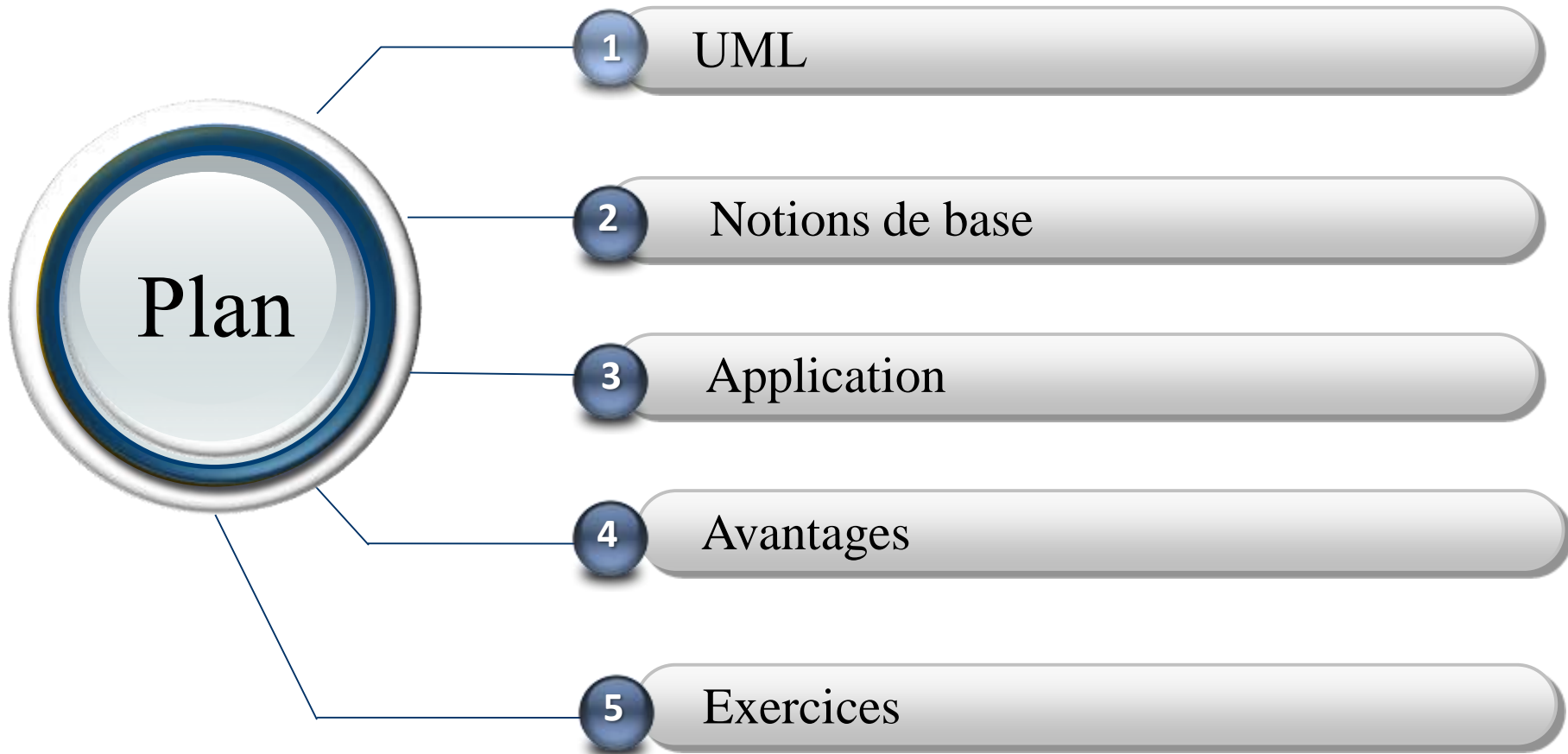




# Informatique 3: UML

**Dr. Ing. NOUBISSI Justin-Hervé**



# Uniform Modeling Language (UML)

- »»»» Langage de modélisation et de spécification orienté objet
- »»»» Notations sous forme graphique
- »»»» 14 diagrammes identifiés (UML 2.3)

## Les sous-ensembles

- Les **vues** : organisationnel, dynamique, temporel, architectural, etc.
- Les **diagrammes** : description graphique du contenu des vues (notions abstraites).
- Les **modèles d'élément** : briques de base d'UML utilisées dans plusieurs types de diagramme.

# Uniform Modeling Language (UML)

## 9 diagrammes principaux

- »»»» Le **Diagramme de classe** : les classes et leurs relations.
- »»»» Le **Diagramme d'états** : comportement d'une classe en terme d'états.
- »»»» Le **Diagramme d'activité** : comportement d'une opération en terme d'actions.
- »»»» Le **Diagramme de séquence** : représentation temporelle des objets et de leurs interactions.
- »»»» Le **Diagramme des cas d'utilisation** : fonctions du système du point de vue de l'utilisateur.

# Uniform Modeling Language (UML)

## 9 diagrammes principaux (suite)

»»»» Le **Diagramme objet**: les objets et leurs relations.

»»»» Le **Diagramme de composant**: composant physique de l'application.

»»»» Le **Diagramme de déploiement**: déploiement des composants sur des dispositifs matériels.

»»»» Le **Diagramme de collaboration**: représentation spatiale des objets, des liens et des interactions.

# Notions de Base

## 1. Diagramme de classe



# Notions de base

## Relations



**Dépendance** : La classe A utilise brièvement la classe B



**Association** : La classe A est en relation avec la classe B pour un certain temps



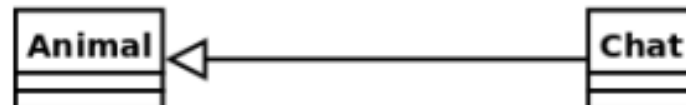
**Agrégation** : La classe A appartient à la classe B



**Composition** : La classe A fait partie de la classe B



**Généralisation (ou Héritage)** : La classe A est une classe B



## 2. Diagramme Objet

- ✓ Permet d'obtenir une structure statique du système
- ✓ Utile pour préciser la structure des classes

nom de l'objet

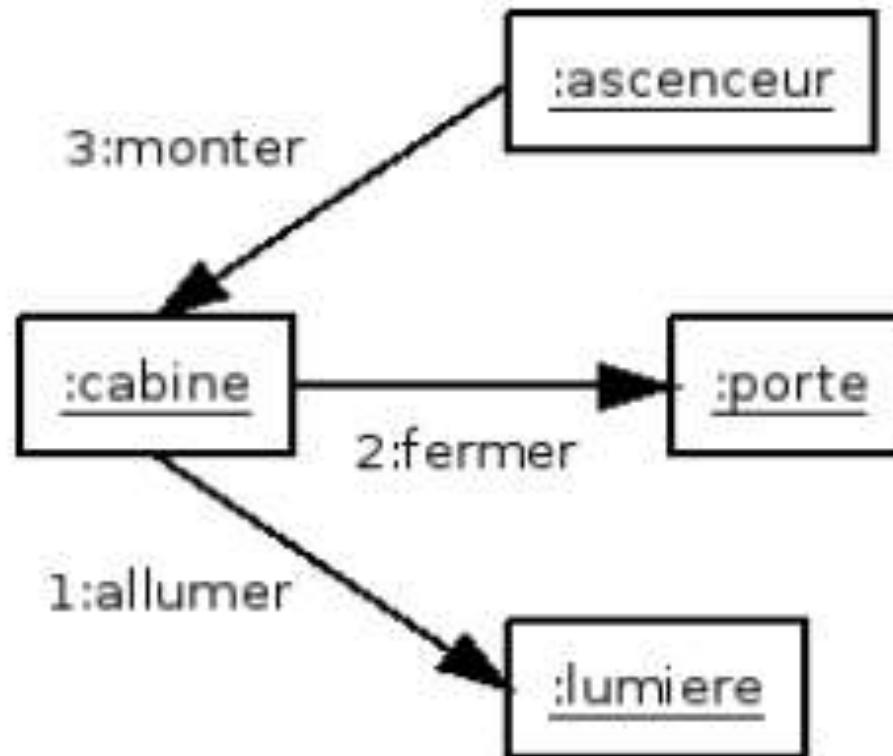
nom de l'objet : classe



# Notions de base

## 3. Diagramme de collaboration

- ✓ Décrit les interactions entre objets
- ✓ Extension du diagramme d'objets en insistant sur leurs interactions et en explicitant l'ordre d'envoi des messages



## 4. Diagramme de de cas d'utilisation

### Un cas d'utilisation



Représente un élément fonctionnel identifié dans un système

Exemple: Visualisation d'un rapport de travail

### Un acteur



« déclenche » un cas d'utilisation

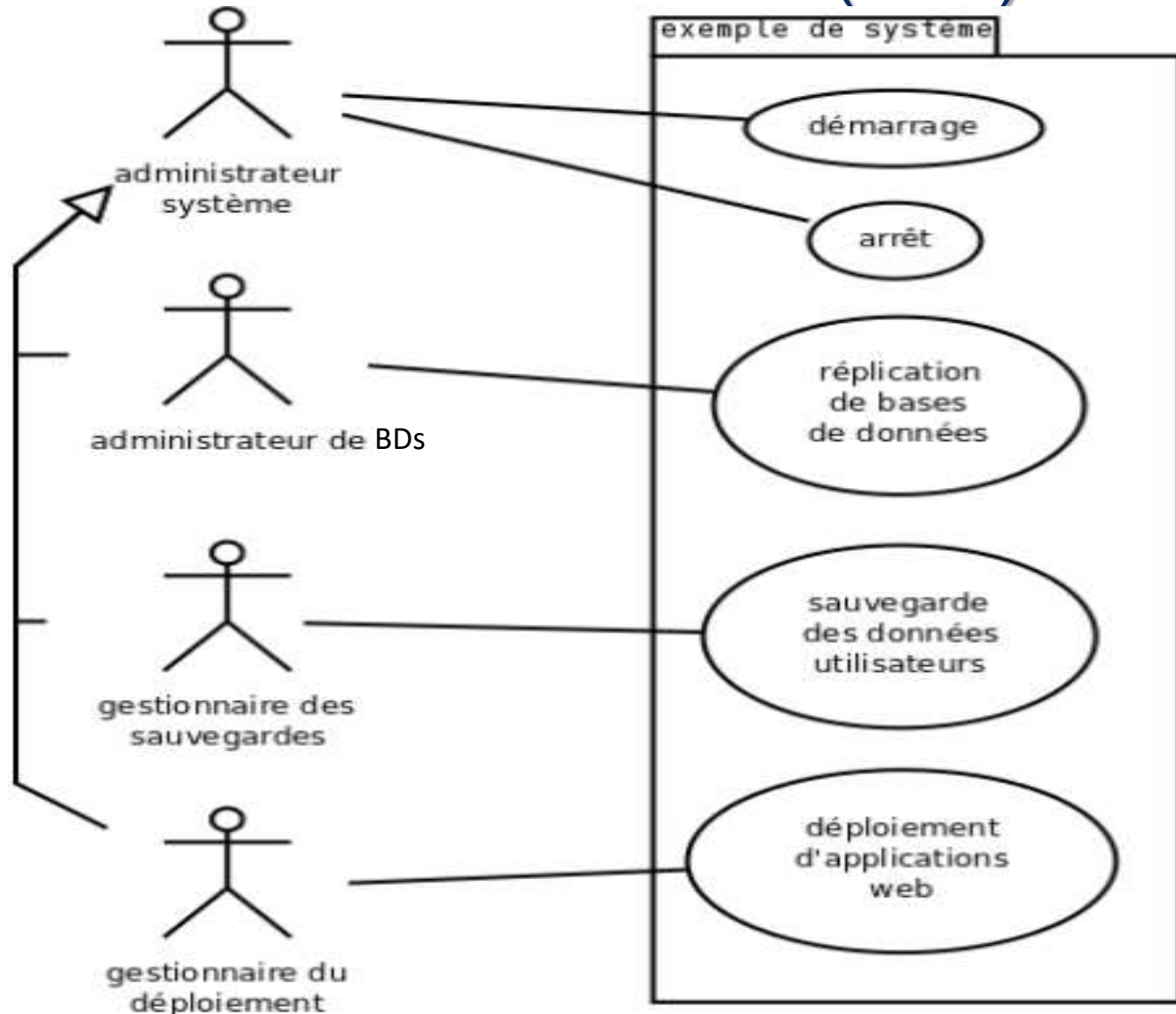


Peut être un humain ou un système externe au cas d'utilisation

Exemple: une horloge système qui déclenche un cas d'utilisation à un instant donné

# Notions de base

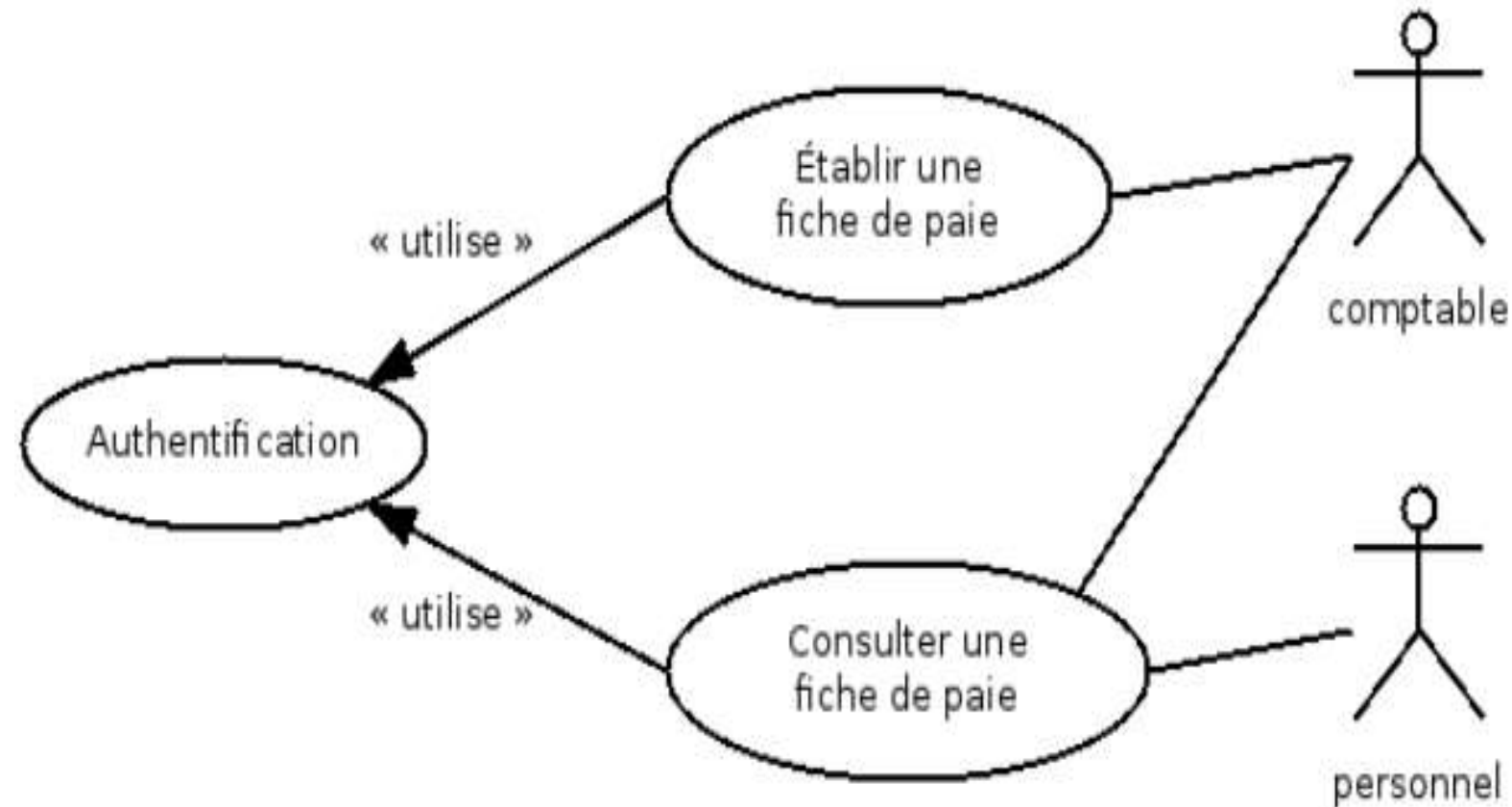
## 4. Diagramme de de cas d'utilisation (suite)



# Notions de base

## 4. Diagramme de de cas d'utilisation (suite)

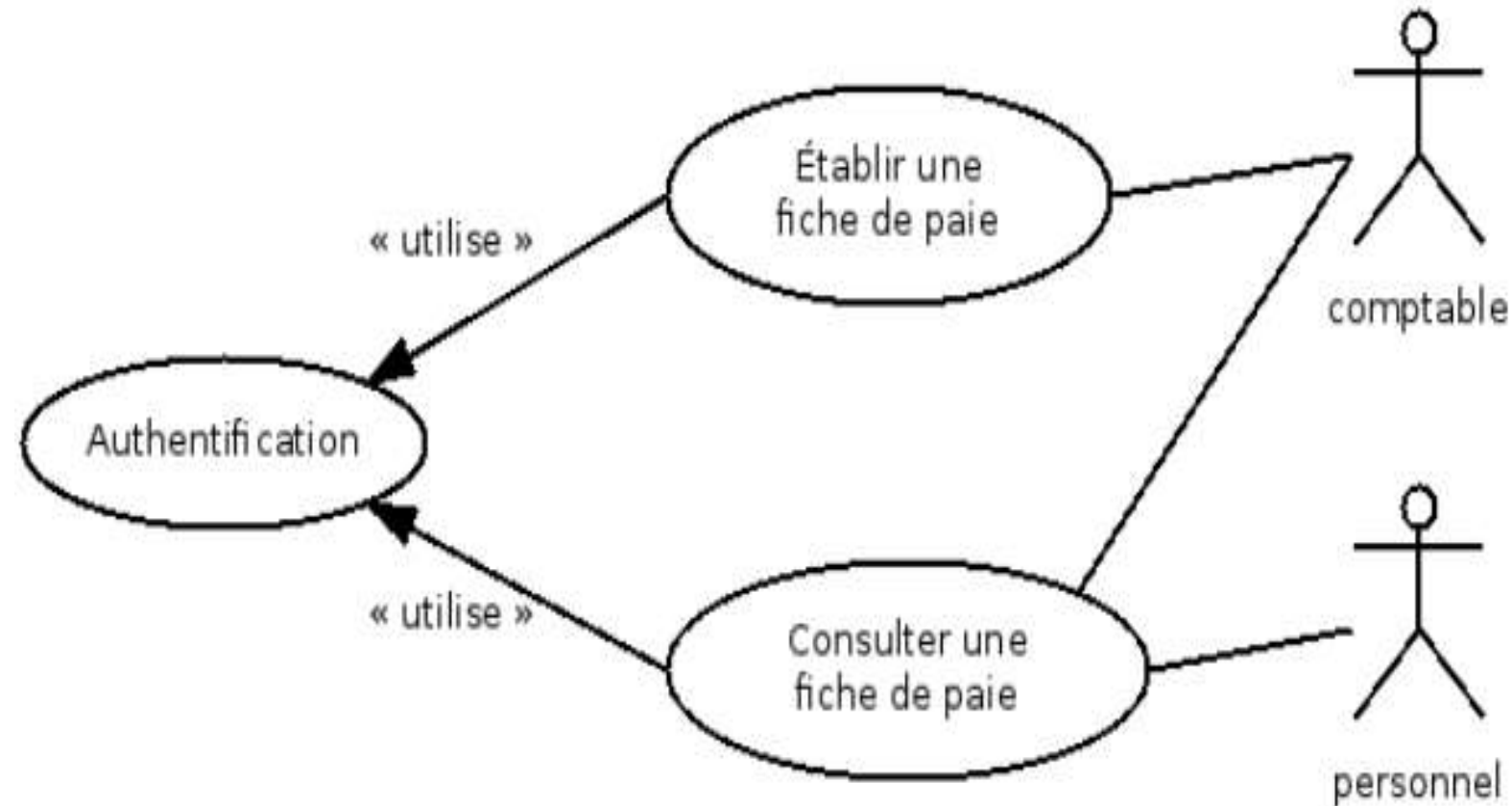
### Liens entre cas d'utilisation



# Notions de base

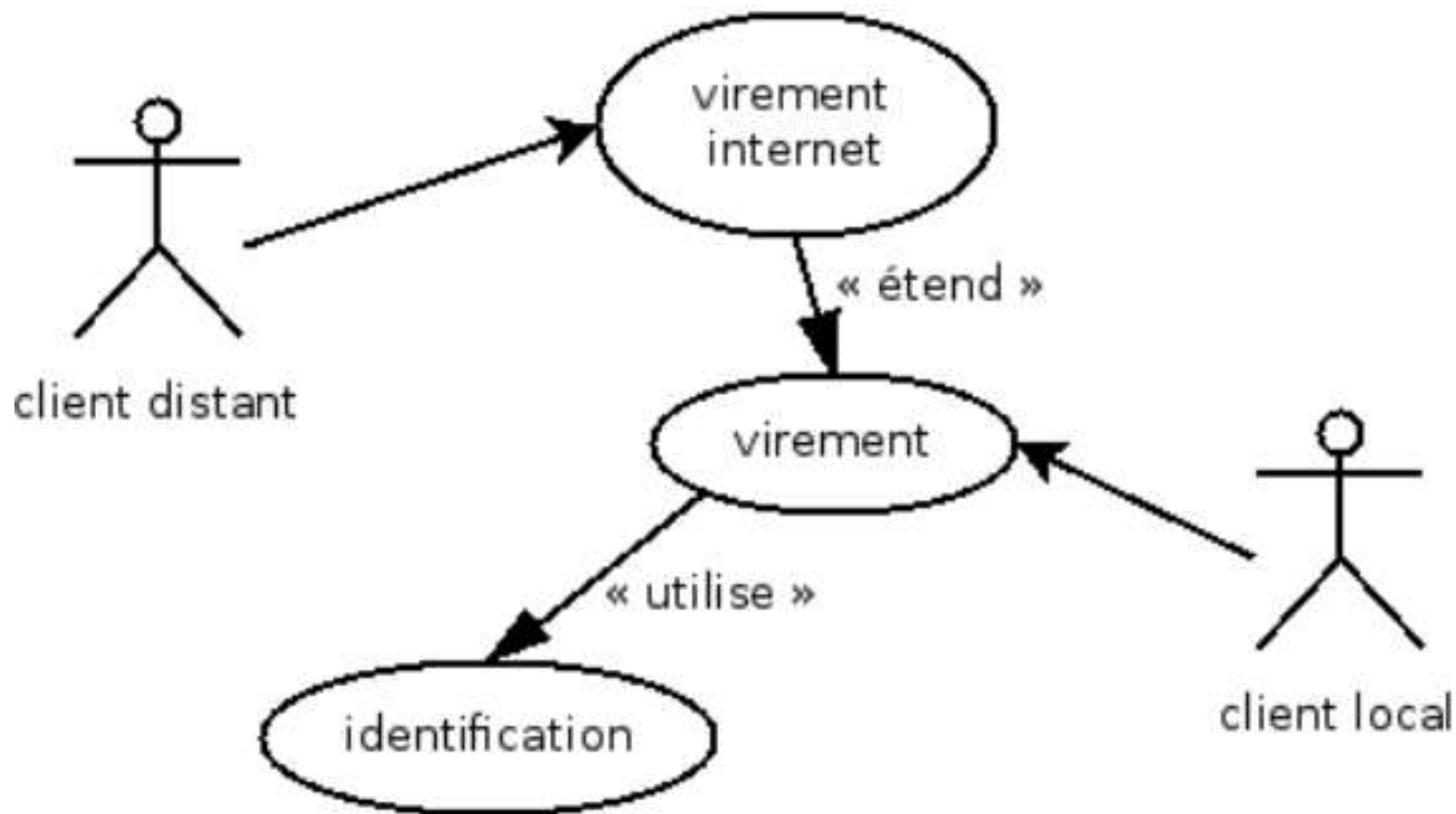
## 4. Diagramme de de cas d'utilisation (suite)

### Liens entre cas d'utilisation



## 4. Diagramme de de cas d'utilisation (suite)

### Liens entre cas d'utilisation

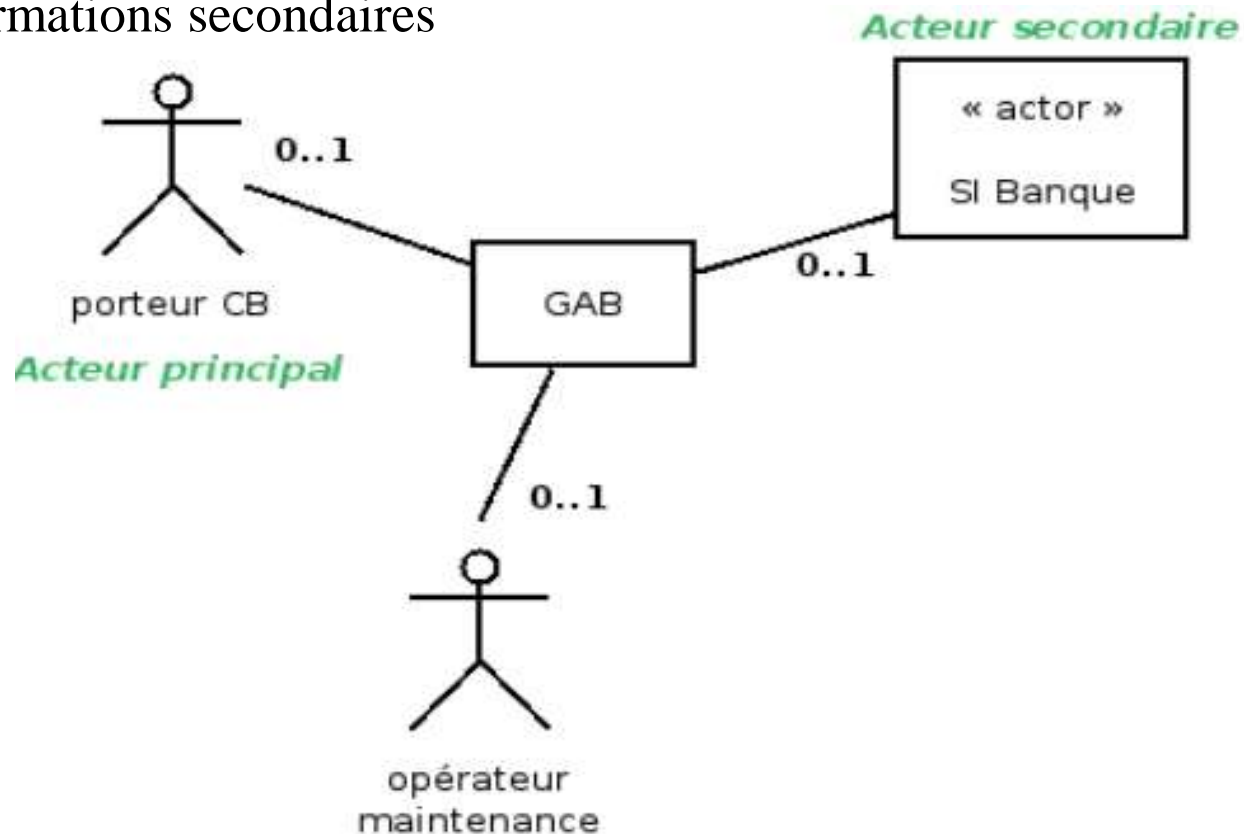


# Notions de base

## 4. Diagramme de de cas d'utilisation (suite)

### Acteur principal et acteur secondaire

- ✓ Un acteur principal produit un résultat observable
- ✓ Un acteur secondaire est sollicité par le système pour des informations secondaires



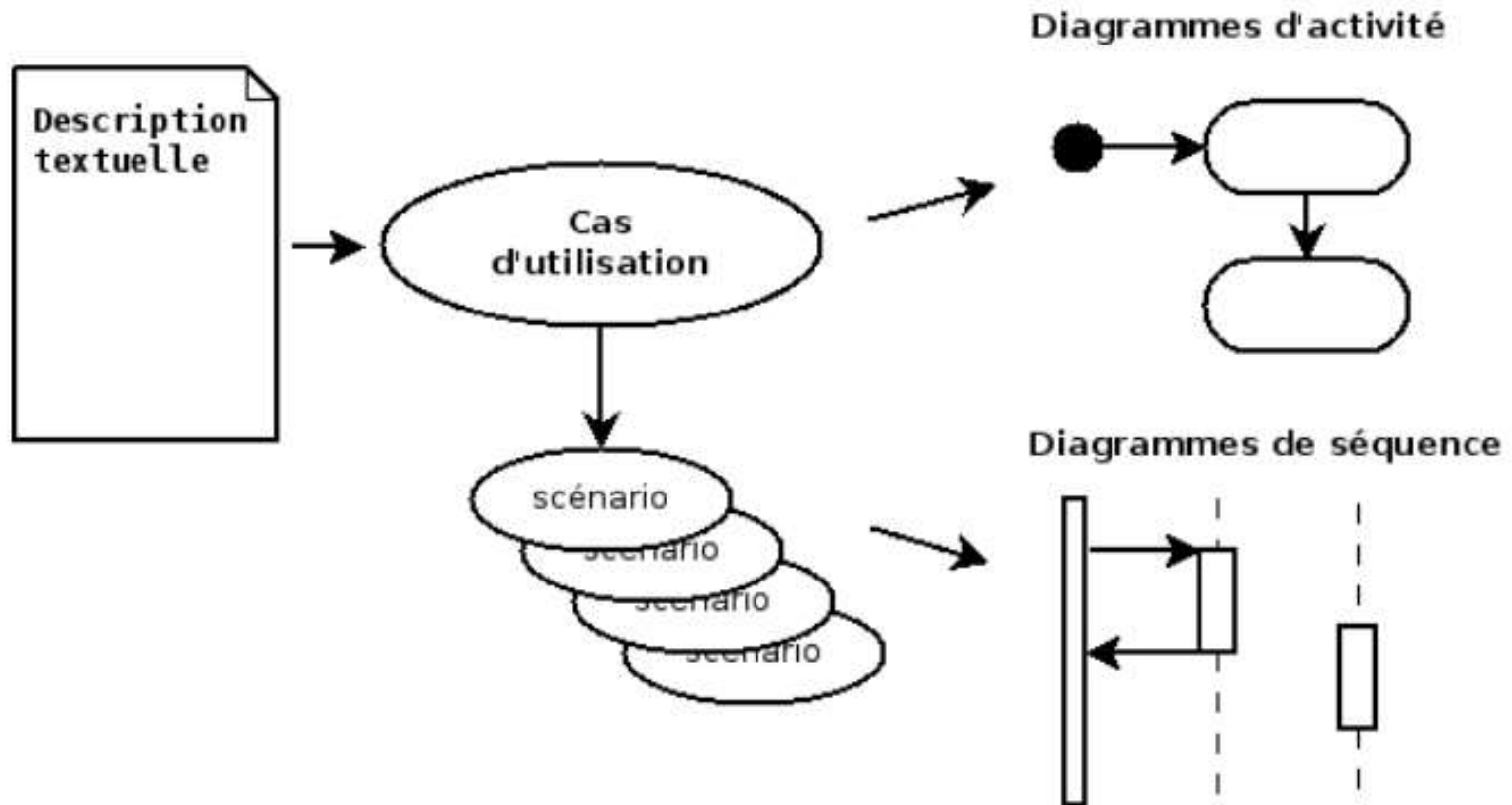
## 4. Diagramme de de cas d'utilisation (suite)

Une fois les cas identifiés, il faut les décrire :

- Description textuelle de toutes les interactions entre les acteurs.
- Début et fin clairement identifiés.
- Variantes possibles :
  - cas nominaux ;
  - alternatives ;
  - cas d'erreur...
- Spécification des préconditions, enchaînements :
  - besoins interfaces homme-machine ;
  - contraintes non fonctionnelles (fréquence, disponibilité, fiabilité, performance)...



## 4. Diagramme de de cas d'utilisation (suite) Démarche



## 5. Diagramme de séquence

### Objectif

Illustrer les données échangées entre utilisateurs et composants dans un exemple particulier d'utilisation du système.

### Corollaire

- Un diagramme de séquence est un scénario.
- Ce n'est pas une description exhaustive du comportement du système.

### Conséquences

- Toujours donner un titre à diagramme de séquence.
- Toujours rattacher un diagramme de séquence à un cas d'utilisation.

## 5. Diagramme de séquence (suite)

### Composants d'un diagramme de séquence:

- ✓ un axe vertical non gradué (implicite), le temps ;
- ✓ des objets et utilisateurs, un par axe
- ✓ des messages (flèches entre objets/utilisateurs) ;
- ✓ des étiquettes

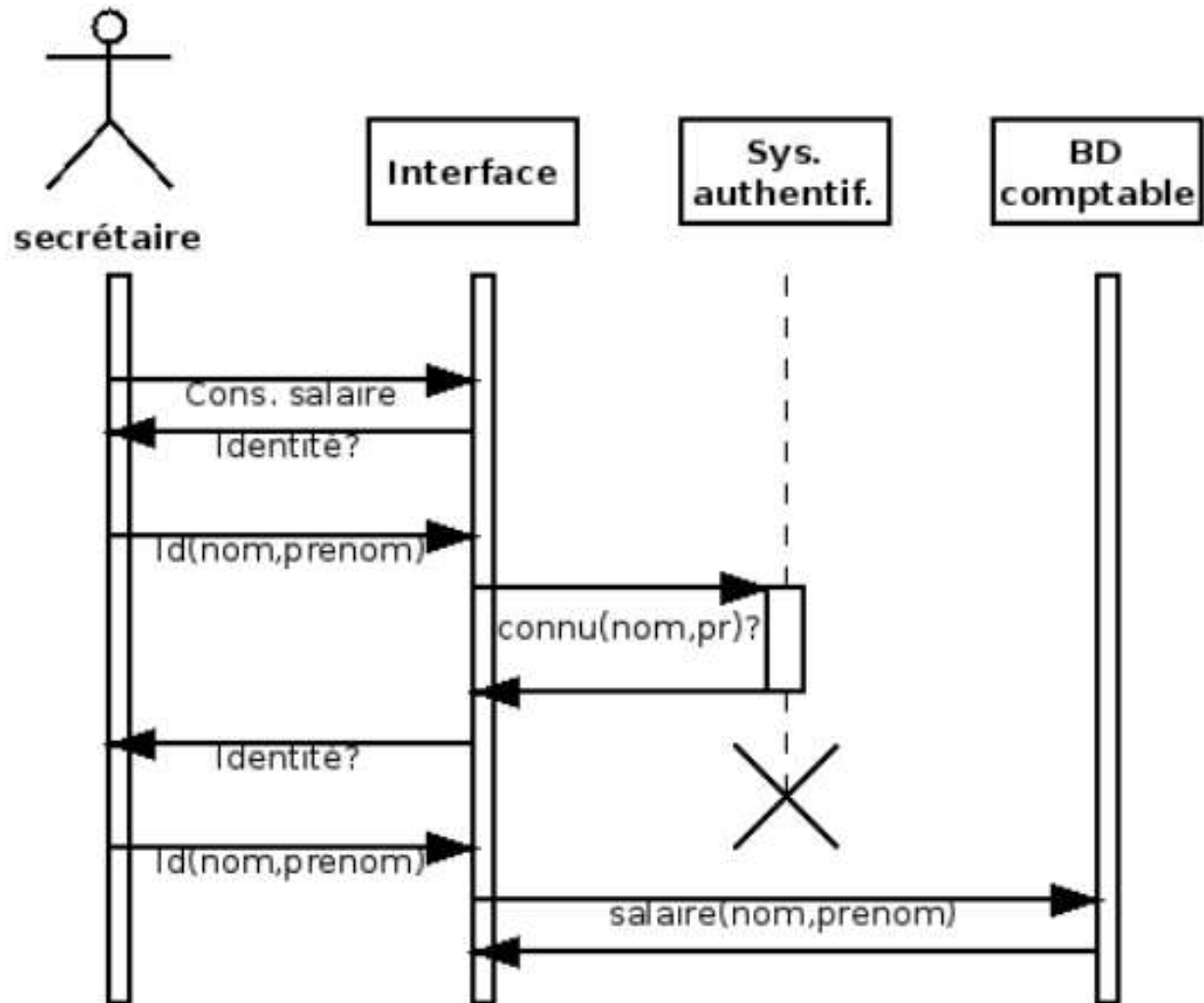
### Un axe est composé de:

- ✓ Un début: quoi, existence
- ✓ Une fin: une croix si l'objet meurt.
- ✓ Un tracé: épais si actif, pointillé si inactif

# Notions de base

## 5. Diagramme de séquence (suite)

### Exemple



***Merci pour votre attention***

