

# INGÉNIERIE DES EXIGENCES

## Analyse pilotée par l'interaction avec les utilisateurs (OOI)

IE032  
v203a

2015-03-11

Luc LAVOIE  
Département d'informatique  
Faculté des sciences



Luc.Lavoie@USherbrooke.ca  
<http://info.usherbrooke.ca/lavoie>

# TABLE DES MATIÈRES

- Aperçu
- Étapes et techniques
- Démarche
- Impact sur les procédés
- Documentation
- Atouts
- Limites
- Exemple
- Vocabulaire usuel
- Références
- À suivre



# APERÇU

- Centré sur les utilisateurs
  - plus précisément sur l'interaction entre ceux-ci et le système
- S'intéresse à la façon d'utiliser le système et non pas au problème lui-même
  - l'expertise relative au problème demeure à l'utilisateur et n'est pas encapsulée dans le système

# ÉTAPES ET TECHNIQUES

Étapes	Techniques
Recenser les cas d'utilisation (et le diagramme de structure?)	DCU (DS?)
Définir les cas d'utilisation	CU essentiels
Faire le modèle <b>objet</b> des données	UML – Diag. statiques
Faire le modèle des fonctions	UML – Diag. dynamiques
Rédiger les cas d'utilisation	CU détaillés

# DÉMARCHE

**Faire de diagramme de contexte**

**DCU**

**Recenser et définir les cas d'utilisation**

**(DS?)**

**CU essentiels**

**Faire le modèle objet des données**

**UML - Diagrammes statiques**

**Faire le modèle des fonctions**

**UML - Diagramme dynamiques**

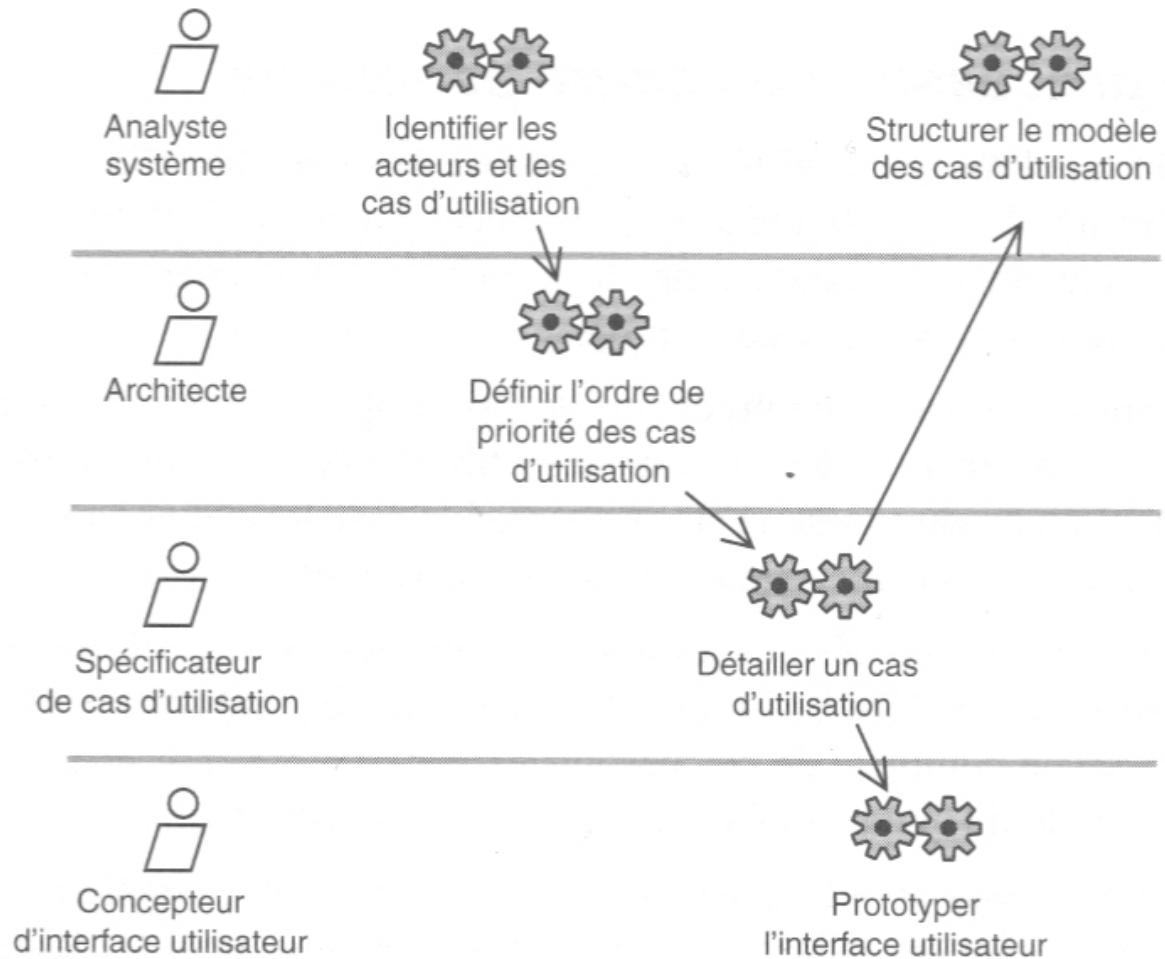
**Rédiger les cas d'utilisation détaillés**

**CU détaillés**

# IMPACT SUR LES PROCÉDÉS

**Figure 2.6**

*Enchaînement d'activités avec travailleurs et activités figurant dans des « travées ».*



tiré de Jacobson

# DOCUMENTATION

- Document d'analyse
- Document de spécification
  
- Ajout de deux sections aux documents
  - la caractérisation du problème
  - la liste des exigences de la solution

# ATOUTS

- Vision collée sur la réalité virtuelle (telle que perçue par les utilisateurs) ce qui facilite l'adhésion au système.
- Les CU et les scénarios forment une excellente base en vue de la vérification et de la validation.



# LIMITES

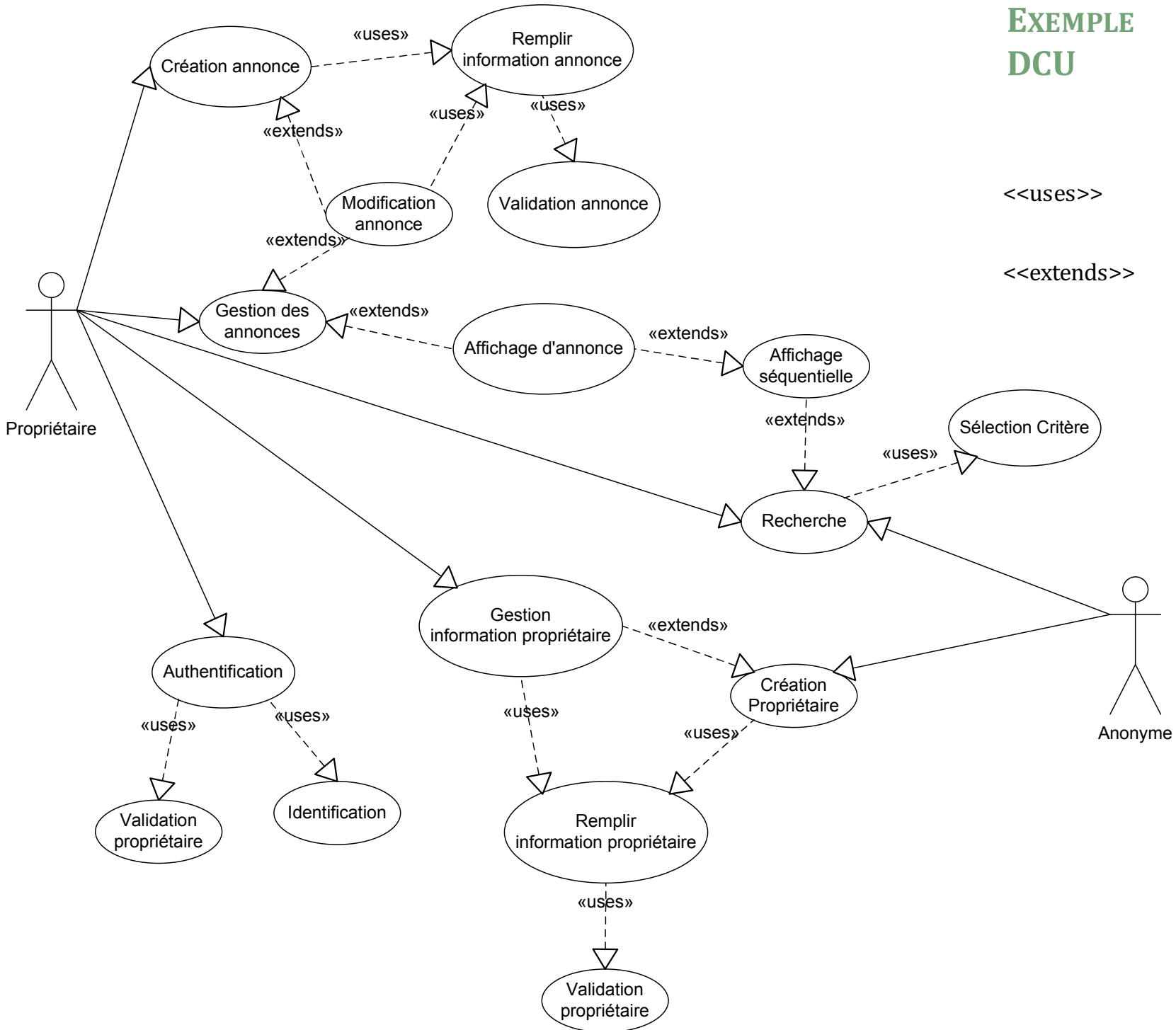
- Le domaine du problème doit être le même que celui de la solution (inconvenient sinon).
- Les CU anticipent une vision particulière du système et l'imposent par la suite à moins de procéder à des itérations « à partir de zéro »
  - ceci induit la répétition de nombreuses activités et des coûts élevés.
- Surspécification fréquente (UML).
- Plusieurs problèmes ne se définissent pas en terme d'utilisateurs, donc de CU.
- Les automatismes « échappent » à la méthode.

# SYNTHÈSE

## MÉTHODE

- Document de vision
- Diagramme de structure
- Cas d'utilisation essentiels
- Maquettes
- Modèle conceptuel de données (objet)
- Modèle conceptuel de traitement (objet)
- Cas d'utilisation détaillés

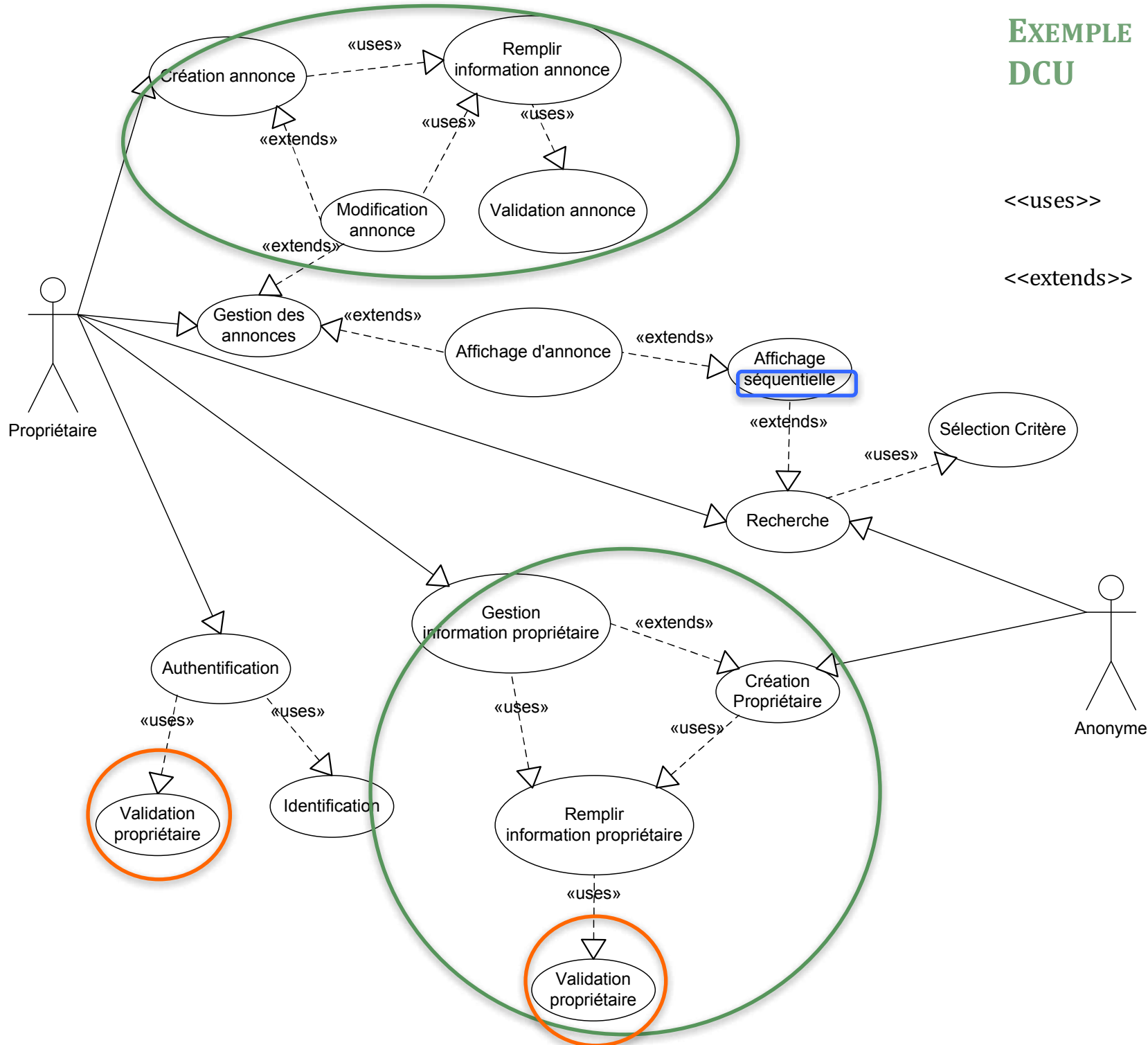
# EXEMPLE DCU



<<uses>>

<<extends>>

# EXEMPLE DCU



# CAS D'UTILISATION (1/2)

<b>CAS #</b>	<identificateur unique (phrase courte avec ver d'action)>	
<b>Objectif</b>	<une description de l'objectif en relation avec le contexte>	
<b>Portée, niveau ou type</b>	<Système, sous-système, application> <Service, Fonction, Sous-fonction> <Essentiel, Détaillé>	
<b>Préalable</b>	<état préalable requis (environnement)>	
<b>Conséquent (S)</b>	<état subséquent garanti (succès)>	
<b>Conséquent (E)</b>	<état subséquent garanti (échec)>	
<b>Acteurs</b>	<acteurs susceptibles d'utiliser ce cas>	
<b>Dépendances</b>	<autres éléments requis par celui-ci pour s'accomplir>	
<b>Déclancheur</b>	<évènement provoquant le démarrage du cas>	
<b>DESCRIPTION</b>	<b>Étape</b>	<b>Action</b>
	1	<début>
	2	<...>
	...	<...>
	n	<fin>
<b>EXCEPTIONS</b>	<b>Étapes</b>	<b>Garde Action</b>
	1a	<condition d'exception> : <traitement de l'exception>
	1b	<condition d'exception> : <traitement de l'exception>
<b>VARIANTES</b>		<b>Garde Action</b>
	1	<condition d'existence de la variante> : <traitement de la variante>

# CAS D'UTILISATION (2/2)

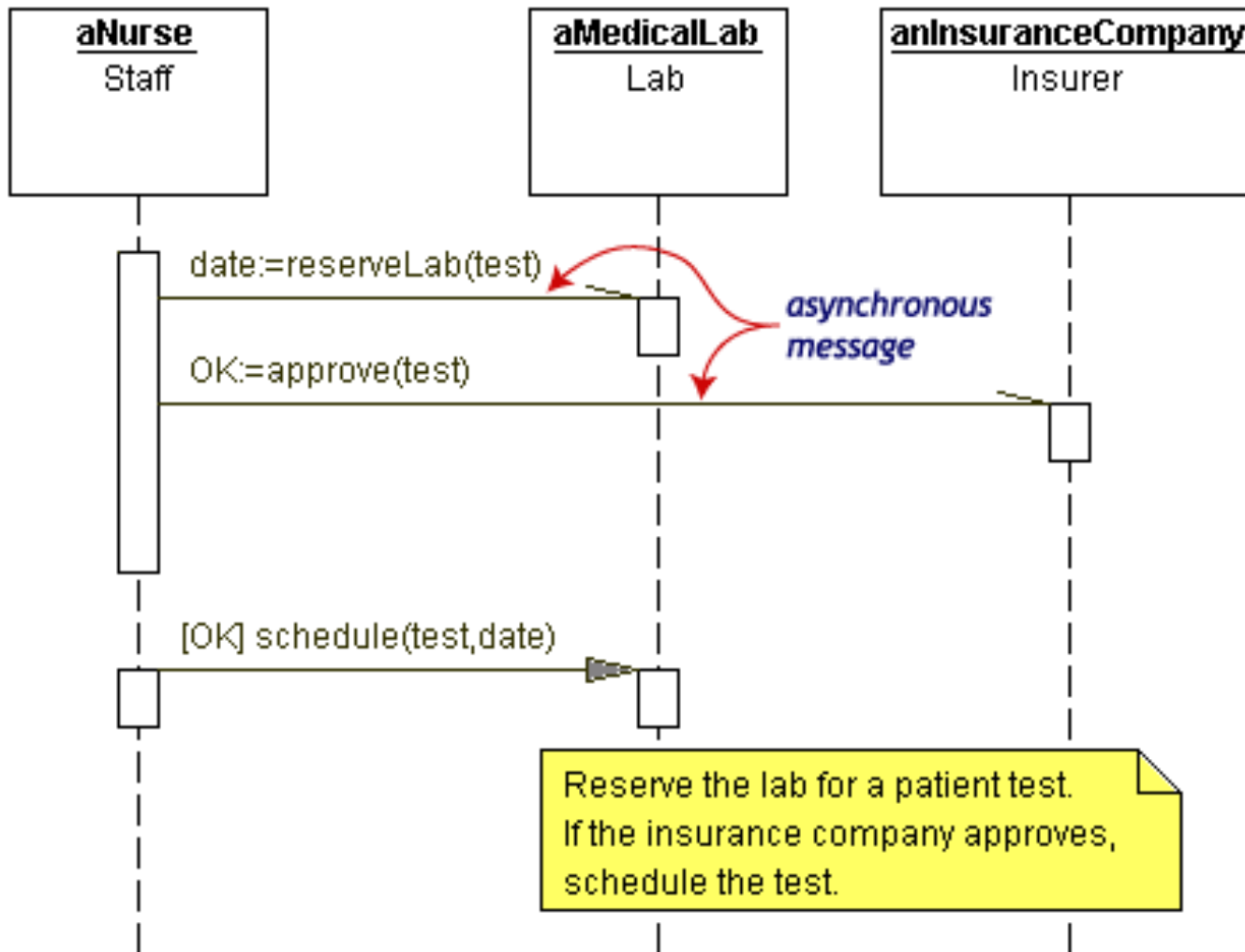
<b>COMPLÉMENT</b>	<identificateur du cas>
<b>Priorité</b>	<voir IEEE 1233>
<b>Criticité</b>	<voir IEEE 1233>
<b>Performance</b>	<voir IEEE 1233>
<b>Frequence</b>	<fréquence d'utilisation>
<b>Médiateurs</b>	<entre l'acteur et le cas: équipement, fichier, etc.>
<b>Problèmes ouverts</b>	<liste des problèmes ouverts en attente de solution>
<b>Échéance</b>	<date de mise à disposition>
<b>Cas utilisateurs</b>	<...>
<b>Cas utilisés</b>	<...>
<b>...autre attribut...</b>	<...au besoin...>

Attention, il y a beaucoup de façons différentes de présenter les CU, de les catégoriser et même de les définir, voir par exemple Larman, Jacobson, Sommerville, etc.

# AFFINEMENT DES CU

- Définition des classes
  - Diagrammes de classes (UML-C)
- Définition des états des objets ou des classes
  - Diagramme d'états (UML-E)
- Définition de leurs fonctions
  - Diagramme de séquence (UML-S)
  - Diagramme d'activité (UML-A)

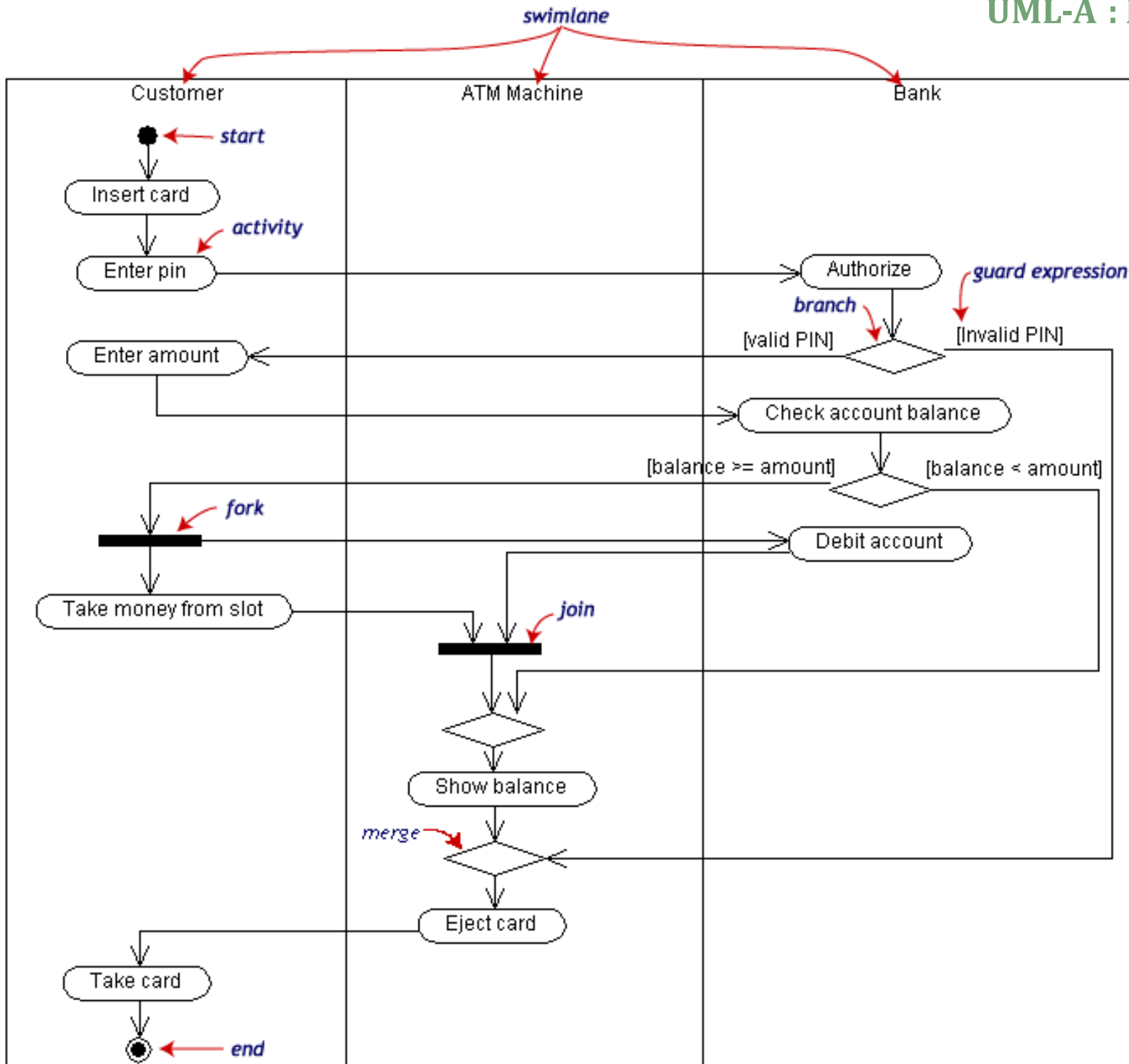
# UML-S : EXEMPLE 1

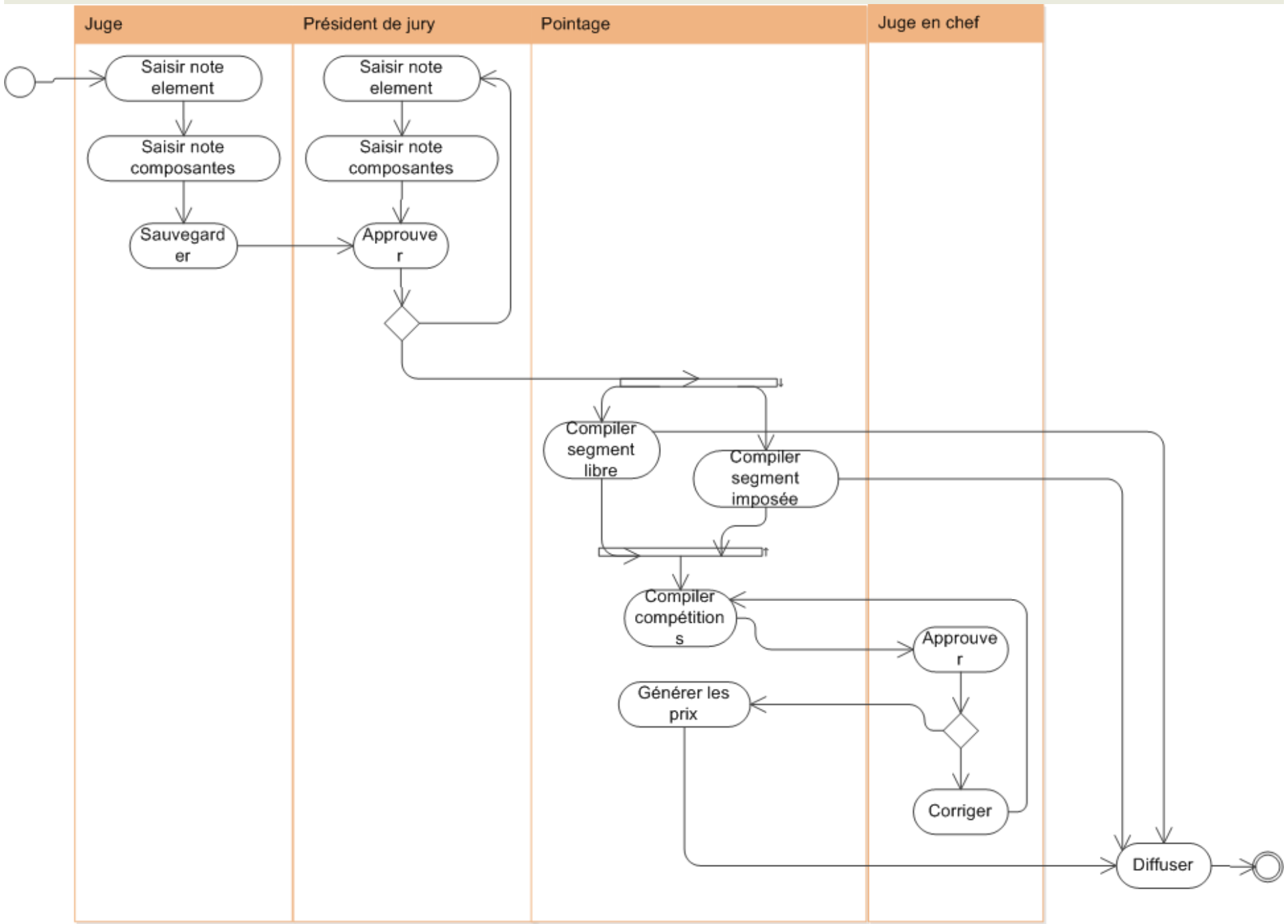


- simple (a/s)
- ←--- simple (a/s) – facultatif
- ▶ synchrone
- ▶ asynchrone
- or →▶
- or →▶



# UML-A : EXEMPLE 1





# UML...

- Pour tout ce qui a trait à la (très étendue et très complexe) notation UML, vous pouvez consulter, entre autres ouvrages, ceux de Laurent Audibert – voir le site
  - <http://laurent-audibert.developpez.com/Cours-UML/>
- Se rappeler que cette notation est d'abord dédiée à la conception et non à l'analyse.