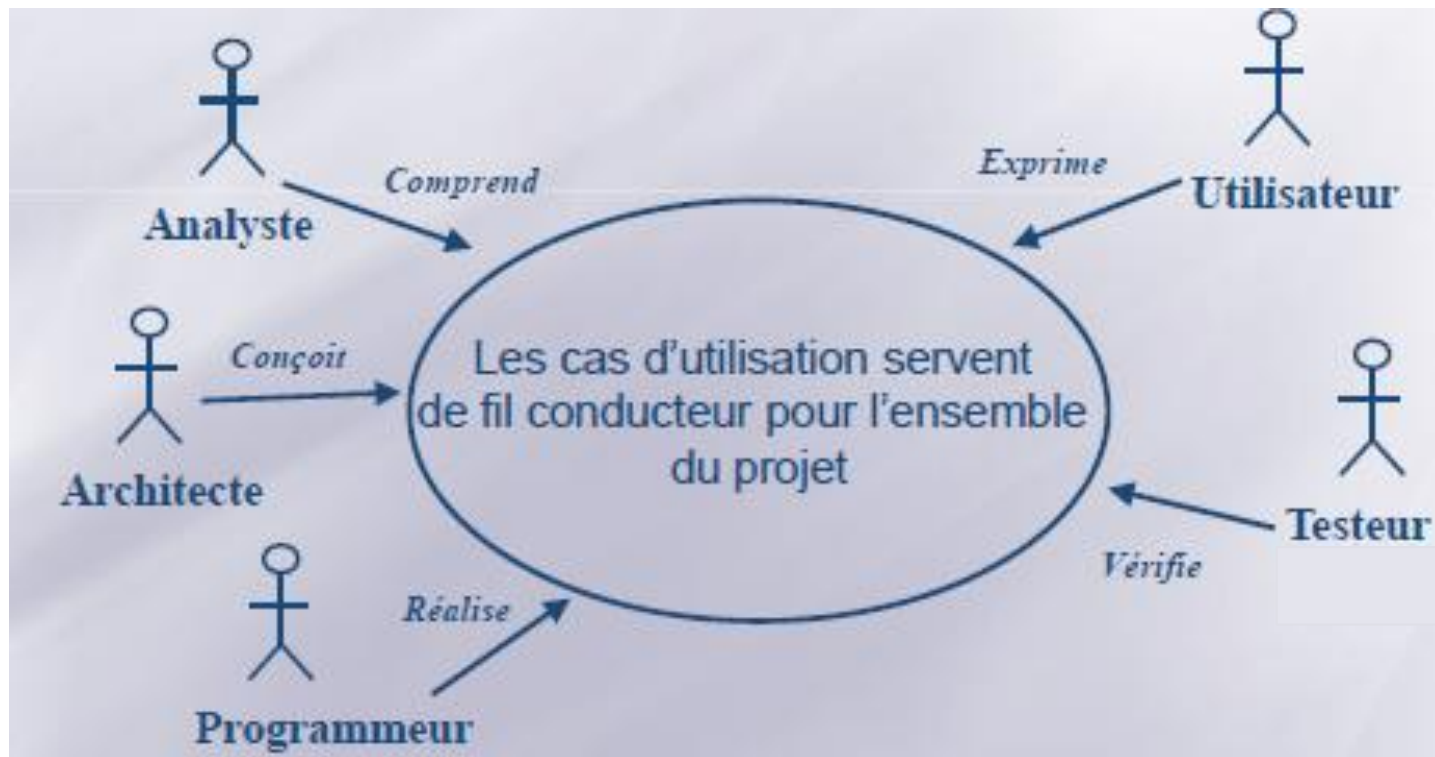


**Relation entre un cas d'utilisation et un acteur**

# Les associations (Suite)

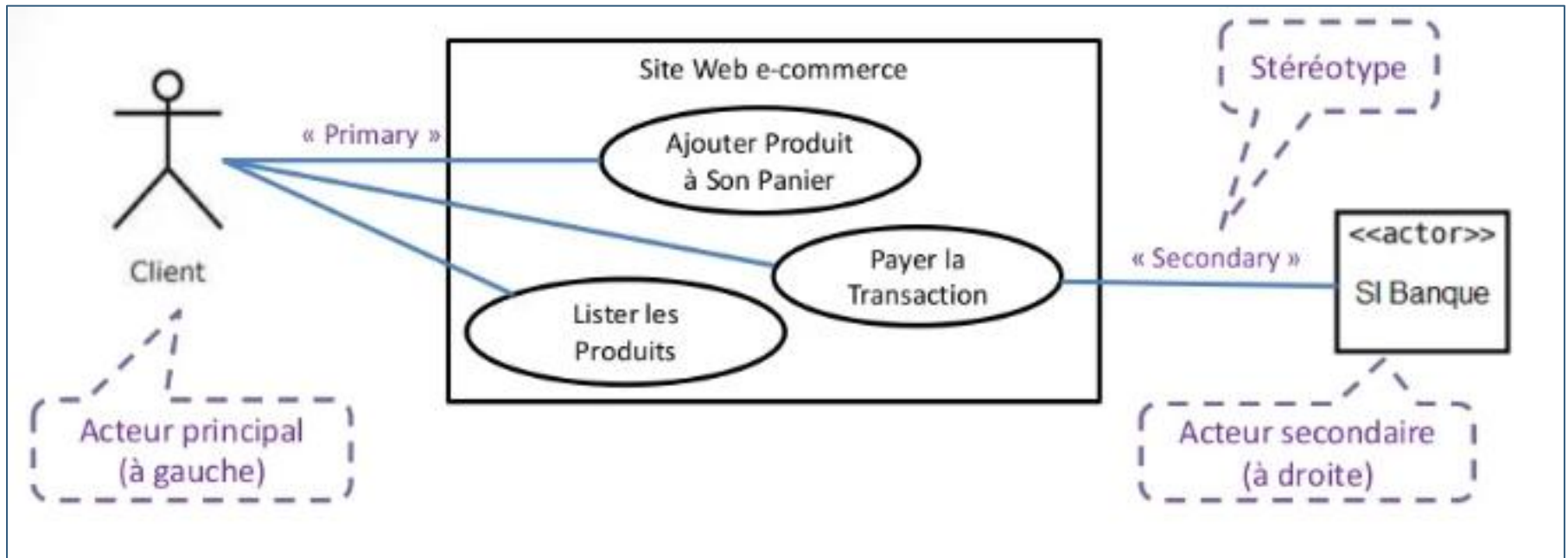
27

Les cas d'utilisation interviennent tout au long du cycle de vie d'un projet



# Les associations (Suite)

28



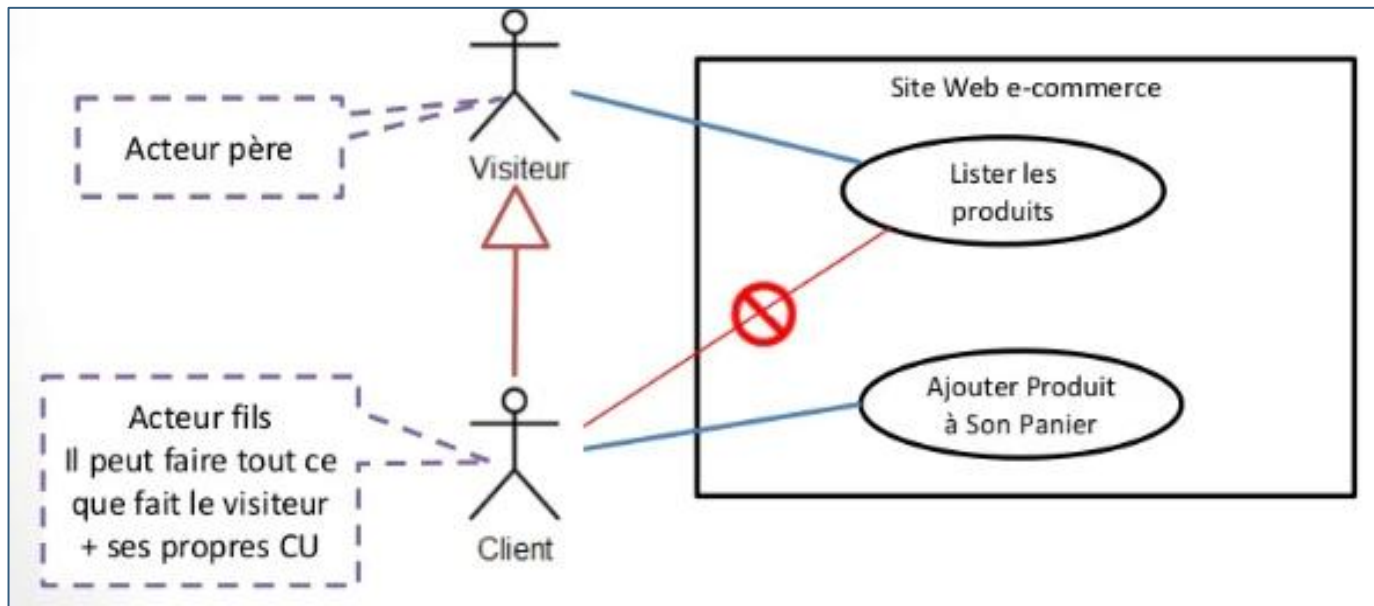
**Acteur principal et acteur secondaire**

# Les associations (Suite)

29

## Les acteurs : la hiérarchie

- La seule relation qui peut exister entre les acteurs est la relation de **généralisation**.
- L'acteur qui **hérite** peut faire tout ce que l'acteur père peut faire.



# Les associations (Suite)

30

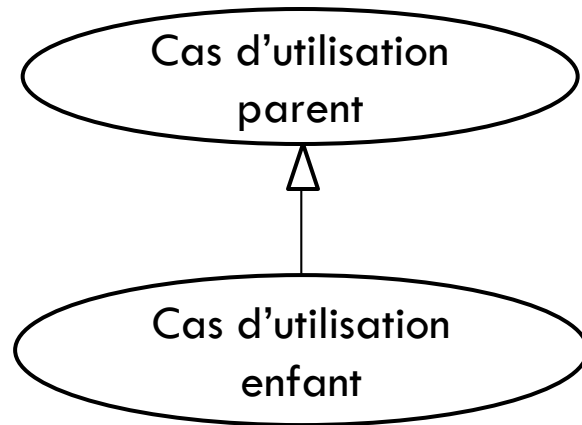
- UML propose trois types de relations entre les cas d'utilisation :
  - ▣ **La généralisation** : permet de formaliser les variations sur le même CU.
  - ▣ **L'inclusion** : permet d'inclure un CU dans un autre.
  - ▣ **L'extension** : permet à un CU d'incorporer à un endroit précis, un autre CU optionnel.

# Les associations (Suite)

31

## □ Relation de Généralisation :

Un cas A est une généralisation d'un cas B si B est un cas particulier de A («une sorte de»). On le représente par une flèche à extrémité triangulaire.

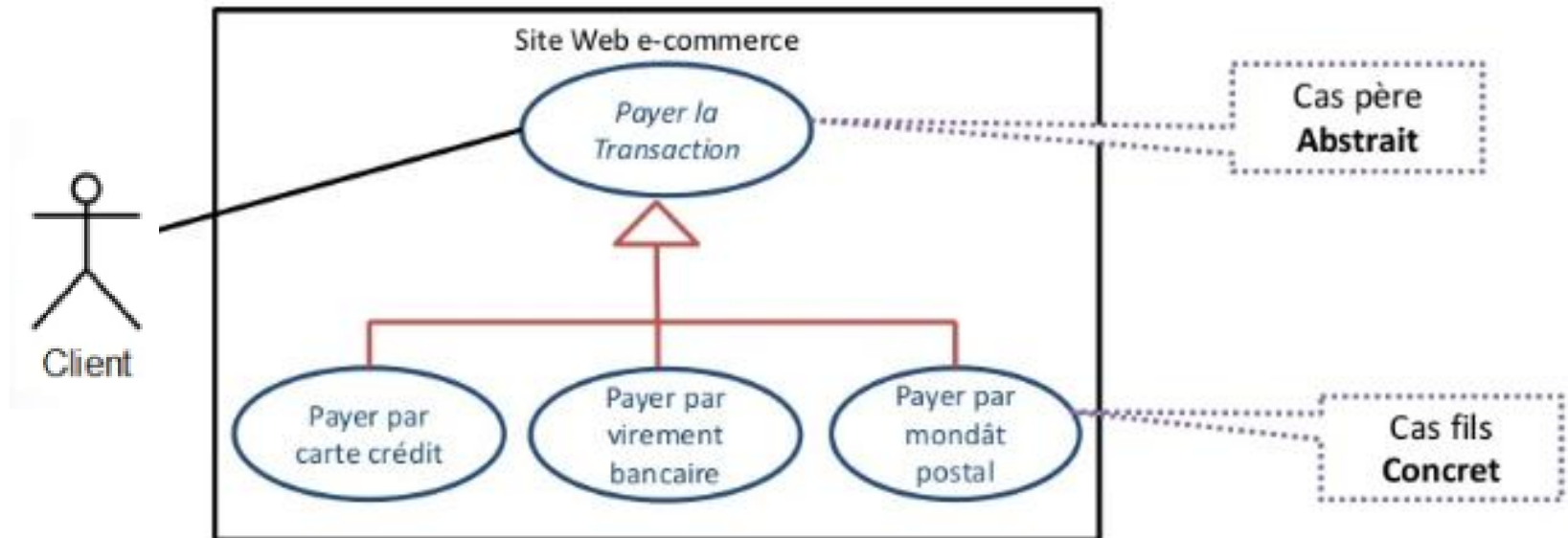


Représentation d'une relation de généralisation

# Les associations (Suite)

32

- ❑ La généralisation permet d'exprimer un comportement commun entre plusieurs cas d'utilisation.
- ❑ Le cas père ne s'instancie que par le biais de l'un de ces descendants.



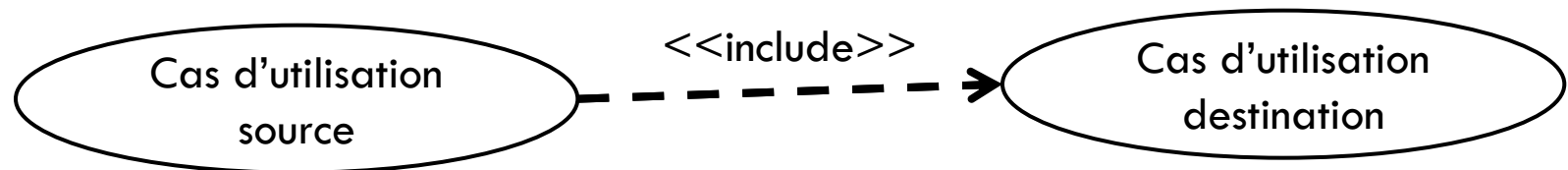


# Les associations (Suite)

33

## □ Relation d'Inclusion :

Un cas A inclut un cas B si le comportement décrit par le cas A inclut le comportement du cas B. Lorsque A est sollicité, B l'est obligatoirement, comme une partie de A. Cette relation est représentée par une flèche en traits pointillés et par le mot clé `<<include>>` placé sur la flèche.



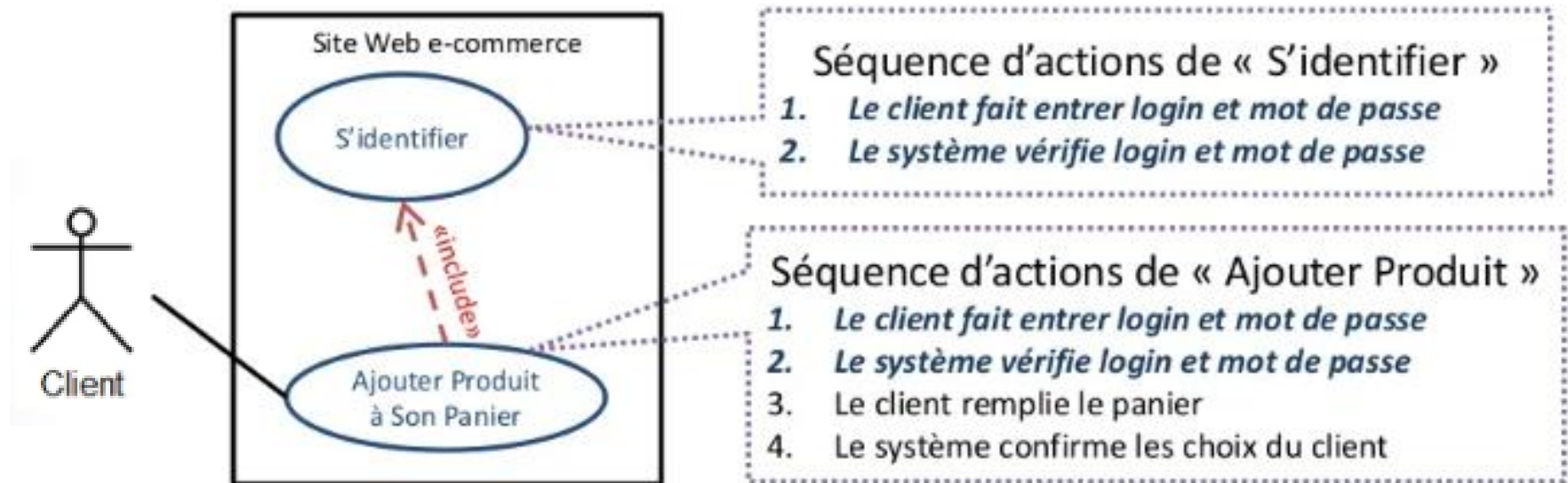
Représentation d'une relation d'inclusion

# Les associations (Suite)

34

## □ Exemple 1 :

Les actions et les interactions réalisés entre l'acteur et le système dans le cas inclus sont dupliqués entièrement dans le cas de base.



# Les associations (Suite)

35

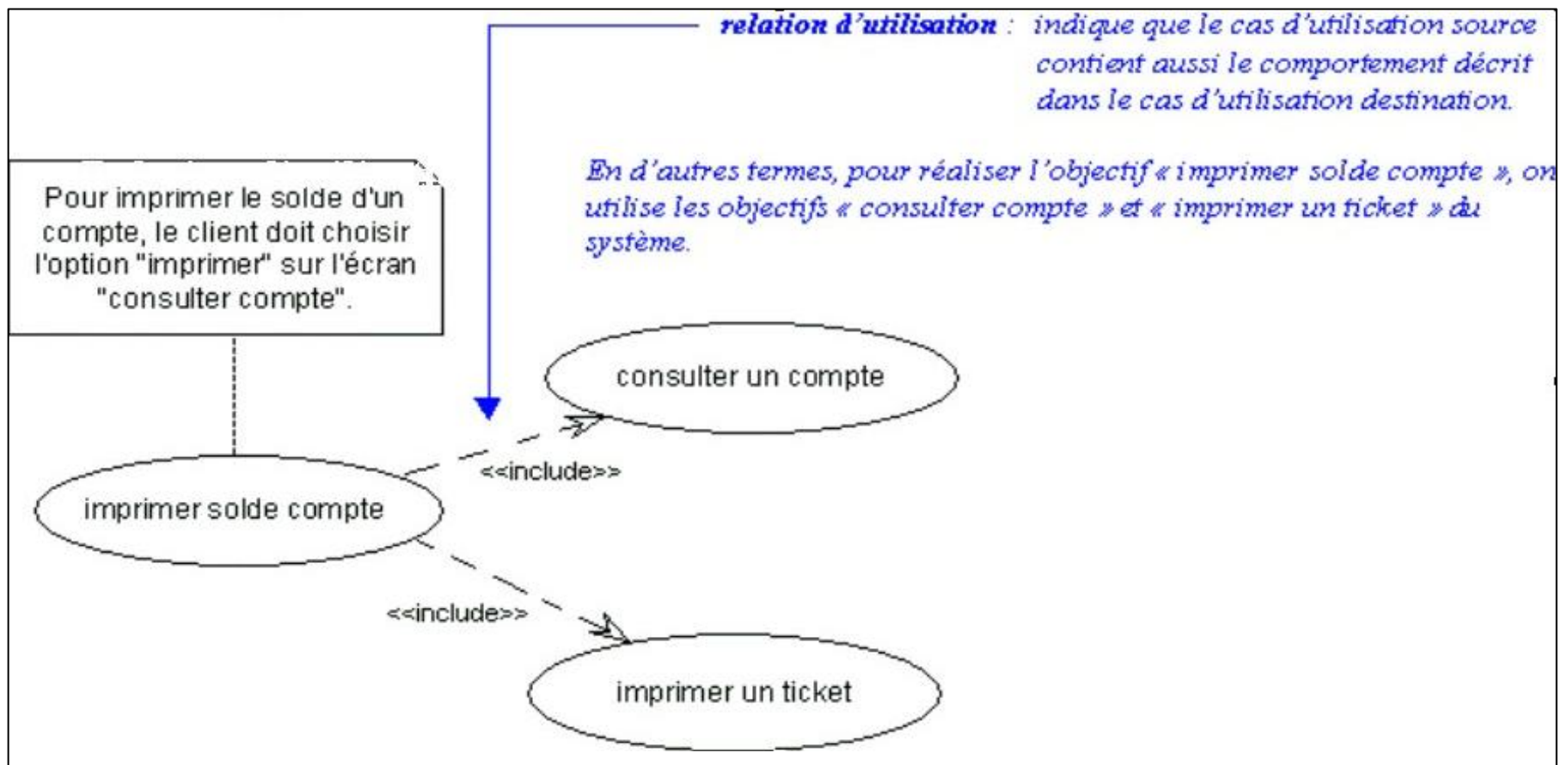
## □ Exemple 2 :

Si on considère le fait qu'un client peut **imprimer le solde** de son compte et **imprimer le ticket**. Quelle relation peut exister entre ce cas et le cas **consulter solde compte** ?

# Les associations (Suite)

36

## Exemple 2 :

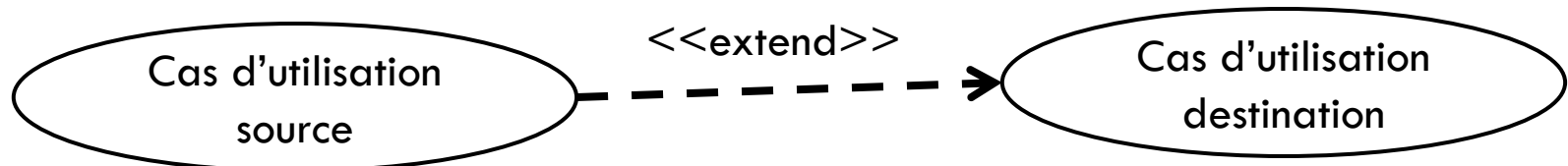


# Les associations (Suite)

37

## □ Relation d'Extension :

Si le comportement de A peut être étendu par le comportement de B, on dit alors que B étend A. Exécuter A peut éventuellement entraîner l'exécution de B. Une extension est souvent soumise à condition. Contrairement à l'inclusion, l'extension est **optionnelle**. Cette relation est représentée par une flèche en traits pointillés et par le mot clé <<extend>> placé sur la flèche.

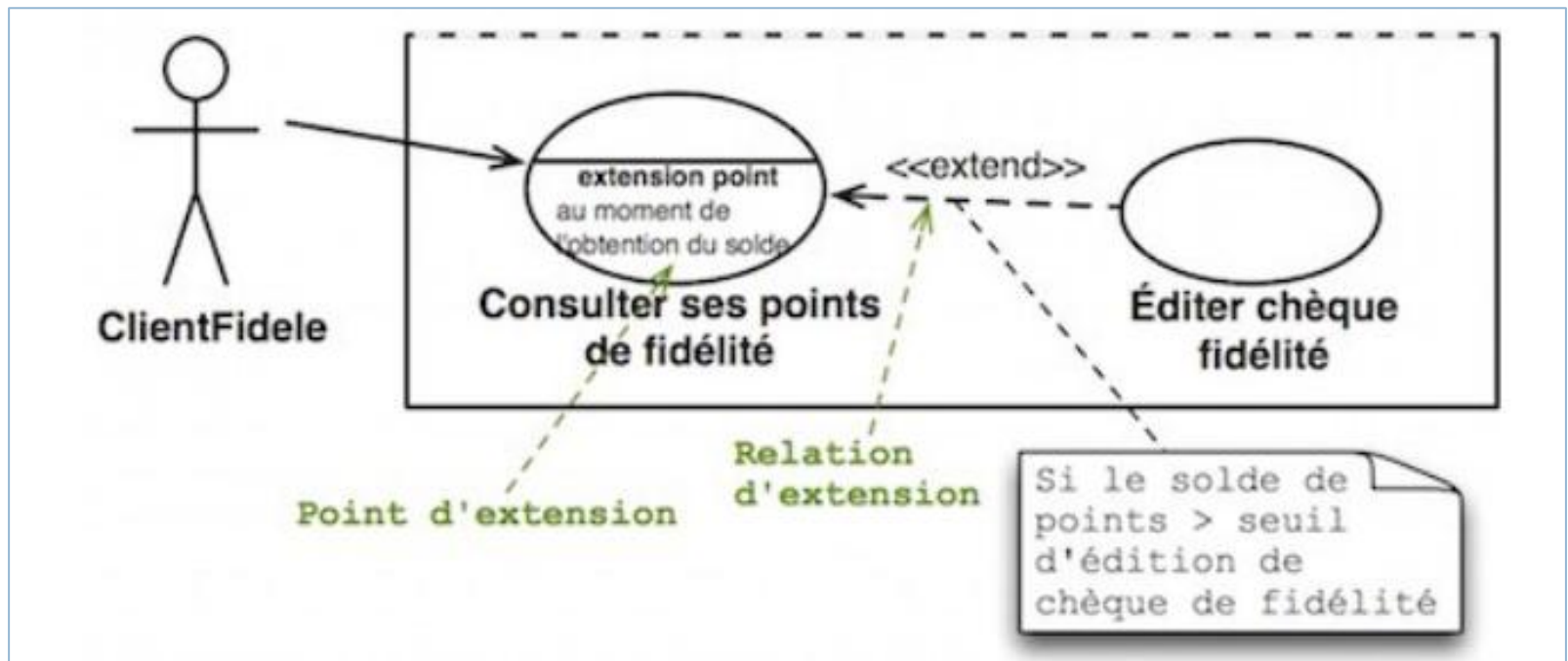


Représentation d'une relation d'extension

# Les associations (Suite)

## □ Exemple :

Un client muni d'une carte de fidélité consulte son solde de points de fidélité et peut, selon une règle établie, éditer un chèque de fidélité.



# Description textuelle

- 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 et c'est l'objectif de *la description textuelle*.
- Généralement la description textuelle comprend trois parties :

# Description textuelle (suite)

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

**Nom** : utiliser un verbe à l'infinitif suivi d'un complément.

**Objectif** : une description résumée du cas d'utilisation.

**Acteurs principaux** : ceux qui vont réaliser le 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.



# Description textuelle (suite)

La deuxième contient toujours une séquence nominale qui décrit de déroulement normal du cas.

**Les pré conditions** : 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 scénarios alternatifs qui sont les variantes du scénario nominal et enfin les scénarios d'exception qui décrivent les cas d'erreurs.

**Des post conditions** : elles décrivent l'état du système à l'issue des différents scénarios.

# Description textuelle (suite)

La troisième partie de la description d'un cas d'utilisation est une rubrique optionnelle. Elle contient généralement des spécifications non fonctionnelles (spécifications techniques...). Elle peut éventuellement contenir une description des besoins en termes d'interface graphique.

# Description textuelle (suite)

<b>Nom du CU</b>	Payer par carte de crédit
<b>Acteurs</b>	Client (principal) , système bancaire (secondaire), service facturation (secondaire)
<b>Résumé</b>	Le client choisit de payer sa commande par carte de crédit
<b>Préconditions</b>	Commande établie par le même client Session du client toujours active (déjà authentifié)
<b>Scénario Nominal</b>	<ol style="list-style-type: none"> <li>1. Le Client choisit de payer par carte de crédit</li> <li>2. Le système présente un formulaire de saisie</li> <li>3. Le client saisie ses paramètres et remplit le formulaire</li> <li>4. Le système vérifie les informations avec le système bancaire du client</li> <li>5. Le système bancaire du client réalise la transaction</li> <li>6. Le système bancaire du client informe le système que la transaction est réussie</li> <li>7. Le système informe le client par email et informe le service facturation</li> <li>8. Le client confirme qu'il a été informé</li> </ol>
<b>Enchainements Alternatifs</b>	<p>A1. informations saisies temporairement erronés : <i>démarre au point 3 du scénario nominal</i></p> <p>4.1. Le système détecte des erreurs de saisies</p> <p>4.2. Le système informe le client ; <i>scénario nominal reprend au point 2</i></p>
<b>Enchainements d'Erreurs</b>	<p>E1. solde insuffisant : <i>démarre au point 4 du scénario nominal</i></p> <p>5.1. Le système bancaire du client interdit la transaction (solde insuffisant)</p> <p>5.2. Le système bancaire du client informe le système de l'échec de la transaction (solde insuffisant)</p> <p>5.3. Le système annule l'opération de paiement et informe le client que son solde est insuffisant</p> <p>5.4. Le système revient à la page de la gestion du panier du client ; <i>le CU se termine en échec.</i></p>
<b>Post-conditions</b>	<p>Compte bancaire du système crédité</p> <p>La transaction est enregistrée</p> <p>Un nouveau message de facturation est enregistré</p>

# Exercice

- Un client peut consulter son compte soit directement dans la banque soit sur internet. Il peut retirer l'argent après avoir s'authentifier. Comme il peut effectuer des virements qui nécessitent une vérification du solde si le montant dépasse 500 DH.

# Solution

45

Borne interactive d'une banque

