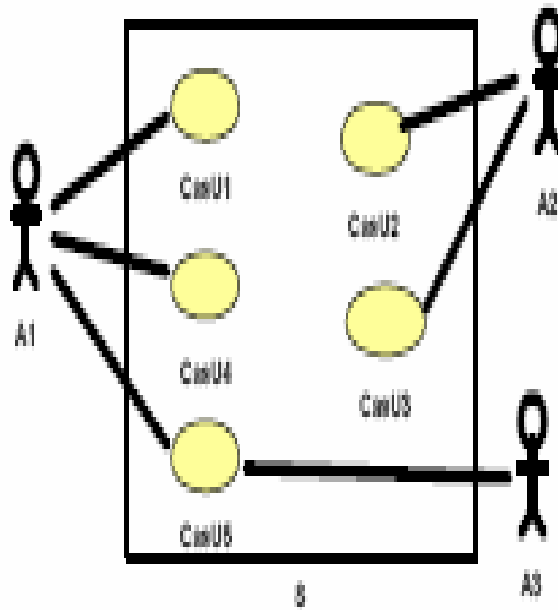


UML: Diagramme de cas d'utilisation

Unité:



Formateur: Y. Naji
Analyse et conception orientée objet (UML)

Plan du cours

Unité:

- ❑ Objectifs du diagramme use case
- ❑ Éléments du diagramme
 - ❖ Acteurs
 - ❖ Généralisation entre Acteurs
 - ❖ Cas d'utilisation
 - ❖ Relation entre acteur et cas d'utilisation
- ❑ Associations et cas d'utilisation
- ❑ Exemple: Gestion d'une bibliothèque
- ❑ Description textuelle des cas d'utilisation
- ❑ Exercice

Formateur: Y.Naji

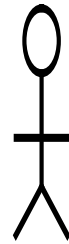
Analyse et conception orientée objet (UML)

Objectifs

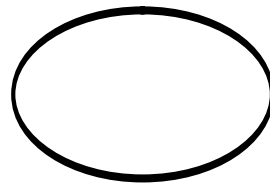
- ❑ Il capture le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit
- ❑ Il permet de recueillir, d'analyser et d'organiser les besoins et de recenser les grandes fonctionnalités d'un système
- ❑ Il scinde la fonctionnalité du système 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
- ❑ Pour élaborer les cas d'utilisation, il faut se fonder sur des entretiens avec les utilisateurs.

Éléments du diagramme

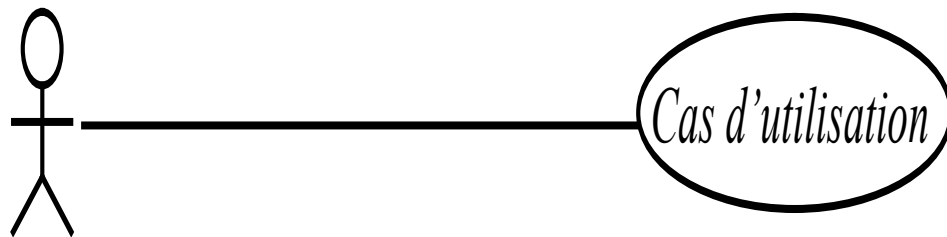
Acteur



Cas d'utilisation

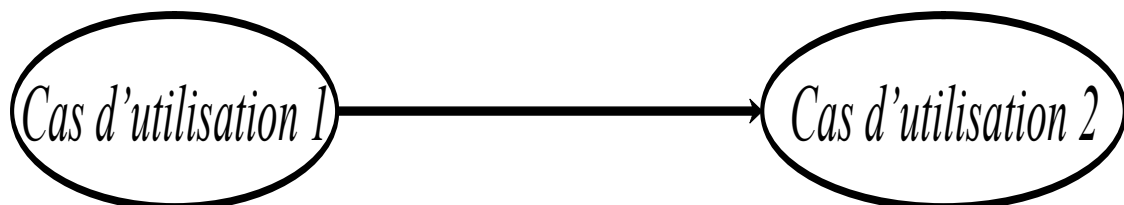


Relation entre acteur et cas d'utilisation



Unité:

Relation entre cas d'utilisations

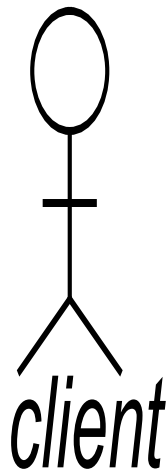


Formateur: Y. Naji

Analyse et conception orientée objet (UML)

Acteurs

- ❑ un acteur est la description d'un ensemble cohérent de rôle qu'un utilisateur joue lorsqu'il interagit avec le système
- ❑ les acteurs sont à l'extérieur du système et dialoguent avec lui
- ❑ Il se représente par un petit bonhomme avec son nom (*i.e.* son rôle) inscrit dessous.



`<<acteur>>`
formateur

Acteurs

On distingue plusieurs types d'acteurs:

- ❑ Acteurs principaux

- ❖ personnes qui utilisent les fonctions principales du système

- ❑ Acteurs secondaires

- ❖ personnes qui effectuent des tâches administratives ou de maintenance

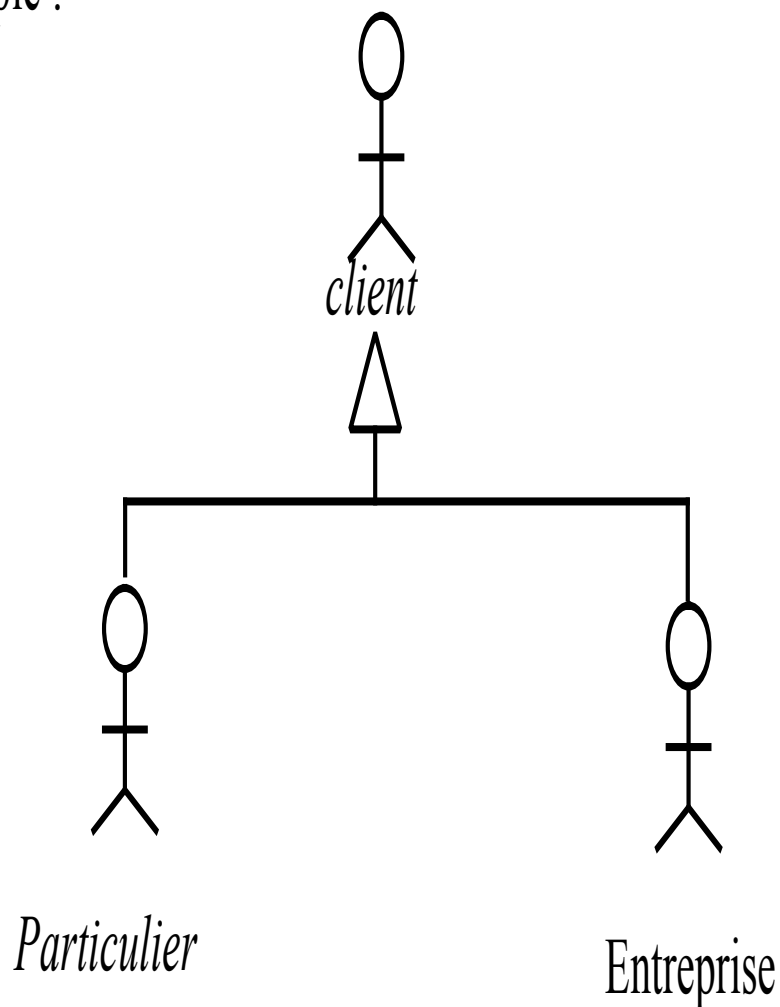
- ❑ Matériel externe

- ❖ dispositifs matériels faisant partie du domaine de l'application

- ❑ Autres systèmes

Généralisation entre Acteurs

- ❑ Les acteurs peuvent avoir des associations de généralisation
- ❑ Exemple :



Unité:

Formateur: Y.Naji

Analyse et conception orientée objet (UML)

Cas d'utilisation

Unité:

- ❑ Un cas d'utilisation est une unité cohérente représentant une fonctionnalité visible de l'extérieur
- ❑ Un cas d'utilisation est décrit par un verbe à l'infinitif et un complément
- ❑ C'est une description d'une fonctionnalité et d'une interaction entre un acteur et un système.

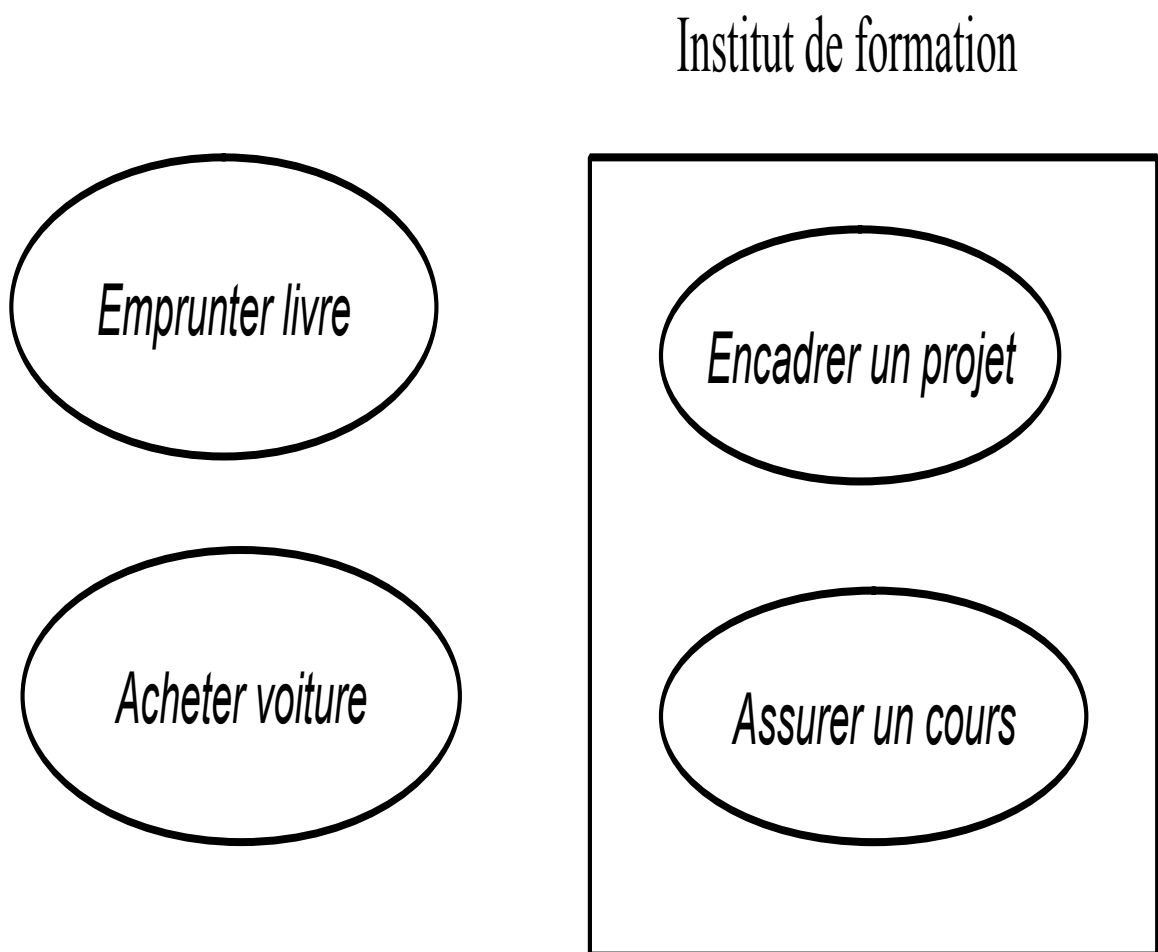
Formateur: Y. Naji

Analyse et conception orientée objet (UML)

Cas d'utilisation

- ❑ Un cas est représenté par une ellipse
- ❑ un ensemble de cas peut être placé dans un rectangle qui symbolise le système

Unité:

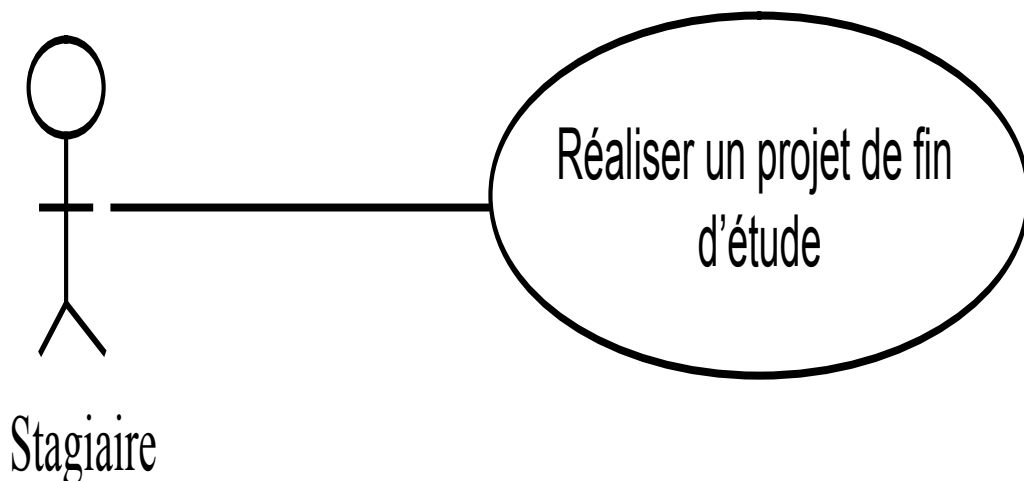


Formateur: Y. Naji

Analyse et conception orientée objet (UML)

Relation entre acteur et cas d'utilisation <<communiqué>>

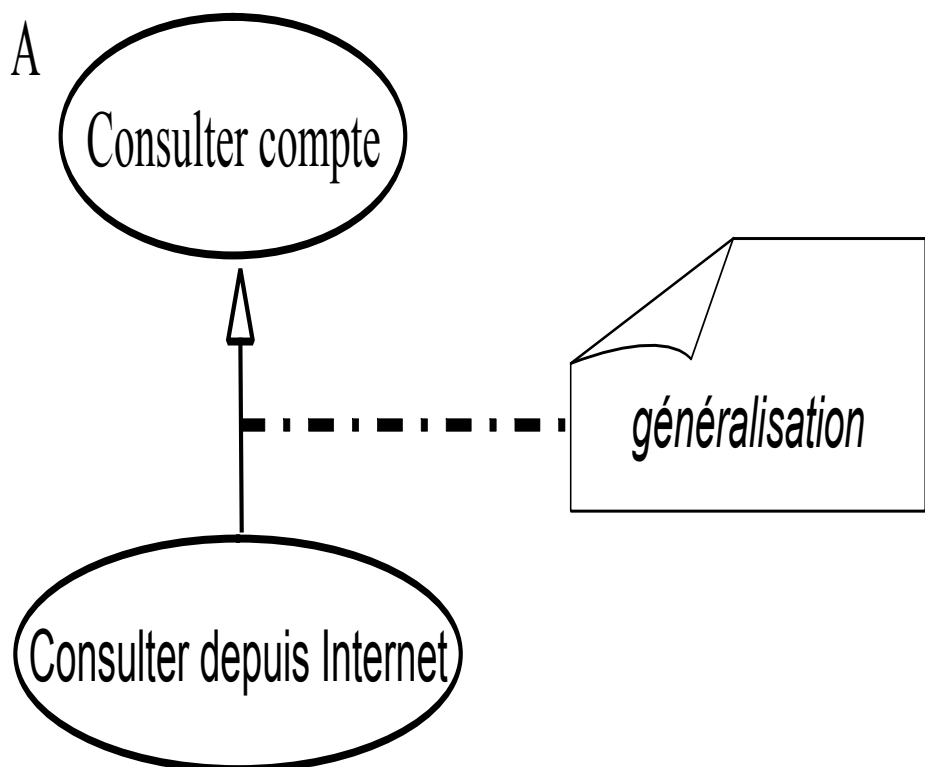
- ❑ La participation d'un acteur est représentée par une ligne solide entre l'acteur et le cas d'utilisation.
- ❑ Le sens de la flèche indique l'initiateur de l'interaction.
- ❑ C'est la seule relation possible entre un acteur et les cas d'utilisation.



Associations et cas d'utilisation

la généralisation

- ❑ L'association de généralisation entre cas d'utilisation a la même sémantique que pour les classes
- ❑ Un cas A est une généralisation d'un cas B si B est un cas particulier de A

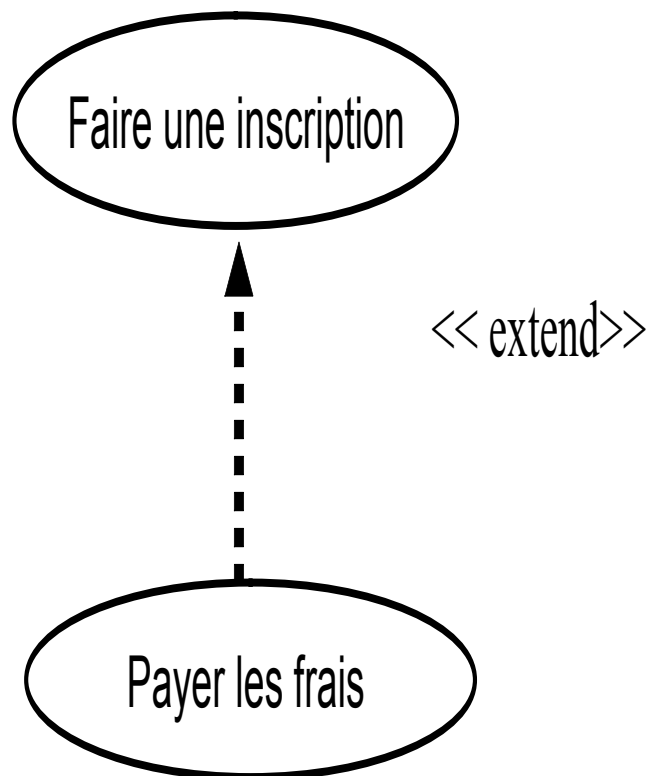


Associations et cas d'utilisation

<<extend>>

- ❑ 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.

Unité:



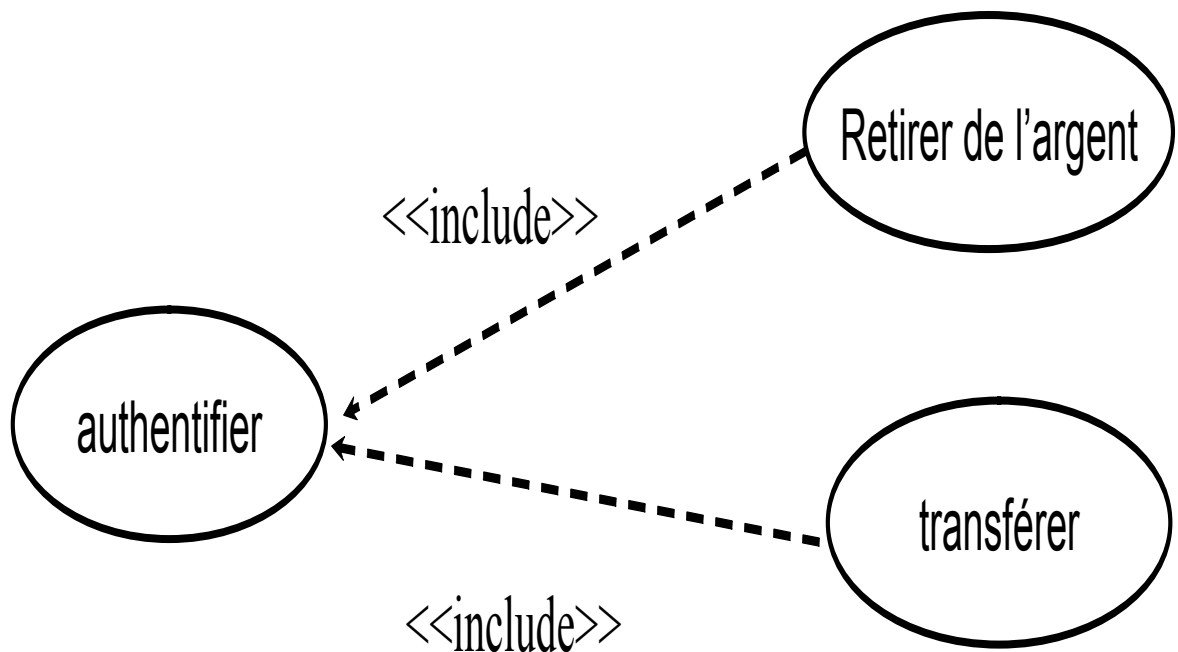
Formateur: Y. Naji

Analyse et conception orientée objet (UML)

Associations et cas d'utilisation

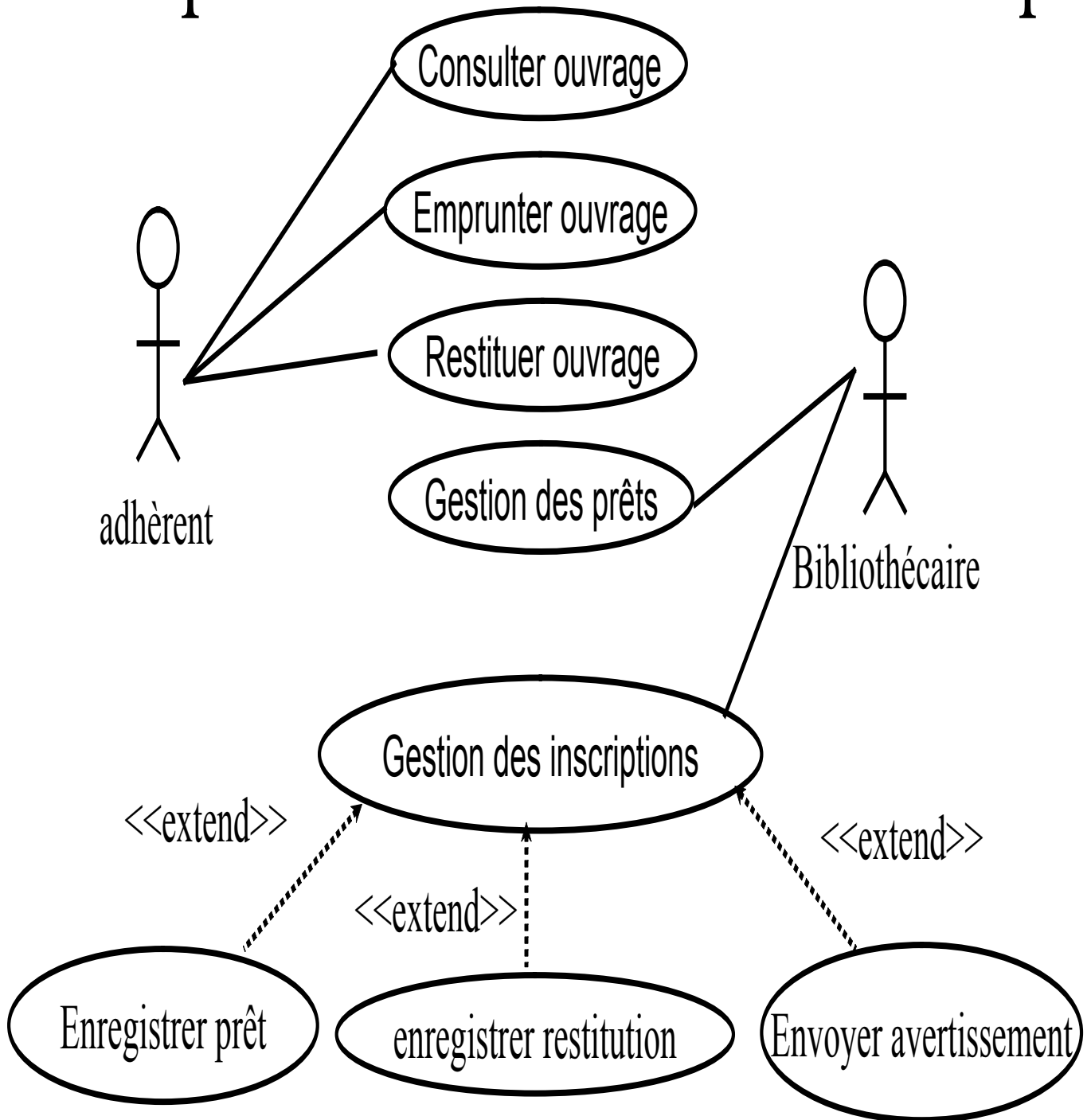
<<include>>

- ❑ indique que le cas d'utilisation source contient aussi le comportement décrit dans le cas d'utilisation destination.
- ❑ On utilise ce stéréotype lorsque que l'on souhaite factoriser un cas d'utilisation partagé par plusieurs autres cas d'utilisation



Exemple: Gestion d'une bibliothèque

Unité:



Formateur: Y. Naji

Analyse et conception orientée objet (UML)

Description textuelle des cas d'utilisation

- ❑ Permet de fournir un format de présentation textuelle à la fois souple et riche
- ❑ La description textuelle d'un cas d'utilisation n'est pas normalisée, il n'y a aucune proposition de l'OMG sur ce point concernant les standards UML.
- ❑ On peut représenter un cas d'utilisation selon 3 types de scénario
 - ❖ **Scénario nominal** : C'est le scénario « idéal » (tout se passe bien) pour le cas d'utilisation. On décrit un enchaînement d'actions (Acteur-Système) qui conduisent au bon déroulement du cas d'utilisation. On parle également de scénario de base, scénario normal.
 - ❖ **Scénario alternatif** : Partant du scénario nominal, on étudie chaque point de l'enchaînement et s'il existe une variante, on propose « une nouvelle façon de dérouler le cas d'utilisation ».
 - ❖ **Scénario d'exception** : Même description que pour le scénario alternatif sauf que les post_conditions du cas d'utilisation ne sont pas remplies.

Description textuelle des cas d'utilisation

- le scénario peut être représenté par:
- ❖ **Cas d'utilisation** :Nom du cas d'utilisation
- ❖ **Acteurs** :Liste des acteurs qui interviennent pour le cas d'utilisation
- ❖ **But** :On indique en une phrase quel est l'objectif du cas d'utilisation (ce qu'il offre aux acteurs)
- ❖ **Résumé Métier** :On propose une brève description du cas d'utilisation
- ❖ **Pré conditions** :Les conditions nécessaires pour déclencher le cas d'utilisation
- ❖ **Post conditions** :Les conditions de sortie du cas d'utilisation (état du système après réalisation du cas d'utilisation)
- ❖ **Commentaires** :On peut indiquer ici les règles de gestion liées au cas d'utilisation ou tout autre commentaire pertinent à la compréhension du cas d'utilisation (par exemple les cas d'utilisation en relation avec celui décrit).

exemple

Cas d'utilisation : *gestion de prêt d'un livre*

☐ **Acteurs** : Le bibliothécaire

☐ **But** : Permettre au bibliothécaire de gérer les emprunts de livre de sa bibliothèque.

☐ **Résumé Métier**:

❖ L'adhérent présente au bibliothécaire le livre qu'il désire emprunter.

❖ Le bibliothécaire identifie l'adhérent et enregistre la référence du livre que l'adhérent souhaite emprunter.

☐ **Pré conditions**:

❖ Le client est un adhérent de la bibliothèque (il a une référence adhérent).

❖ Les livres disponibles sont référencés.

☐ **Post conditions** : L'emprunt est enregistré.

☐ **Commentaires** :

❖ Un adhérent ne peut pas emprunter plus de cinq livres.

❖ Chaque emprunt ne peut excéder deux semaines.

Unité:

Formateur: Y.Naji

Analyse et conception orientée objet (UML)

exemple

Scénario Nominal

- 1) Le bibliothécaire identifie l'adhérent.
- 2) Le système vérifie l'existence de l'adhérent dans la liste des clients de la bibliothèque.
- 3) Le système vérifie que l'adhérent est autorisé à emprunter un livre.
- 4) Le bibliothécaire enregistre la référence du livre que l'adhérent a choisi.
- 5) Le système calcule la date de retour de prêt.

Unité:

Formateur: Y.Naji

Analyse et conception orientée objet (UML)

exemple

 Scénario(s) Alternatif(s)1. Point 2 : 2.1 La référence client est inconnu

2.1.1 Le système indique au bibliothécaire que la référence adhérent est inconnu.

2.1.2 Le système propose au bibliothécaire de renseigner une nouvelle fois la référence.

2.1.3 Reprise de l'enchaînement du scénario nominal au point 1.

2. Point 5 : 5.1 La référence du livre est inconnue

5.1.1 Le système indique au bibliothécaire que la référence du livre est inconnu.

5.1.2 Le système propose au bibliothécaire de renseigner une nouvelle fois la référence.

5.1.3 Reprise de l'enchaînement du scénario nominal au point 4.

 Exception(s)3. Point 3 : 3.1 L'adhérent a déjà emprunté cinq livres

3.1.1 Le système indique au bibliothécaire que l'adhérent ne peut plus emprunter de livre.

Exercice d'application

Question:

gestions des cours dans une
université

Solution

Unité:

Formateur: Y. Naji

Analyse et conception orientée objet (UML)