




# Génie Logiciel - Cas d'utilisation

## Sommaire

<b>CAS D'UTILISATION .....</b>	<b>6</b>
<b>CAS D'UTILISATION .....</b>	<b>7</b>
<i>Cas d'utilisation et méthode UP.....</i>	<i>8</i>
<i>Cas d'utilisation : Objectifs.....</i>	<i>11</i>
<i>Cas d'utilisation : Objectifs.. Détail.....</i>	<i>12</i>
<i>Cas d'utilisation : Objectifs.. Détail.....</i>	<i>13</i>
<i>Cas d'utilisation : Objectifs.. Détail.....</i>	<i>14</i>
<i>Cas d'utilisation et méthodes Agiles .....</i>	<i>15</i>
<i>Voir Mini-projet et Plan d'itérations.....</i>	<i>16</i>
<i>Quelques définitions préalables .....</i>	<i>17</i>
<i>La variante «Rebecca Wirfs-Brock».....</i>	<i>21</i>
<i>Rédaction d'un cas d'utilisation.....</i>	<i>23</i>
<i>Extensions : les scénarios d'échec - ou variantes -</i>	<i>32</i>

<i>Extensions : les scénarios de variantes .....</i>	<i>35</i>
<i>Exercice des bougies .....</i>	<i>37</i>
<i>Diagrammes de cas d'utilisation.....</i>	<i>38</i>
<i>Remarques sur les acteurs.....</i>	<i>43</i>
<i>Relations entre cas d'utilisation <math>\Rightarrow</math> include, extends &amp; généralise .....</i>	<i>44</i>
<i>Relation include.....</i>	<i>45</i>
<i>Relation «généralise».....</i>	<i>48</i>
<i><math>\Rightarrow</math> Représentation d'un «SOIT-SOIT».....</i>	<i>48</i>
<i>La spécialisation s'applique également aux acteurs !</i>	<i>54</i>
<i>Relation extends <math>\Rightarrow</math> Cas d'utilisation optionnels</i>	<i>63</i>
<i>Exercice « Gestion de projet » .....</i>	<i>67</i>
<i>Comment définir le modèle des cas d'utilisation ? démarche générale</i>	<i>68</i>
	<i>En PRELIMINAIRE : qu'est-ce qu'un bon cas d'utilisation?</i>
	<i>69</i>



**Règle : Se conformer au principe du processus métier élémentaire** 74

<i>La démarche de l'analyste.....</i>	<i>77</i>
<i>Etape 1 : Délimiter les frontières du système à développer</i>	<i>79</i>
<i>Etape 2 : Identifier les acteurs principaux et les acteurs auxiliaires</i>	<i>82</i>
<i>Etape 3 : Identifier les cas d'utilisation ....</i>	<i>83</i>
<i>Quelques recommandations .....</i>	<i>89</i>



*Rédaction des étapes .....* 90



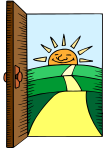
*Rédaction des étapes*



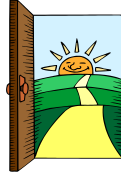
91



*Rédaction des étapes ..... 92*



*Oublier les interfaces utilisateurs 93*



*Oublier les interfaces utilisateurs 94*

*Le niveau de détail..... 97*

## *Cas d'utilisation*

### Objectif

- Identifier et enregistrer les **besoins fonctionnels** du système

~~Objet~~

○ *C*omment ?

⇒ En racontant la manière dont on va utiliser le système



*Du point de vue de l'utilisateur final !*

# Cas d'utilisation

## Historique

- 1986 Ivar Jacobson (un des trois «amigos»)



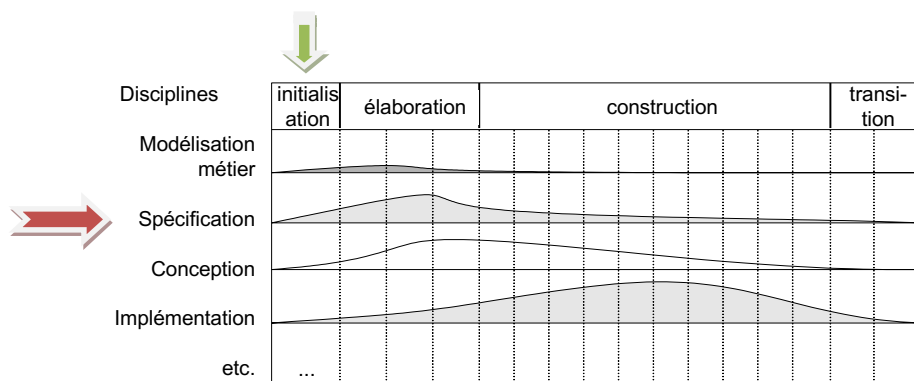
- **Puis adopté** et documenté par la communauté scientifique

## Bibliographie

- «Rédiger des cas d'utilisation efficaces» Allistair Cockburn
- UML 2 spécification

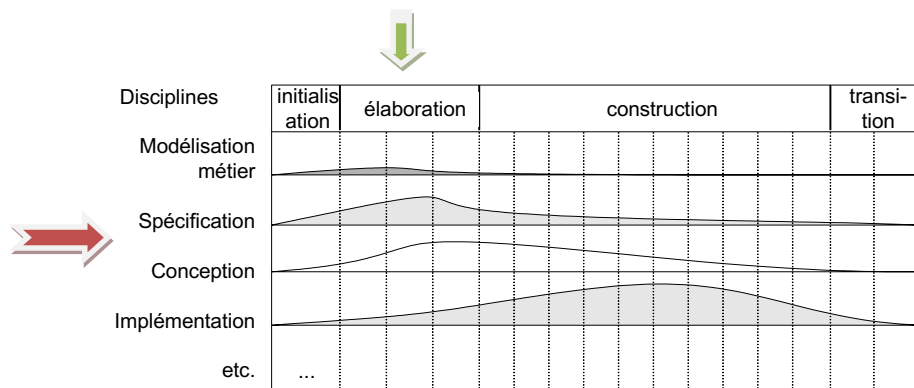
## CAS D'UTILISATION ET MÉTHODE UP

Le modèle des cas d'utilisation fait partie de la discipline **Spécifications**



### ✓ Commence pendant la phase d'initialisation

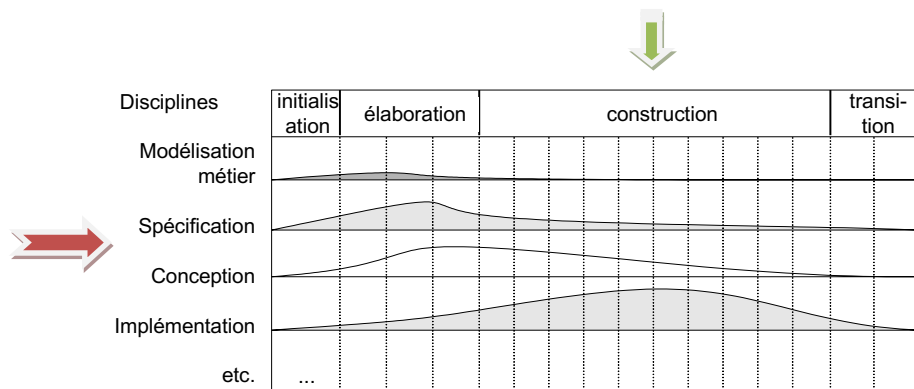
- ⇒ Etablir la liste des acteurs
- ⇒ Etablir la liste des cas d'utilisation
  - *Décris de manière très succincte*
- ⇒ Définition d'un diagramme de cas d'utilisation pour identifier les «frontières du système»



➤ Commence pendant la phase d'initialisation

✓ Pic pendant la phase d'élaboration

- ⇒ Rédaction à proprement parler
- ⇒ Sert de base à la **planification des itérations** de la phase de construction



➤ Commence pendant la phase d'initialisation

➤ Pic pendant la phase d'élaboration

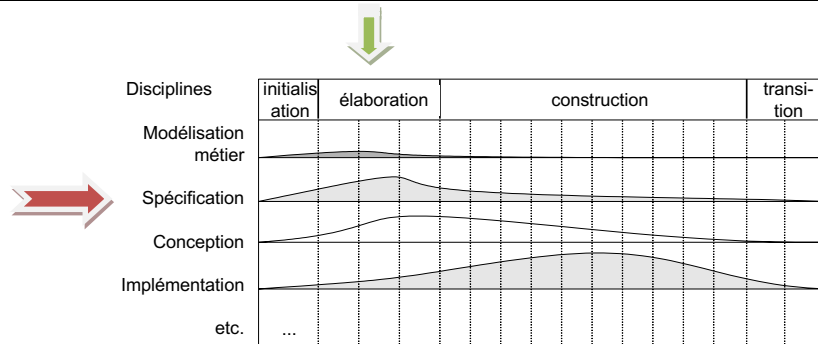
✓ Affinage pendant la phase de construction

- ⇒ Poursuite de la rédaction
  - Rédaction de détails accomplie au départ de l'itération

## CAS D'UTILISATION : OBJECTIFS

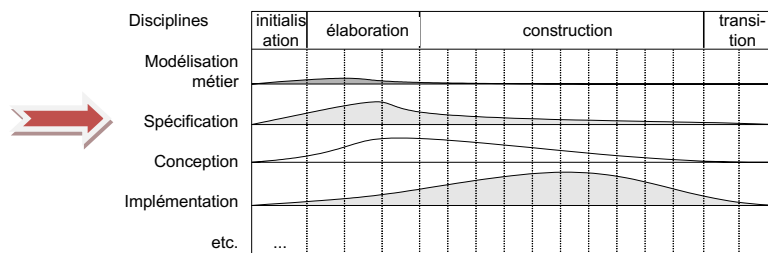
✂ Les cas d'utilisation = élément essentiel de la méthode UP !

UP ⇒ « **Un développement est piloté par les cas d'utilisation** »



⇒ Sert de base à la **planification des itérations**

## CAS D'UTILISATION : OBJECTIFS.. *DETAIL*



### ✓ 1/ Capturer les besoins du client

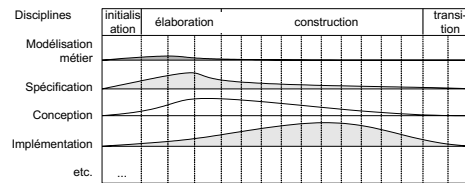
⇒ Cas d'utilisation = **Fonctionnalités** offertes pour répondre aux **besoins du client**

**+** Rajouter : spécifications dites « complémentaires »

⇒ *Définition des contraintes d'utilisation*

○ Temps de réponse désiré, convivialité, etc..

## CAS D'UTILISATION : OBJECTIFS.. *DETAIL*



➤ 1/ Capturer les besoins du client

✓ 2/ Planification et partage du travail

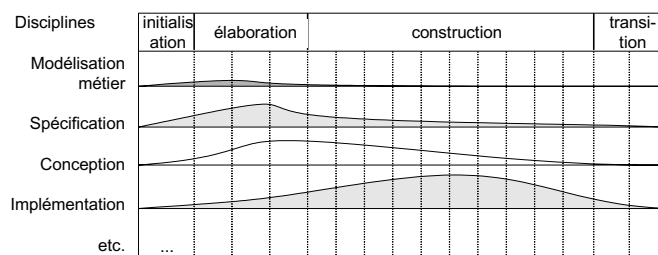
UP ⇒ «Un développement est piloté par les cas d'utilisation»

Objectif d'une itération ?

Par exemple...

- Implémenter un cas d'utilisation au complet
- Implémenter un scénario particulier
- Implémenter des cas d'utilisation simultanément par des équipes différentes

## CAS D'UTILISATION : OBJECTIFS.. *DETAIL*



➤ 1/ Capturer les besoins du client

➤ 2/ Planification et partage du travail

✓ 3/ Manuels d'utilisation

Leur rédaction s'articule essentiellement autour des cas d'utilisation

## CAS D'UTILISATION ET MÉTHODES AGILES

### XP, Scrum...

~~Cas d'utilisation~~ ⇒ « **Histoires** » (« stories »)

Une **histoire** = *scénario* d'un cas d'utilisation



**Méthodes pilotées indirectement par les cas d'utilisation**

### Planification en 2 phases

1. Etablir la liste globale des « histoires »
2. Phase de développement  
⇒ Répartir ces différentes histoires entre chacune des itérations



**VOIR MINI-PROJET ET PLAN D'ITERATIONS**



## QUELQUES DÉFINITIONS PRÉALABLES

⇒ Acteur

⇒ Scénario

⇒ Cas d'utilisation

**Acteur** ⇒ Scénario ⇒ Cas d'utilisation ⇒ Diagramme de cas d'utilisation

Une entité - **extérieure** - qui interagit avec le système

P.e. :

- **Une personne physique**

Une même personne peut jouer plusieurs **rôles**..



*Chacun de ces rôles donnera lieu à un acteur différent !*

*Exemple : GAPS*



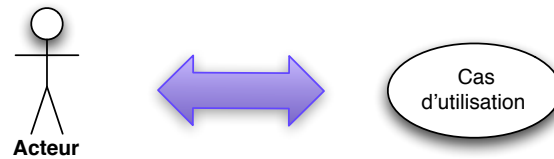
Les **droits** d'un applicatif sont définis au niveau des **rôles**

- **Un appareil (extérieur au système à analyser)**

- Un système informatique connecté par Internet
- Un capteur de signaux
- Un serveur de base de données
- etc.

Acteur ⇨ **Scénario** ⇨ Cas d'utilisation ⇨ Diagramme de cas d'utilisation

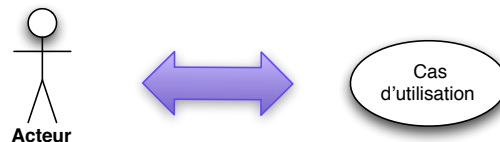
Une **suite d'actions** et d'interactions entre les **acteurs** et le **système** à analyser



Traiter une vente

### Scénario principal

1. Un client arrive à la caisse avec différents articles
2. Le caissier enregistre chaque article acheté
3. Le client présente sa carte de crédit et s'authentifie
4. Etc.



### Scénario principal :

1. Un client arrive à la caisse avec différents articles
2. Le caissier enregistre chaque article acheté
3. Le client présente sa carte de crédit et s'authentifie
4. Etc.

### Scénario ⇨

- Une «**histoire**» **possible** se déroulant entre un acteur et le système
- **Plusieurs scénarios** nécessaires pour décrire une **fonctionnalité spécifique** du système

*Exemple* : Traitement d'une vente

- **Scénario 1** : la transaction réussie
- **Scénario 2** : elle échoue (carte de crédit du client échue)

## LA VARIANTE «REBECCA WIRFS-BROCK»

Action d'acteur	Action système
1. Un <b>client</b> se présente à la caisse avec des produits à acheter.	
2. Le <b>caissier</b> saisit l'identificateur de chaque produit. S'il y a plus d'une pièce d'un même produit, le <b>caissier</b> a la possibilité d'en saisir le nombre.	3. La <b>caisse</b> détermine le produit, et en affiche la description et le prix unitaire.
3. Après avoir saisi tous les produits, le <b>caissier</b> signale à la caisse la fin des achats.	5. La <b>caisse</b> calcule et affiche le total.
1. Le <b>caissier</b> annonce le total au <b>client</b> .	
2. Le <b>client</b> effectue son paiement.	
3. Le <b>caissier</b> saisit la somme d'argent payée par le <b>client</b> .	9. La <b>caisse</b> affiche le solde dû au client. La <b>caisse</b> imprime un ticket.
10. Le <b>caissier</b> encaisse le paiement, et extrait le solde dû du tiroir-caisse. Le <b>caissier</b> rend la monnaie au <b>client</b> , et lui donne le ticket. Le <b>client</b> quitte la caisse avec les produits achetés	11. La <b>caisse</b> enregistre l'achat.

Acteur ⇨ Scénario ⇨ **Cas d'utilisation** ⇨ Diagramme de cas d'utilisation

### ⇨ Ensemble de scénarios

- scénarios de succès
- + scénarios d'échecs

Traiter une vente

#### Scénario principal (succès):

1. Un client arrive à la caisse avec différents articles
2. Le caissier enregistre chaque article acheté
3. Le client présente sa carte de crédit et s'authentifie
4. etc.

#### Autres scénarios (échec):

Opération 3 : Si l'autorisation est rejetée

1. le caissier en est informé
2. le caissier demande une autre forme de paiement

## REDACTION D'UN CAS D'UTILISATION

### La philosophie de la «boîte noire»

⇒ Le système n'est décrit qu'en termes de **responsabilités**

- Ce que fera le système (le «**QUOI**»)
- Sans décrire **comment** il le fera

### Du bon..

⇒ «Le système enregistre la vente»

### Du mauvais..

⇒ «Le système enregistre la vente dans une base de données MySQL»

### Ou pire encore..

⇒ «Le système envoie une commande SQL «insert» pour enregistrer la vente»

## Rédaction d'un cas d'utilisation : quel niveau de détail?

⇒ Style Résumé

⇒ Style Standard

⇒ Style détaillé

*Quel niveau*



⇒ Cela dépend du stade d'avancement au sein de la phase d'analyse..

**Style Résumé** ⇨ [Style Standard](#) ⇨ [Style détaillé](#)

⇨ Se contenter de présenter le scénario principal...

Un texte informel suffit 😊

Traiter une vente

*Un client arrive à la caisse avec les articles qu'il désire acheter. Le caissier enregistre chacun des articles en utilisant le système à développer. Ce dernier affiche alors le détail des achats effectués ainsi que le total. Le client effectue son paiement au moyen d'une carte de crédit. Le paiement est validé par le système puis enregistré. Le système opère une mise à jour de l'inventaire et imprime un reçu pour le client.*

[Style Résumé](#) ⇨ **Style Standard** ⇨ [Style détaillé](#)

⇨ Description des différents scénarios qui composent le cas d'utilisation

Traiter une vente

**Scénario principal (succès):**

1. Un client arrive à la caisse avec différents articles
2. Le caissier enregistre chaque article acheté
3. Le client présente sa carte de crédit et s'authentifie
4. Etc.

**Autres scénarios (échecs ou variantes):**

Opération 3 : *Si l'autorisation est rejetée*

1. Le caissier en est informé
2. Le caissier demande une autre forme de paiement

Style Résumé ⇒ Style Standard ⇒ **Style détaillé**

⇒ Préface

⇒ Liste des parties prenantes et des intérêts

⇒ Pré-conditions (garanties de succès)

⇒ Post-conditions

⇒ Scénario principal

⇒ Scénarios d'échec

⇒ Spécifications particulières

Préface ⇒ **Liste des parties prenantes et des intérêts**  
⇒ Pré/post-conditions ⇒ Scénarios ⇒ Spécifications particulières

Espèce de contrat établi entre le système et les parties prenantes

- Enumérer tous les acteurs concernés
- Préciser les responsabilités du système pour chacun d'eux

*Exemple : Une vente*

⇒ 2 acteurs : **Caissier** et **Vendeur**

⇒ Quels sont leurs intérêts ?

**Le caissier**

Désire un moyen efficace de saisie, à même de lui garantir aucune erreur de paiement (les différences de caisse sont retenues sur son salaire..).

**Le vendeur**

Désire une mise à jour de ses commissions sur les ventes.


Préface ⇒ Liste des parties prenantes et des intérêts

⇒ **Pré/post-conditions** ⇒ Scénarios ⇒ Spécifications particulières

Origine scientifique : **Bertrand Meyer**, langage objet **Eiffel**


### 1. Pré-conditions

Ce qui doit toujours être vrai avant le début d'un scénario pour que ce dernier puisse s'accomplir correctement.

 Ces pré-conditions ne sont pas vérifiées par le cas d'utilisation : elles sont tenues pour acquises.

### 2. Post-conditions (garanties de succès)

Ce qui doit toujours être vrai lorsque le scénario se termine avec succès.

 Ces post-conditions sont garanties par le cas d'utilisation et devraient être tenues pour acquises par les autres cas d'utilisation.

## **Pré/post-conditions**

 Voir aussi la programmation des **assertions** en Java.

### *Exemple : Une vente*

#### **Pré-conditions**

Le caissier s'est identifié et il a été authentifié

#### **Post-conditions**

La vente a été enregistrée

Préface ⇒ Liste des parties prenantes et des intérêts

⇒ Pré/post-conditions ⇒ **Scénarios** ⇒ Spécifications particulières

## Le scénario principal

⇒ Décrit le **chemin normal** qui devrait satisfaire les intérêts des parties prenantes.



### Un conseil...

- Décrire la voie standard, « quand tout se passe bien »
- Ne pas indiquer les exceptions ni les variantes

1. Le Client arrive à la caisse avec des marchandises et/ou des services à acheter.
2. Le Caissier démarre une nouvelle vente.
3. Le Caissier entre le code de l'article.
4. Le Système enregistre l'article et présente sa description, son prix et le total en cours. Le prix est calculé à partir d'un ensemble de règles.
5. etc.

Préface ⇒ Liste des parties prenantes et des intérêts

⇒ Pré/post-conditions ⇒ **Scénarios** ⇒ Spécifications particulières

## EXTENSIONS : LES SCENARIOS D'ECHEC - OU VARIANTES -

### Scénario principal

1 ...

2 ...

**3.** Le Caissier entre le code de l'article

4 ...

5 ...

#### 3a. Code invalide

1. Le Système signale l'erreur et rejette la saisie

#### 3b. Il y a plusieurs articles de la même catégorie et leur identification individuelle n'a pas d'importance (exemple : cinq cartons de lait de soja):

1. Le Caissier peut saisir le code de l'article et sa quantité



Un scénario d'échecs peut lui-même avoir des extensions...

*7e. Le Client présente des coupons*

- 1.** Avant de traiter le paiement, le Caissier enregistre chaque coupon et le Système réduit le prix en conséquence. Le Système enregistre les coupons utilisés pour des raisons de comptabilité

**1a. Un coupon ne correspond à aucun article acheté**

1. Le Système signale l'erreur au Caissier.

Les extensions qui peuvent intervenir à tout instant...

.. notées au moyen d'une astérisque (\*)

**\*a.** *À tout moment, si le système tombe en panne*

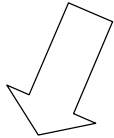
Le Caissier relance le Système, se reconnecte et demande la récupération de l'état précédent.

## EXTENSIONS : LES SCENARIOS DE VARIANTES

### Deux types d'extensions

⇒ **Scénarios d'échec** ⇒ *problème rencontré*

⇒ **Variantes** ⇒ *variante au sein du scénario principal*



### Variantes

- Plus rarement utilisées
- But : alléger le scénario principal
- Condition de déclenchement = condition de déclenchement de la variante.

Préface ⇒ Liste des parties prenantes et des intérêts

⇒ Pré/post-conditions ⇒ Scénarios ⇒ **Spécifications particulières**

Pour modéliser des **spécifications non fonctionnelles**

⇒ **Contraintes** qui se rattachent au cas d'utilisation

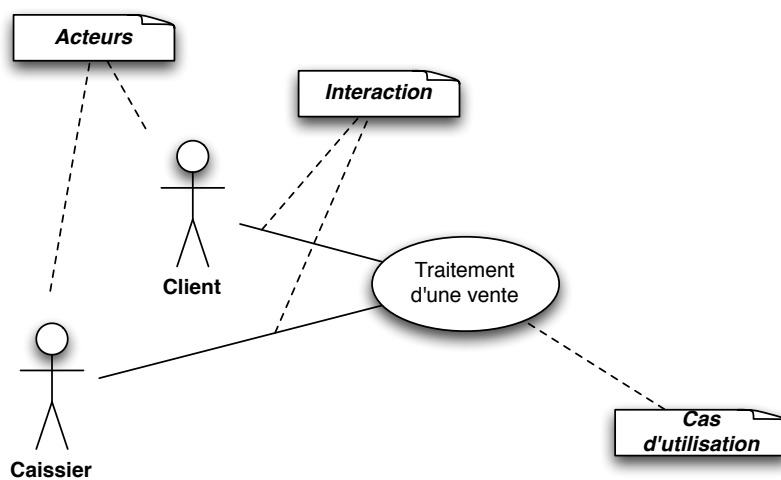
- Interface utilisateur à écran tactile sur grand écran plat...
- Le texte doit être visible à un mètre de distance...
- Obtention de la réponse à la demande d'autorisation en trente secondes au maximum, dans 90 % des cas...
- Récupération fiable quand l'accès à un service distant (inventaire, par exemple) échoue...

## ➡ EXERCICE DES BOUGIES

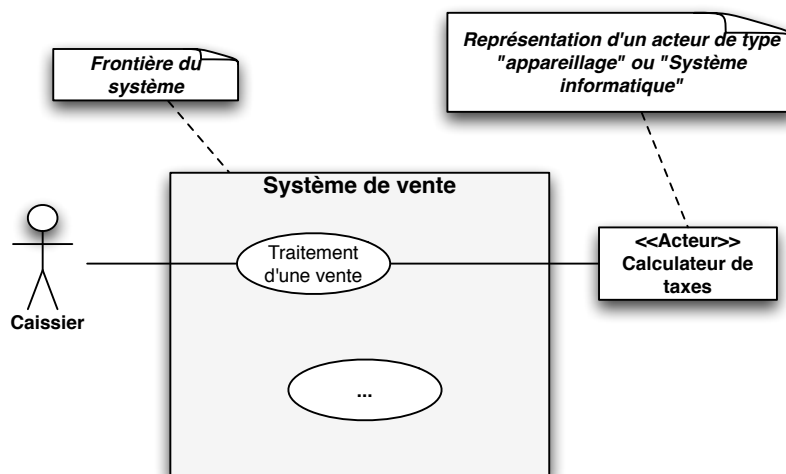
## DIAGRAMMES DE CAS D'UTILISATION

### Représentation graphique UML

**Figure 1: Un diagramme de cas d'utilisation**

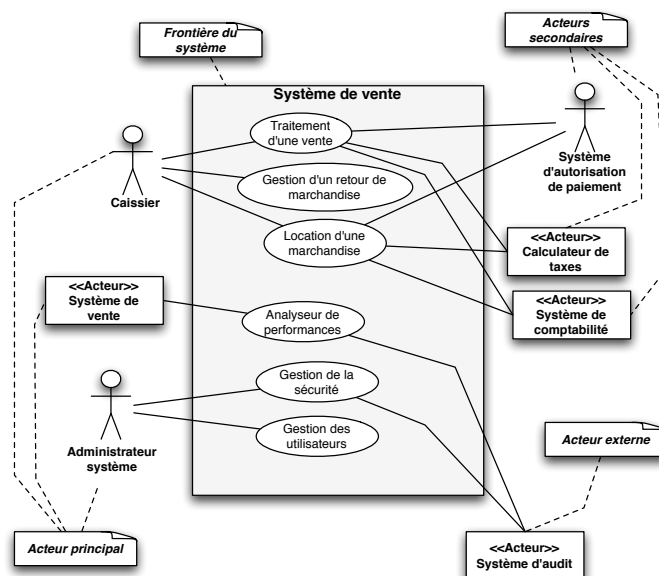


## Représentation des acteurs dans un diagramme de contexte



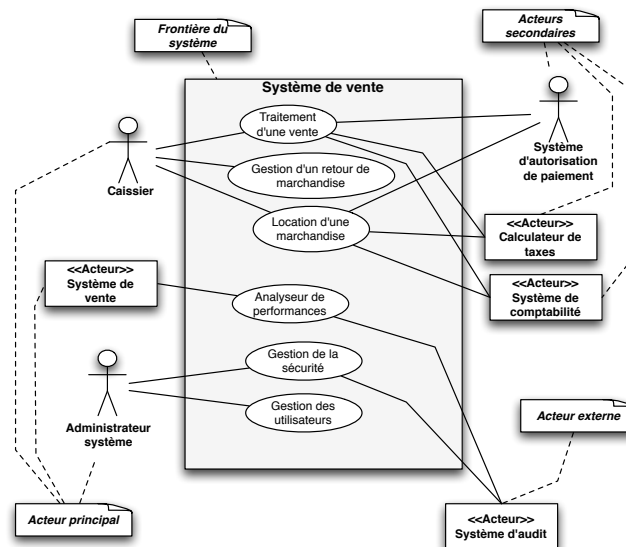
### Les acteurs principaux

- Les **acteurs principaux** attendent des services de la part du système.
- ✗ Ce sont eux qui initient le déclenchement des cas d'utilisation.



## Les acteurs secondaires (ou « acteurs auxiliaires »)

- Sollicités par le système pour aider à accomplir tel ou tel cas d'utilisation
- Un acteur secondaire fournit des services au système



ELS-Mars 2018

## Les acteurs externes

- Concernés simplement par l'accomplissement de tel ou tel cas d'utilisation...
  - Sans être à la source de son accomplissement : ça n'est pas un acteur principal...
  - Sans fournir de service à son accomplissement : ça n'est pas un acteur secondaire.
  - Sans attendre réellement de service, *simplement concerné...*
- Typiquement, un acteur externe à un cas d'utilisation ne fait que de recevoir de l'information relative à son accomplissement

*Exemple d'acteurs externes :*

- Un système externe d'analyse de performance
- Un système externe d'audit

ELS-Mars 2018

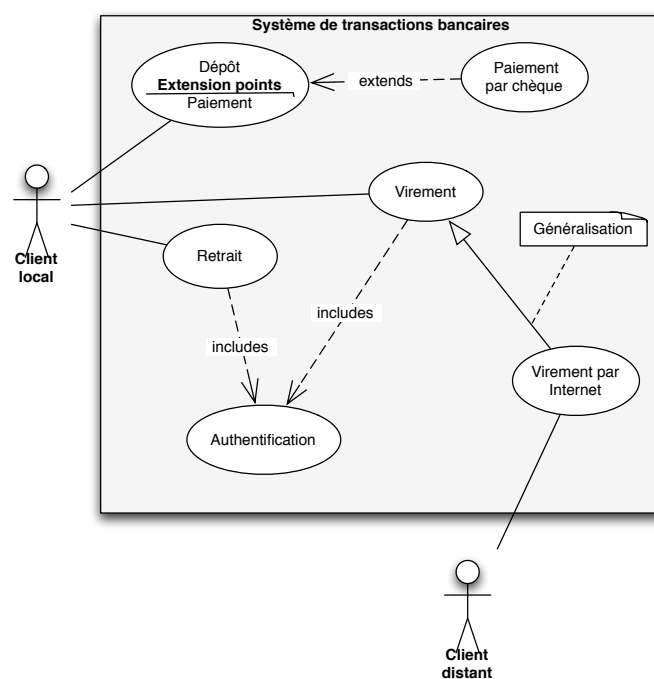
## REMARQUES SUE LES ACTEURS

✍️ Dépendant du cas d'utilisation, un acteur sera parfois..

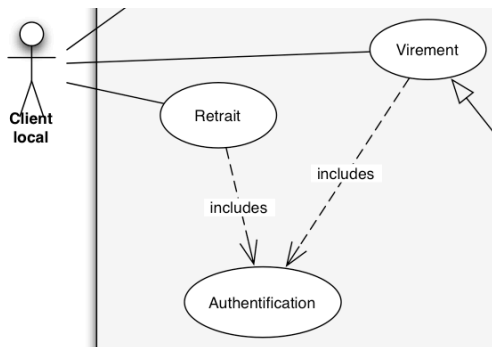
- principal
- secondaire
- externe

✍️ Du code sera spécifiquement développé pour chacun des acteurs

## RELATIONS ENTRE CAS D'UTILISATION ➡ include, extends & généralise



## RELATION INCLUDE



### ○ Factorisation

Une même suite d'étapes apparaît dans plusieurs cas d'utilisation

### ○ Décomposition

Décomposition d'un cas d'utilisation « de base » trop complexe

## Représentation textuelle de la relation **include**

### Illustration: Cas d'utilisation Retrait

#### Retrait

##### Scénario principal :

1. Le client arrive à la caisse
2. Le client s'authentifie : **include** Authentification
3. etc.

#### Authentification

##### Scénario principal :

1. Le client présente sa carte
2. La carte est reconnue par le système
3. Le client saisit son mot de passe
4. etc.

~~✗~~ L'inclusion est en principe **obligatoire !**

**Sinon..**

- Utiliser la relation **extends** (rajouts optionnel)
- Ou la relation **généralise**

*Exemple*

**Etape X:** Le client effectue une transaction, parmi les suivantes:

- **SOIT** Retirer argent
- **SOIT** Déposer argent



**Etape X:** Le client effectue une transaction, parmi les suivantes:

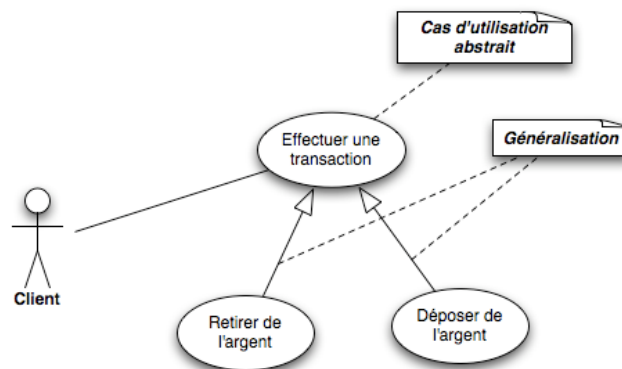
- **Include** RetirerArgent
- **include** DéposerArgent

## RELATION «GÉNÉRALISE»

⇒ REPRÉSENTATION D'UN «**SOIT-SOIT**»

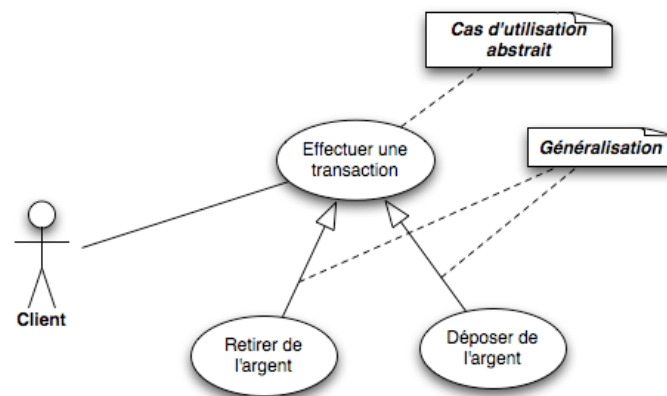
Le client effectue une transaction, parmi les suivantes:

- **SOIT** Retirer argent
- **SOIT** Déposer argent



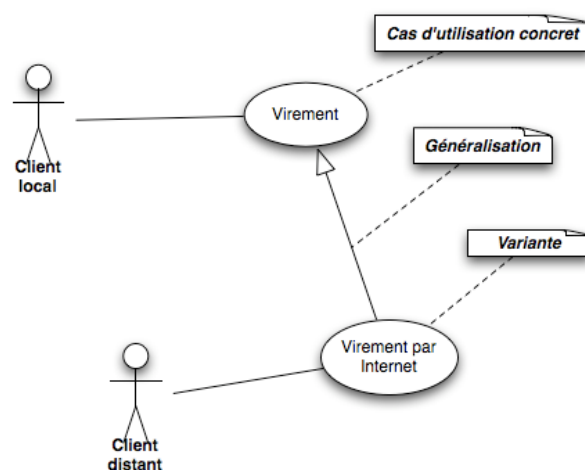


## Représentation d'un «SOIT-SOIT»



- Le cas de base « général » est un cas d'utilisation **abstrait**
- Les cas d'utilisation « enfant » sont de **véritables cas d'utilisation**

## LA SPECIALISATION EN TANT QUE «VARIANTE»

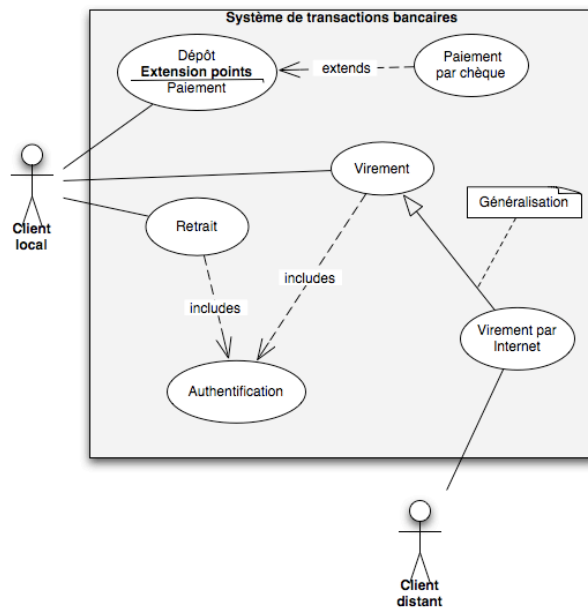


- Virement est un vrai cas d'utilisation **concret**
- Le cas d'utilisation « enfant » Virement par Internet est une **variante** du cas de base.

## ✍ GENERALISATION : REGLES D'UTILISATION

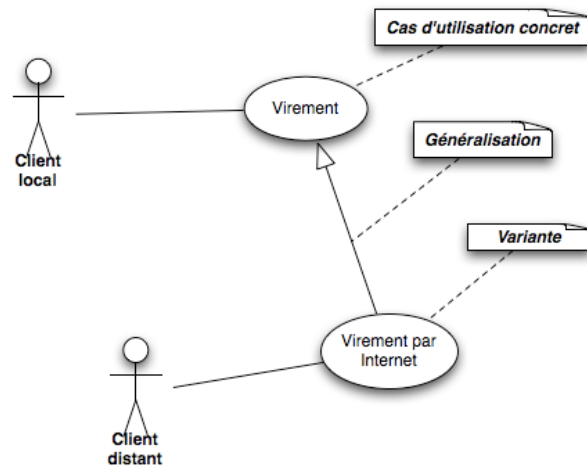
Un cas d'utilisation «enfant» est de même espèce que «parent»

- ⇒ Il interagit avec les mêmes acteurs que le parent (**héritage**)
- ⇒ Il peut lui-même avoir des acteurs spécifiques
- ⇒ Il «**héríte**» également des relations du parent (**includes**, **extends**, ..)



## ✍ GENERALISATION : SUBSTITUTION

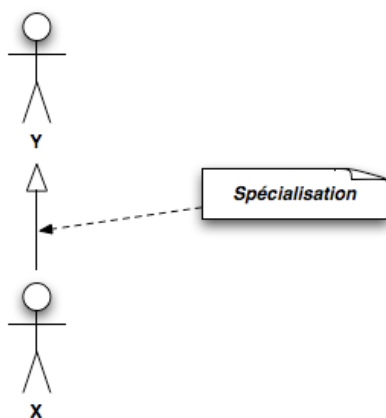
- 😊 Un cas d'utilisation « enfant » peut être **substitué** au cas parent..
- ⇒ Tout ce qui peut être dit au niveau du cas « parent » doit pouvoir être dit au niveau du cas « enfant » (substitution du cas parent par le cas enfant)



⇒ Un client local peut opérer un virement par internet !

## LA SPÉCIALISATION S'APPLIQUE ÉGALEMENT AUX ACTEURS !

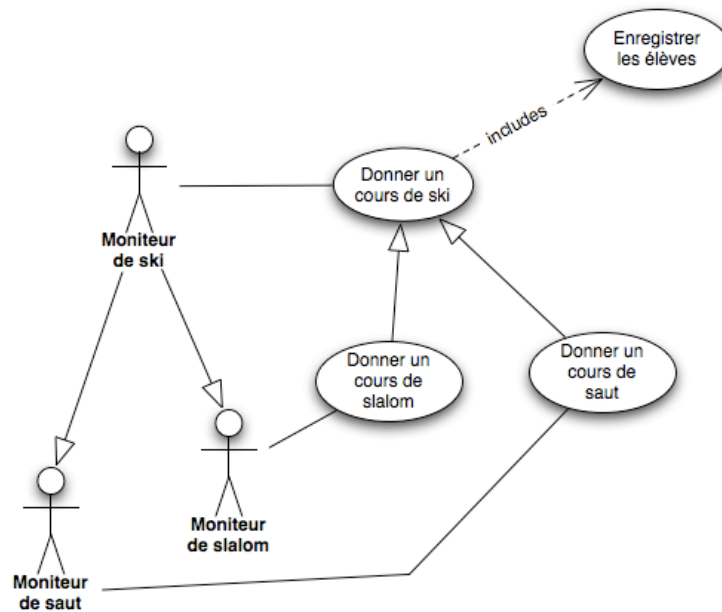
### SPECIALISATION DES ACTEURS



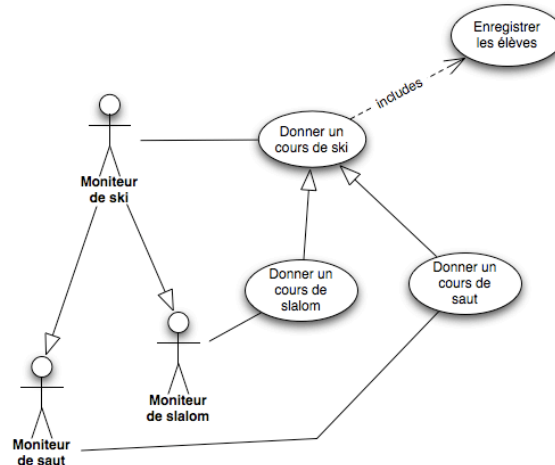
Un acteur X qui spécialise un acteur Y jouit de toutes les interactions attribuées à l'acteur Y

L'acteur X jouit – *en plus* – de certaines interactions qui lui sont propres.

## Exemple : concept valaisan du «Moniteur de ski»



ELS-Mars 2018



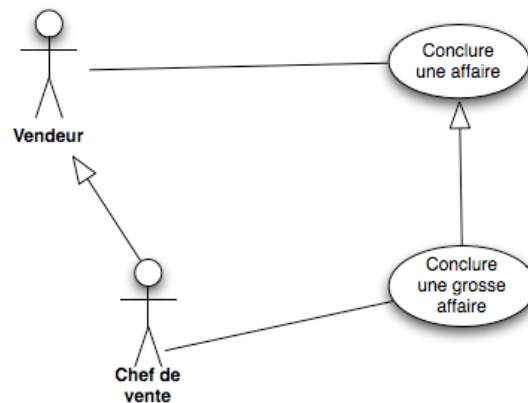
- Un moniteur de slalom **n'est pas un** moniteur de ski
- Un cours de saut **est une** espèce de cours de ski
- Un cours de ski **est soit** un cours de saut, **soit** un cours de slalom
- Un moniteur de ski peut tout enseigner : un moniteur de ski **est un** moniteur de slalom, il **est également un** moniteur de saut

ELS-Mars 2018

## Un deuxième exemple..

Un chef de vente **est-un** vendeur qui a acquis une certaine expérience..

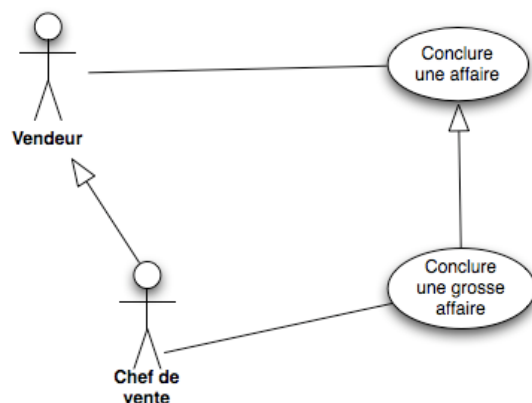
⇒ Il peut conclure une grosse affaire



## Un deuxième exemple..

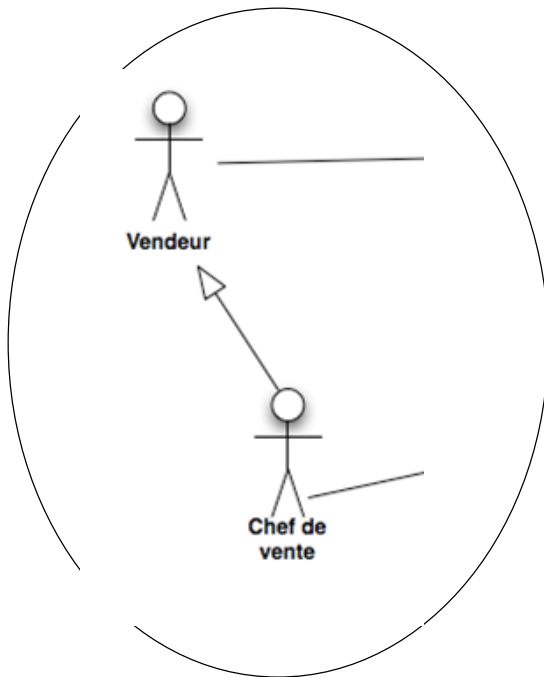
Un chef de vente **est-un** vendeur qui a acquis une certaine expérience..

⇒ Il peut conclure une grosse affaire

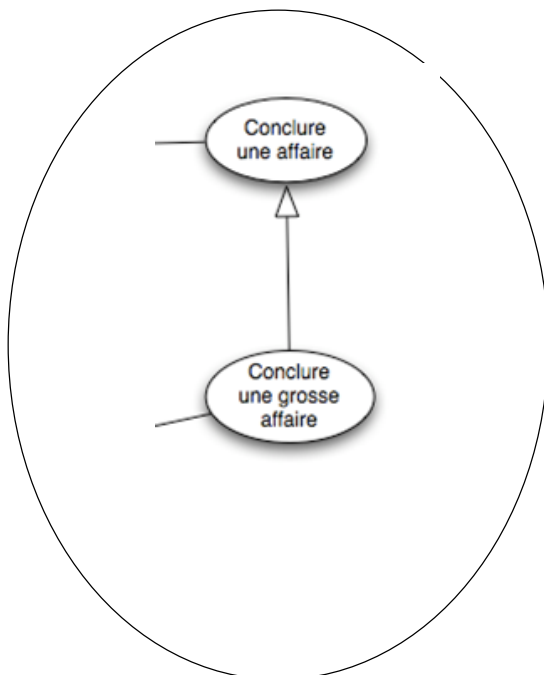


Cet exemple n'est pas correct !



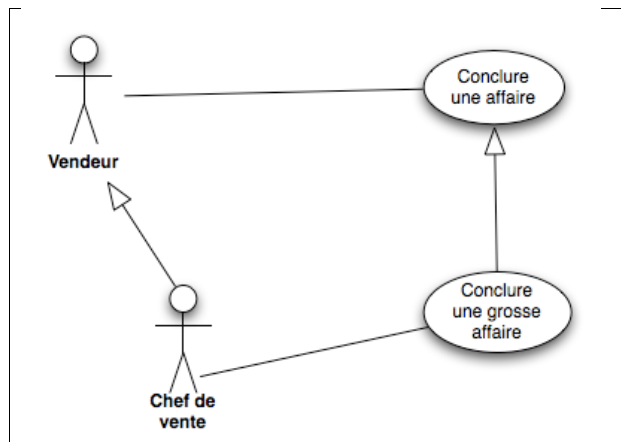


OK !!!



OK !!!





INCORRECT !!



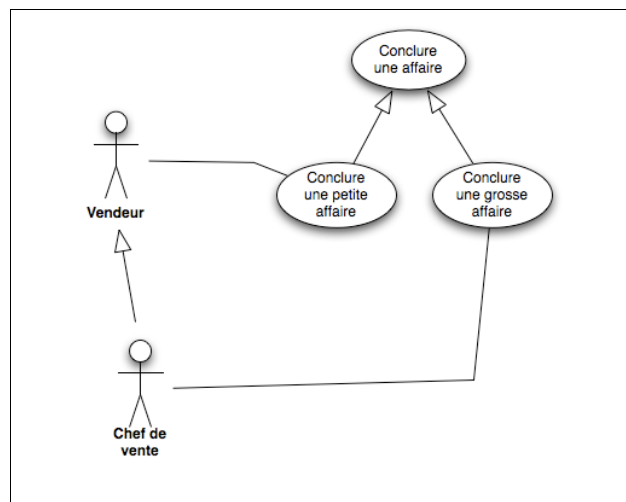
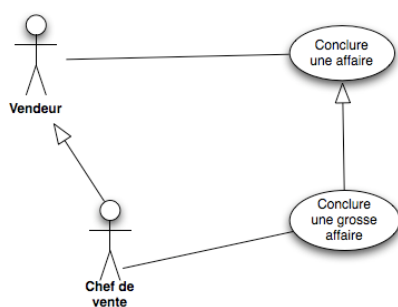
### Principe de substitution ..

Le cas **Conclure une grosse affaire** peut se substituer à **Conclure une affaire**

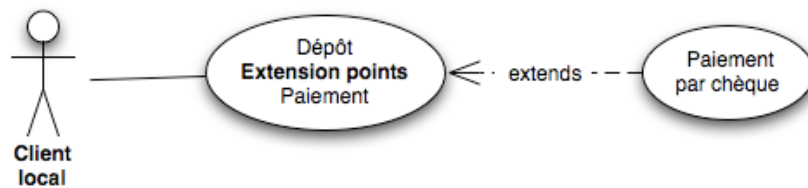
⇒ Ainsi, un simple vendeur peut conclure une grosse affaire...

.. !!

### ET COMME ÇA, ÇA MARCHE..



## RELATION EXTENDS ⇒ CAS D'UTILISATION OPTIONNELS



- **Point d'extension**

⇒ L'endroit du cas de base où aura lieu l'extension

- Condition de déclenchement du comportement facultatif

⇒ **Nom même du cas d'utilisation**

Si « **Paieement par chèque** »...

## UTILISATION DES CAS D'EXTENSION

Deux cas..

- Pour modéliser des **activités optionnelles**

⇒ Pour simplifier la rédaction du cas d'utilisation principal

- Pour modéliser des **activités asynchrones** qui peuvent provoquer des interruptions du cas de base..

.. mais qui en aucun cas ne dérangent son déroulement



## Représentation textuelle

### Dépôt

#### Scénario principal

1. Le client arrive à la caisse
2. Le client s'authentifie
3. Le client opère un dépôt pour un montant de n francs
4. ...

#### Païement par chèque

**Extension du cas:** Dépôt

**Point d'extension:** 3.(Le client opère un dépôt) ) ⇒ *Optionnel*

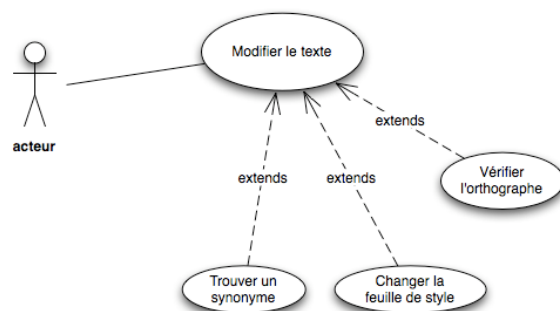
**Déclencheur:** Le dépôt est opéré par chèque plutôt qu'en liquide

#### Scénario principal :

1. Le client présente son chèque
2. La signature est authentifiée par le système
3. etc.

## Autre exemple de relation «extends»

*Un traitement de texte avec correction orthographique...*



**Cas de base:** Modifier le texte

#### Extensions:

- Trouver un synonyme *Optionnel*
- Changer la feuille de style *Optionnel*
- Vérifier l'orthographe *Asynchrone*



## EXERCICE « GESTION DE PROJET »

## COMMENT DÉFINIR LE MODÈLE DES CAS D'UTILISATION ? DÉMARCHE GÉNÉRALE

⇒ Mener une **enquête**

- ⇒ Interroger les futurs utilisateurs
- ⇒ Puis opérer une synthèse des besoins que ces derniers auront exprimés.

○ Comment opérer...



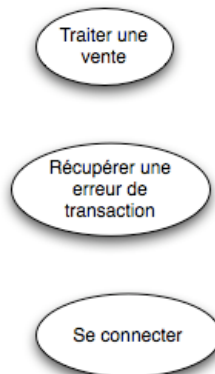
○ Quelques règles à connaître...





## EN PRELIMINAIRE : QU'EST-CE QU'UN BON CAS D'UTILISATION?

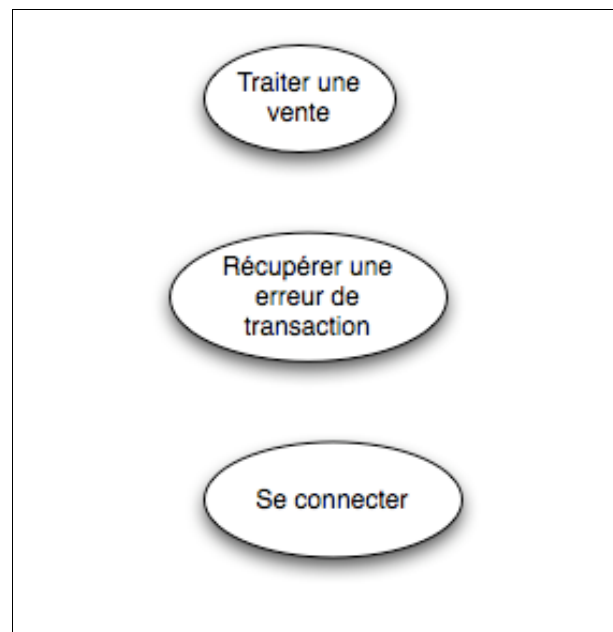
*Exemple : 3 cas d'utilisation potentiels...*



- S'agit-il de cas d'utilisation « valides »



- S'agit-il simplement d'étapes





Un « bon cas d'utilisation »..

est un « **processus métier élémentaire** »

---

**Définition : Un processus métier élémentaire**

- Est une tâche qui ajoute une valeur commerciale **observable** et **mesurable**
- Qui laisse les données du système, lorsqu'elle se termine, dans un état **stable** et **cohérent**

~~✗~~ Concrètement, une telle tâche justifie à elle seule que l'on enclenche l'ordinateur pour lancer l'application !

**Processus métier élémentaire**

- Tâche qui ajoute une valeur commerciale **observable** et **mesurable**
- Qui laisse les données du système, lorsqu'elle se termine, dans un état **stable** et **cohérent**



Traiter une  
vente



Récupérer une  
erreur de  
transaction



Se connecter

## Processus métier élémentaire

- Tâche qui ajoute une valeur commerciale **observable** et **mesurable**
- Qui laisse les données du système, lorsqu'elle se termine, dans un état **stable** et **cohérent**



Traiter une  
vente



Processus métier  
élémentaire



Récupérer une  
erreur de  
transaction



Etape (scénario  
d'erreur)



Se connecter



Etape



**R**ègle : Se conformer au principe du **processus métier élémentaire**



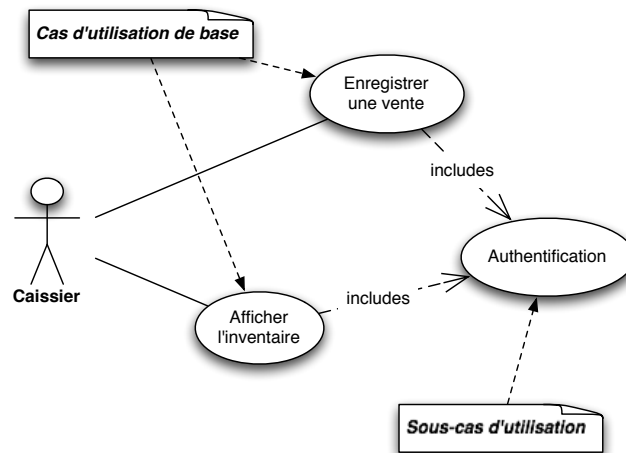
**U**ne erreur fréquente...

Considérer certaines étapes (ou actions) comme des cas d'utilisation de base.

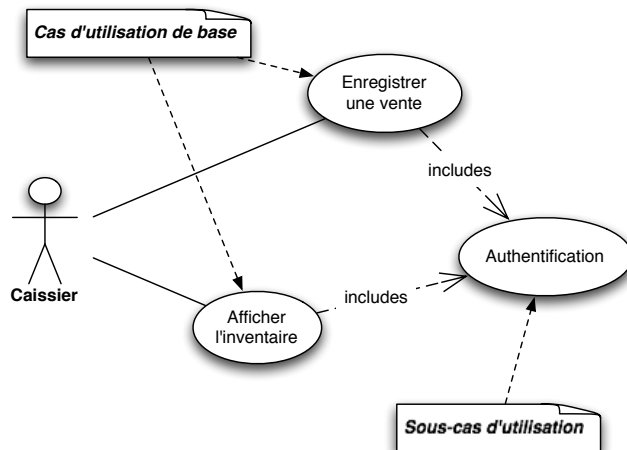


Si la même étape se retrouve dans plusieurs cas d'utilisation de base..

⇒ Extraction et mise en œuvre au moyen d'un **sous-cas d'utilisation**



ELS-Mars 2018



Un **sous-cas d'utilisation** n'est pas un processus métier élémentaire...

.. Il s'agit simplement d'une commodité !

ELS-Mars 2018

## LA DÉMARCHE DE L'ANALYSTE

*Et maintenant, la démarche...*



---

### Définition 1: Définir le modèle de cas d'utilisation

L'analyste procédera typiquement en suivant les 3 étapes suivantes:

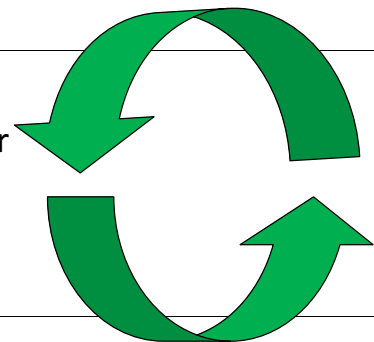
1. Délimiter les **frontières du système** à développer
  2. Identifier les **acteurs** principaux et auxiliaires
  3. Identifier les **cas d'utilisation**
- 

*Et maintenant, la démarche..*

---

### Définition 2: Définir le modèle de cas d'utilisation

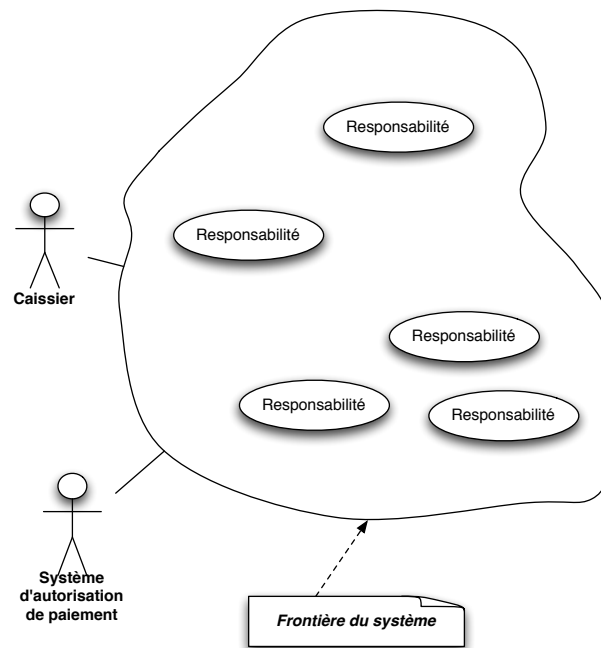
1. Délimiter les **frontières du système** à développer
  2. Identifier les **acteurs** principaux et auxiliaires
  3. Identifier les **cas d'utilisation**
- 



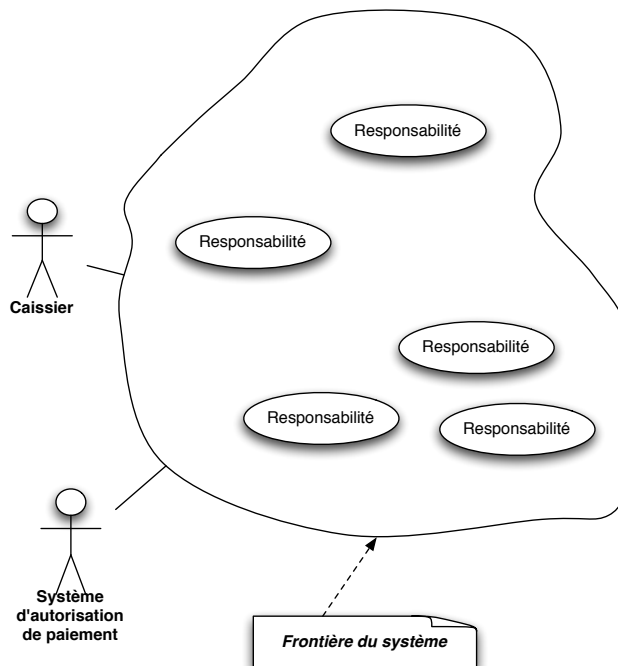
Ces trois étapes ne seront pas nécessairement séquentielles !

**Processus opéré généralement de manière itérative**

## ETAPE 1 : DÉLIMITER LES FRONTIÈRES DU SYSTÈME À DÉVELOPPER



ELS-Mars 2018



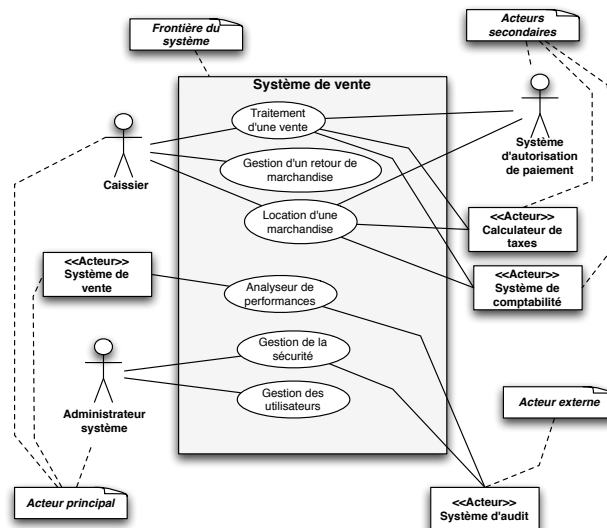
~~✗~~ Les acteurs ne font jamais partie du système à développer !

ELS-Mars 2018



## UN DIAGRAMME « DE CONTEXTE »

- ⇒ Utilisé pour illustrer la **frontière** du système
- ⇒ Diagramme de cas d'utilisation de **plus haut niveau**



ELS-Mars 2018

## ETAPE 2 : IDENTIFIER LES ACTEURS PRINCIPAUX ET LES ACTEURS AUXILIAIRES

Exemple: *Magasin de fourniture* ⇒ Identifier les **acteurs** et leurs **objectifs**

Acteur	Objectifs
Caissier	Traiter une vente Traiter une location de marchandise Traiter un retour de marchandise Encaisser
Administrateur du système	Ajouter un utilisateur Supprimer un utilisateur Modifier un utilisateur Gérer la sécurité
Directeur	Démarrer le système Arrêter le système
Système des ventes (une application distante qui interroge chaque nœud du système au sein du réseau)	Analyser les données

ELS-Mars 2018

### ETAPE 3 : IDENTIFIER LES CAS D'UTILISATION

*P* *incipe*

1. Identifier les **objectifs** des utilisateurs du système  
⇒ **Processus métier élémentaires**
2. Associer chacun à un cas d'utilisation de base  
⇒ Le nom du cas d'utilisation porte le nom de l'objectif

*T* *echnique* ?

⇒ *Par interview*

Poser deux types de questions:

- **Que faites-vous en ce moment?**  
⇒ Identification d'une procédure correspondant à une **étape**
- **Dans quel but le faites-vous?**  
⇒ Mettre en évidence les objectifs à atteindre ⇒ **Use case de base**



*Une petite interview à titre d'illustration*

**L'Analyste** : «Vous utilisez le système dans quel but?»

**Caissier** ⇒ «D'abord, je me connecte, puis j'enregistre une vente»

**Analyste** : «Pour quelle raison vous connectez-vous?»

**Caissier** ⇒ «Pour que le système m'identifie et m'autorise à enregistrer une vente ou à accomplir d'autres actions»

**Analyste** : «Pour quelle raison le système doit-il vous identifier?»

**Caissier** ⇒ «Pour éviter le vol ou encore la destruction du système»

**L'Analyste** : «Vous utilisez le système dans quel but?»

**Caissier** ⇒ «D'abord, je me connecte, puis j'enregistre une vente»

**Analyste** : «Pour quelle raison vous connectez-vous?»

**Caissier** ⇒ «Pour que le système m'identifie et m'autorise à enregistrer une vente ou à accomplir d'autres actions»

**Analyste** : «Pour quelle raison le système doit-il vous identifier?»

**Caissier** ⇒ «Pour éviter le vol ou encore la destruction du système»

### «buts» identifiés

- Eviter le vol
- Eviter la destruction du système
- Se connecter
- S'authentifier
- Enregistrer une vente

### «buts» identifiés

- Eviter le vol
- Eviter la destruction du système
- Se connecter
- S'authentifier
- Enregistrer une vente

Pas de valeur commerciale !  
⇒ Contraintes

## «buts» identifiés

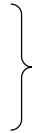
- Eviter le vol
- Eviter la destruction du système



Pas de valeur commerciale !

⇒ Contraintes

- Se connecter
- S'authentifier



Etapes !

- Enregistrer une vente

## «buts» identifiés

- Eviter le vol
- Eviter la destruction du système



Pas de valeur commerciale !

⇒ Contraintes

- Se connecter
- S'authentifier



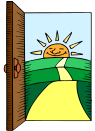
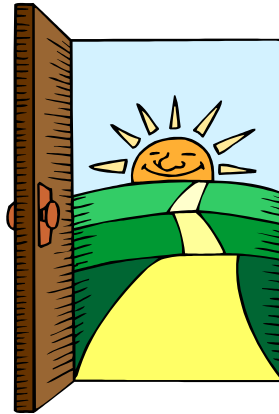
Etapes !

- Enregistrer une vente



Seul vrai cas d'utilisation

## QUELQUES RECOMMANDATIONS



## RÉDACTION DES ÉTAPES

### ✓ Simplicité d'écriture

Une étape est décrite par une locution verbale simple

*Sujet.. Verbe .. Complément*

P.e. «Le client saisit son mot de passe»



## RÉDACTION DES ÉTAPES



### ➤ Simplicité d'écriture

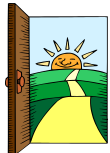
#### ✓ Le point de vue d'un oiseau

Adopter un point de vue extérieur...

Celui d'un observateur tel un oiseau perché sur sa branche qui observe l'interaction entre le ou les utilisateurs et le système.

**P.e. :** *«Le client introduit sa carte bancaire et saisit son code»*

⇒ On se contente de décrire le **QUOI**, le **COMMENT** est ignoré



## RÉDACTION DES ÉTAPES

### ➤ Simplicité d'écriture

### ➤ Le point de vue d'un oiseau

#### ✓ Montrer l'objectif de chaque étape

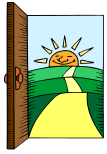
Chaque étape doit correspondre à un **mini-objectif** en soi.

A la lecture, on ne doit se poser la question « Pourquoi fait-il cela ? »

*Exemple*

~~*«Le client appuie sur la touche «RETURN»*~~

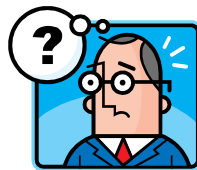
⇒ *«Le client saisit son code»*



## OUBLIER LES INTERFACES UTILISATEURS

○ Selon **Allistair Cockburn**

- ⇒ Adopter un style « essentiel »
- ⇒ Se concentrer sur l'intention, sur l'objectif : le « **QUOI** »



Et l'interface utilisateur ??



## OUBLIER LES INTERFACES UTILISATEURS

Selon **Allistair Cockburn** ⇒ Se concentrer sur: le « **QUOI** »



Et l'interface utilisateur ??

- ⇒ Dénote plutôt une réflexion s'appuyant sur le « **COMMENT** »
  - ⇒ Peut engendrer des prises de décisions trop précoces
  - ⇒ Fermant d'emblée la porte à certaines innovations

...

*Par exemple : le cas d'utilisation «Authentification»*

1. L'utilisateur saisit son nom et son mot de passe dans une boîte de dialogue
2. Le système authentifie l'utilisateur
3. Le système affiche la fenêtre principale
4. etc.




*On s'interdit par la suite d'utiliser un système innovateur du type «authentification par empreintes digitales».*

 **Mais.. mais..**

- Le futur utilisateur du système peut imposer dès le départ une interface
- La spécification d'une interface utilisateur peut déboucher sur une discussion très constructive entre les développeurs et les futurs utilisateurs du système

⇒ Ces derniers peuvent se déterminer sur du « concret »..

---

 *Dans de nombreux cas, la spécification de l'interface utilisateur sera réalisée **simultanément** à la rédaction des cas d'utilisation..*

*.. sous la forme d'un document annexé.*

---



## LE NIVEAU DE DÉTAIL

 En principe **ne pas dépasser 10 à 15 cas par page!**



Et si l'on se retrouve face à 100 voire 200 cas d'utilisation?



Et si l'on se retrouve face à 100 voire 200 cas d'utilisation?

⇒ Exploiter la notion de **paquetage**

### Paquetage ??

⇒ Regroupement de diagrammes qui coopèrent en participant à la même «fonctionnalité»

Par exemple :

- Par thème
- Par acteur