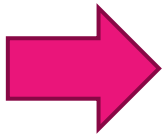
Modélisation

Stéphanie CHOLLET

Rappel : Complexité des logiciels

- Problématique du domaine
- Processus de développement
- Flexibilité inhérente du logiciel

- Conséquences :
  - Le risque de déficiences graves est élevé
  - La mise au point est lente et chaotique
  - La maintenance est disproportionnée
  - Le coût est astronomique



A défaut de réduire la complexité, il faut la maîtriser !

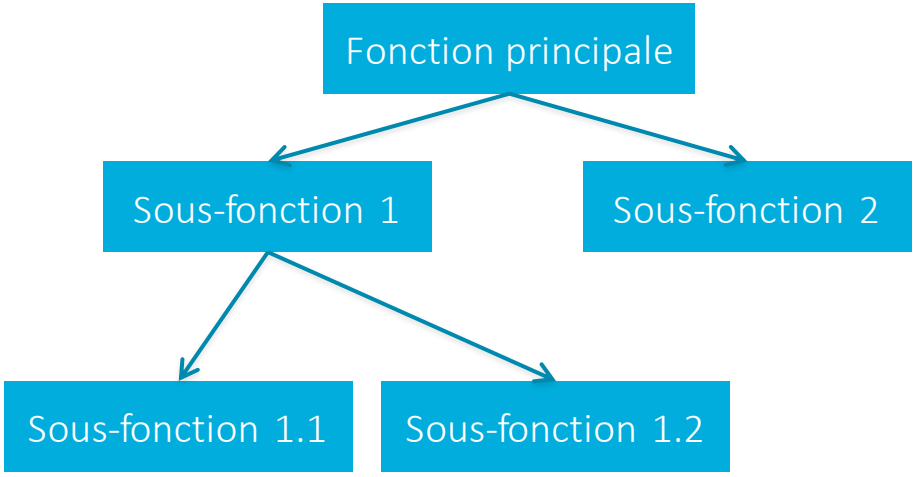
stephanie.chollet@grenoble-inp.fr

La décomposition

- « Diviser pour régner »
  - Raffinage récursif jusqu'à l'obtention d'éléments compréhensibles

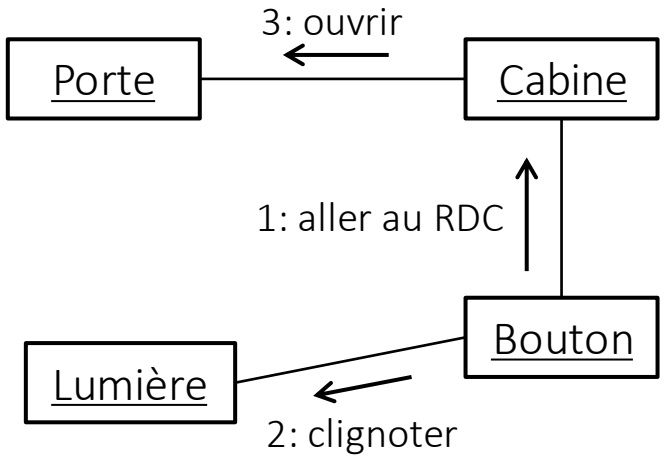
Décomposition algorithmique

- Approche traditionnelle
- Chaque module représente une étape du processus global
- Décomposition fonctionnelle depuis le cahier des charges jusqu'aux sous-programmes



Décomposition objet

- Approche plus récente
- Chaque module représente un objet du domaine de l'application
- Les objets sont des entités autonomes qui collaborent afin de réaliser un projet global



stephanie.chollet@grenoble-inp.fr

Comparaison des approches

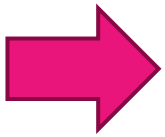
- Approche fonctionnelle :
  - Semble intuitive
  - Mise en avant du « FAIRE »
  - Bien adaptée lorsque tout est connu
- MAIS :
  - Architecture rigide
  - Extension difficile
  - Peu adaptée à la découverte

- Approche objet :
  - Mise en avant de l' « ETRE »
  - Simple (petit nombre de concepts)
  - Raisonnement par abstraction (objets du domaine)
  - Adapté pour l'exploration et l'évolution
- MAIS :
  - Déroutant pour les habitués de la démarche fonctionnelle

stephanie.chollet@grenoble-inp.fr

Avantages de l'objet

- Conduit à des modèles plus stables
  - Basés sur le monde réel
- Structure indépendante des fonctions
  - Evolutivité
- Encapsule la complexité
  - Facilite la réutilisation



L'approche objet gère plus efficacement la complexité
que l'approche fonctionnelle (réutilisabilité, évolutivité, stabilité)

stephanie.chollet@grenoble-inp.fr

La modélisation

Pourquoi modéliser ?

- Permettre une **meilleure compréhension** du monde réel
  - Systèmes réels trop complexes
  - Abstraction des aspects cruciaux du problème
  - Omission des détails

- Exemple : Qu'est ce qu'un terrain de football ?

Stade Vélodrome (Marseille)

Parc des Princes (Paris)

stephanie.chollet@grenoble-inp.fr

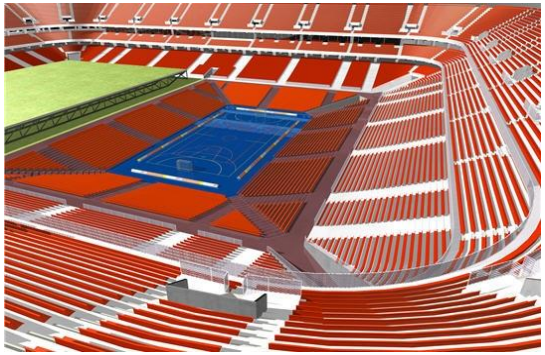
Pourquoi modéliser ?

- Permettre une **conception progressive**
  - Abstractions et raffinements successifs
  - Découpage en modules ou en vues
  - Génération des structures de données et des traitements

- Exemple : Conception par étape d'un stade

Maquette du stade de Lille

Plan des gradins du stade de Lille



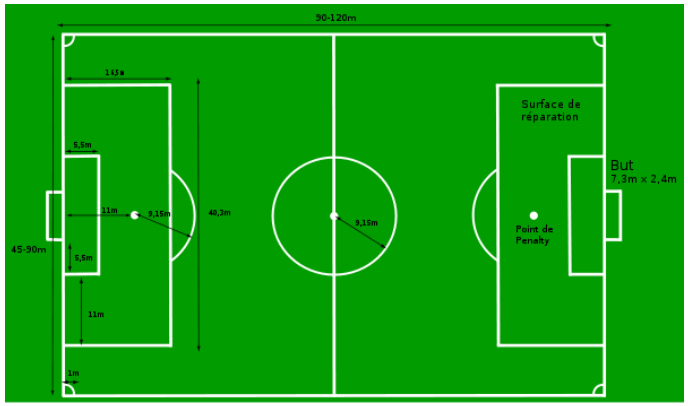
stephanie.chollet@grenoble-inp.fr

Pourquoi modéliser ?

- Faciliter la visualisation du système
  - Diagrammes avec notations simples et lisibles
  - Compréhension visuelle et non seulement intellectuelle

- Exemple : Terrain de football

Modèle d'un terrain de football



stephanie.chollet@grenoble-inp.fr

De quoi a-t-on besoin pour modéliser ?

- Un langage de modélisation
  - Avec une notation claire
  - Avec une sémantique précise

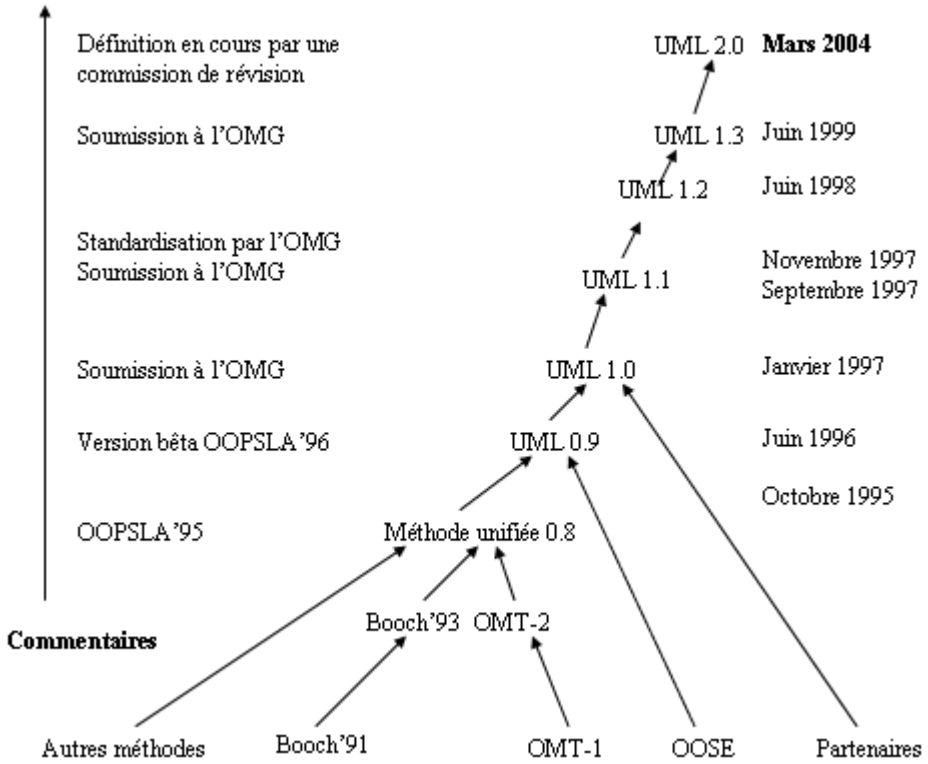
- Caractéristiques d'un langage de modélisation :
  - Générique
  - Expressif
  - Flexible (configurable, extensible)
  - Syntaxe et sémantique non ambiguës

stephanie.chollet@grenoble-inp.fr

UML

- Notations utilisée pour des modélisations orientées objets

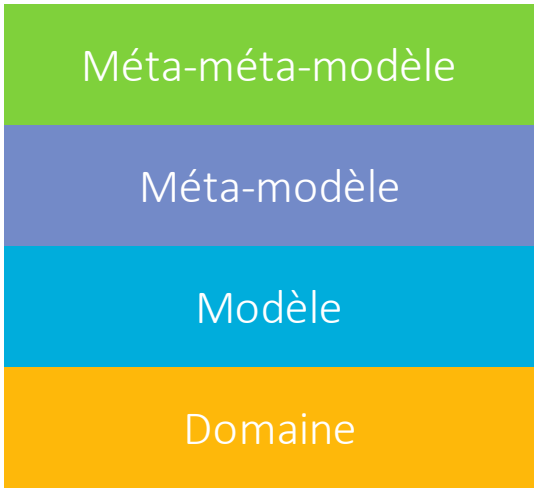
- Notations unifiées basées sur de nombreuses méthodes



stephanie.chollet@grenoble-inp.fr

Objectifs de UML

- Représenter des systèmes entiers
- Etablir un couplage explicite entre les concepts et les artefacts exécutables
- Prendre en compte les facteurs d'échelle
- Créer un langage de modélisation utilisable à la fois par les humains et les machines



stephanie.chollet@grenoble-inp.fr

Contenu de UML

- Pas une notation MAIS des notations UML :
  - Diagramme des cas d'utilisation
  - Diagramme de classes
  - Diagramme d'objets
  - Diagramme de séquence
  - Diagramme de collaboration
  - Diagramme d'états
  - Diagramme d'activités
  - Diagramme de composants
  - Diagrammes de déploiement
  - Langages de contraintes (OCL)
  - Langage d'actions
  - Langage textuel
  - …

stephanie.chollet@grenoble-inp.fr

Classement des diagrammes

Diagrammes structurels

- Diagramme de classes
- Diagramme d'objets
- Diagramme de déploiement
- Diagramme de composants

Diagrammes comportementaux

- Diagramme de cas d'utilisation
- Diagramme de séquence
- Diagramme d'activités
- Diagramme d'états
- Diagramme de collaboration

 Vue statique

 Vue dynamique

stephanie.chollet@grenoble-inp.fr

Caractéristiques des diagrammes

- La plupart des diagrammes se présentent sous la forme de graphes, composés de sommets et d'arcs

- Les diagrammes contiennent des éléments de visualisation qui représentent des éléments de modélisation
  - Les diagrammes peuvent montrer tout ou partie des caractéristiques des éléments de modélisation selon le niveau de détail utile dans le contexte d'un diagramme donné
  - En général, un diagramme représente un aspect ou un point de vue particulier

stephanie.chollet@grenoble-inp.fr

Utilisation des modèles

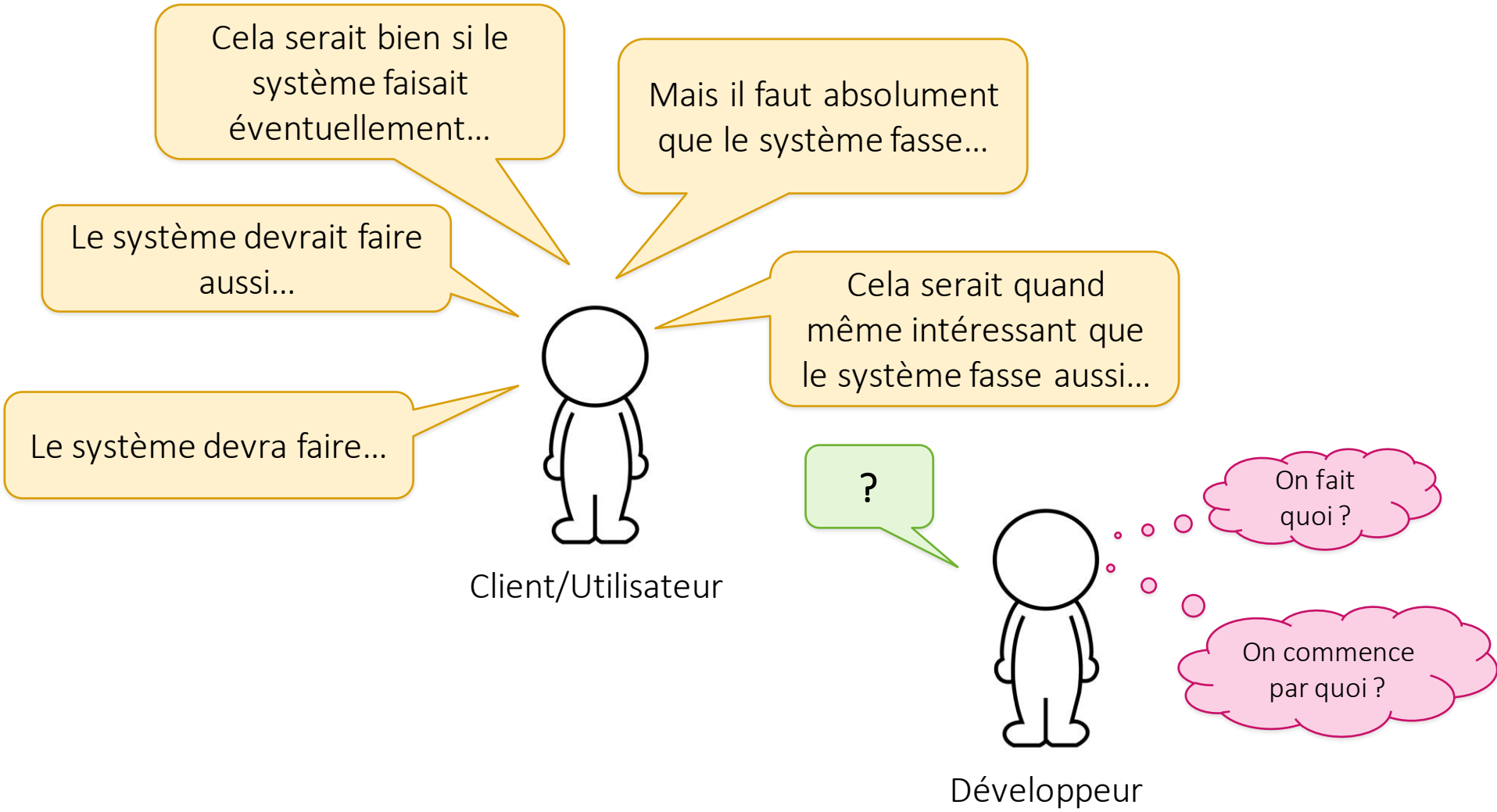
- Pour l'analyse et la conception :
  - Nécessaires pour comprendre le besoin
  - Nécessaires pour définir l'application : son comportement et les données qu'elle manipule
    - Modèles utilisés : Diagrammes de cas d'utilisation, de séquence, de classes, d'objets, d'états…
- Pour l'implantation :
  - Pour définir la structure de l'application
    - Modèles utilisés : Diagrammes de classes, de composants, de déploiement, de collaboration, d'activités…

stephanie.chollet@grenoble-inp.fr

Diagramme de cas d'utilisation

Modélisation des interactions

Mais que faisons-nous ?



stephanie.chollet@grenoble-inp.fr

Diagramme de cas d'utilisation

- Formalisé par Ivar Jacobson

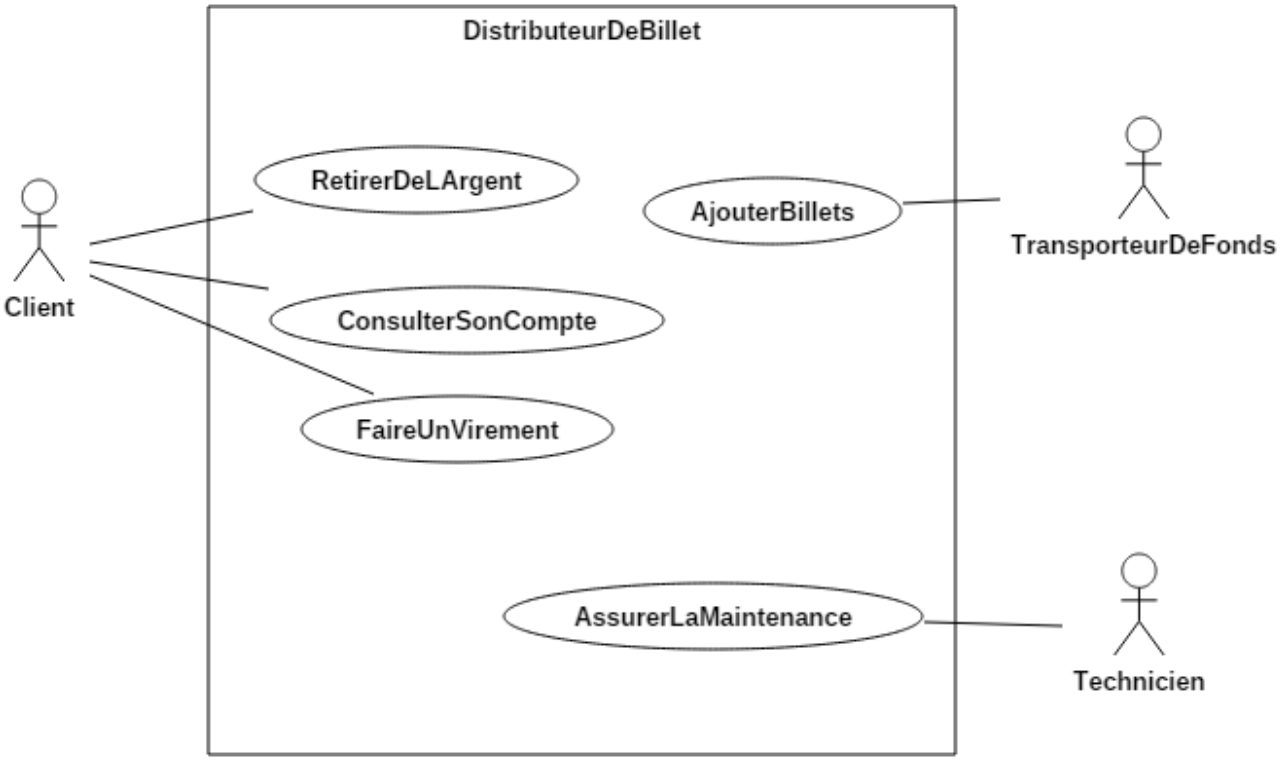
- Objectifs :
  - Décrire, sous la forme d'actions et de réactions, le comportement d'un système du point de vue d'un utilisateur
  - Définir les limites précises du système

- Un cas d'utilisation est :
  - Une manière spécifique d'utiliser le système
  - L'image d'une fonctionnalité du système, déclenchée en réponse à la stimulation d'un acteur externe

stephanie.chollet@grenoble-inp.fr

Exemple de diagramme de cas d'utilisation

- Représente les fonctions du système d'un point de vue des utilisateurs



stephanie.chollet@grenoble-inp.fr

Diagramme de cas d'utilisation

- Un diagramme de cas d'utilisation décrit :
  - Les acteurs
  - Les cas d'utilisation
  - Le système

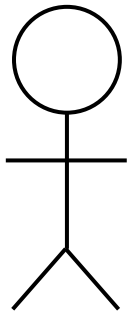
- Un modèle de cas d'utilisation ~~peut~~ doit être formé :
  - De descriptions textuelles
  - De diagrammes de séquences
  - De plusieurs sous-diagrammes de cas d'utilisation
  - …

stephanie.chollet@grenoble-inp.fr

Les acteurs

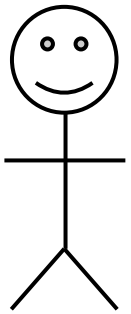
- Un acteur :
  - Est un élément externe du système qui interagit avec le système
  - Prend des décisions, des initiatives. Il est actif.
  - A un rôle qu'un utilisateur joue par rapport au système

- Notation graphique :



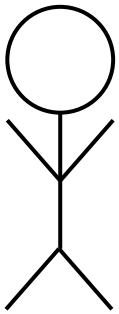
Client

OUI



Client

NON !



Client

NON !

stephanie.chollet@grenoble-inp.fr

Acteur vs. Utilisateur

Ne pas confondre la notion d'acteur et de personne utilisant le système

- Une même personne peut jouer plusieurs rôles
  - Exemple : Maurice est directeur mais peut jouer le rôle de guichetier
- Plusieurs personnes peuvent jouer un même rôle
  - Exemple : Paul et Pierre sont deux clients
- Un acteur n'est pas forcément un être humain
  - Exemple : Un distributeur de billets peut être un acteur

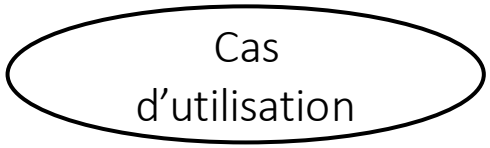
stephanie.chollet@grenoble-inp.fr

Les cas d'utilisation

- Un cas d'utilisation :
  - Est une manière d'utiliser le système
  - Est une suite d'interactions entre un acteur et le système

- Correspond à une fonction **visible par l'acteur**
- Permet d'atteindre **un but** pour un acteur
- Doit être **utile en soi**
- Regroupe un **ensemble de scénarios** correspondant à un même but

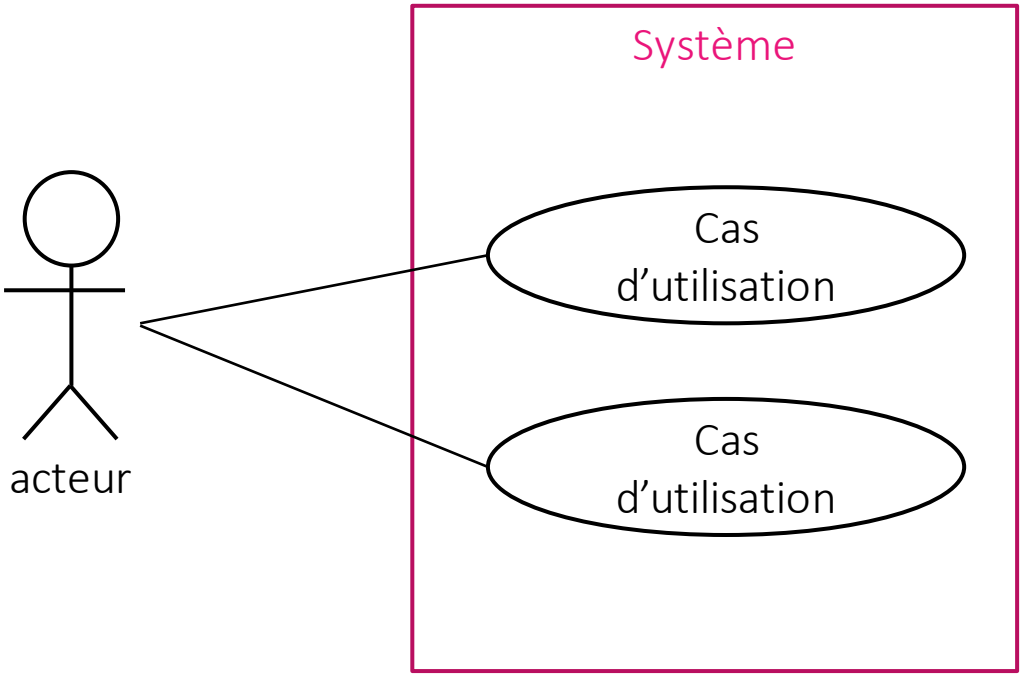
- Notation graphique :



stephanie.chollet@grenoble-inp.fr

Le système

- Le système est l'ensemble des cas d'utilisation
- **Mais** ne contient pas les acteurs



stephanie.chollet@grenoble-inp.fr

Informations textuelles

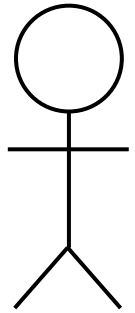
- Tout diagramme de cas d'utilisation doit être accompagné d'informations textuelles.

- Pourquoi ?
  - Pour expliquer les cas d'utilisation
  - Pour préciser certains éléments qui ne peuvent pas être représentés graphiquement
  - Pour faciliter la communication avec le client de l'application

stephanie.chollet@grenoble-inp.fr

Informations textuelles pour les acteurs

- Pour chaque acteur :
  - Choisir un identificateur représentatif de son rôle
  - Donner une brève description contextuelle



Guichetier

Un guichetier est un employé de la banque chargé de faire l'interface entre le système informatique et les clients qu'il reçoit au comptoir. Le guichetier peut réaliser des opérations courantes : création d'un compte, dépôt ou retrait d'argent.

stephanie.chollet@grenoble-inp.fr

Informations textuelles pour les cas d'utilisation

- Pour chaque cas d'utilisation :
  - Choisir un identificateur représentatif
  - Donner une description textuelle simple
  - La fonction réalisée doit être comprise de tous
  - Préciser ce que fait le système, ce que fait l'acteur
  - Préciser quelles sont les pré-conditions et/ou les post-conditions
  - Préciser éventuellement les cas d'erreur

Les cas d'utilisation ne doivent pas se chevaucher !

stephanie.chollet@grenoble-inp.fr

Informations textuelles pour les cas d'utilisation

- Généralement sous la forme d'un tableau
- Exemple :

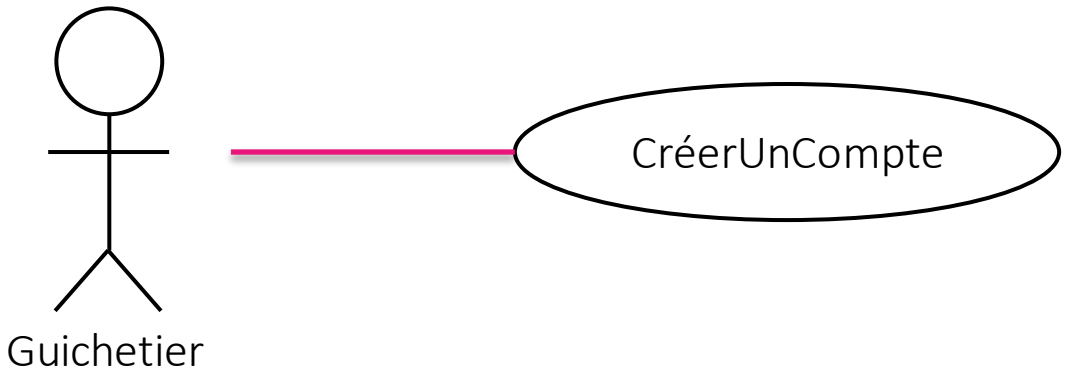
Nom du cas d'utilisation	
But	
Acteur	
Pré-conditions	
Scénario nominal	
Post-conditions	
Exceptions	

stephanie.chollet@grenoble-inp.fr

Relation Acteur-Cas d'utilisation

- Une relation entre acteur et cas d'utilisation représente une communication initiée par l'acteur. Les messages échangés peuvent potentiellement être dans les deux sens.

- Exemple :



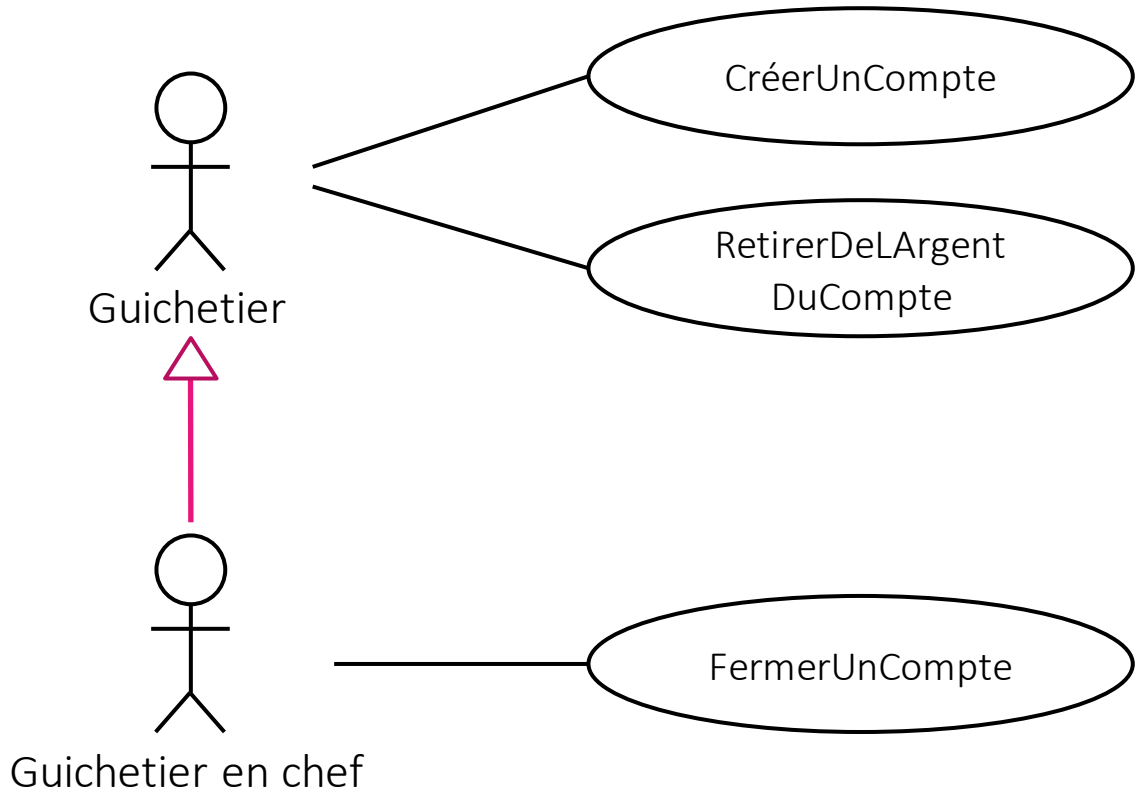
  - La relation sera raffinée par la suite (*Spécifications externes*) :
    - Si l'acteur est humain : interface homme – système
    - Si l'acteur est un logiciel : interface logicielle

stephanie.chollet@grenoble-inp.fr

Relation entre acteurs

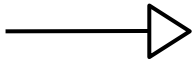
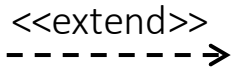
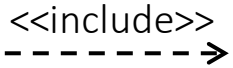
- La seule relation entre acteurs est la relation de généralisation

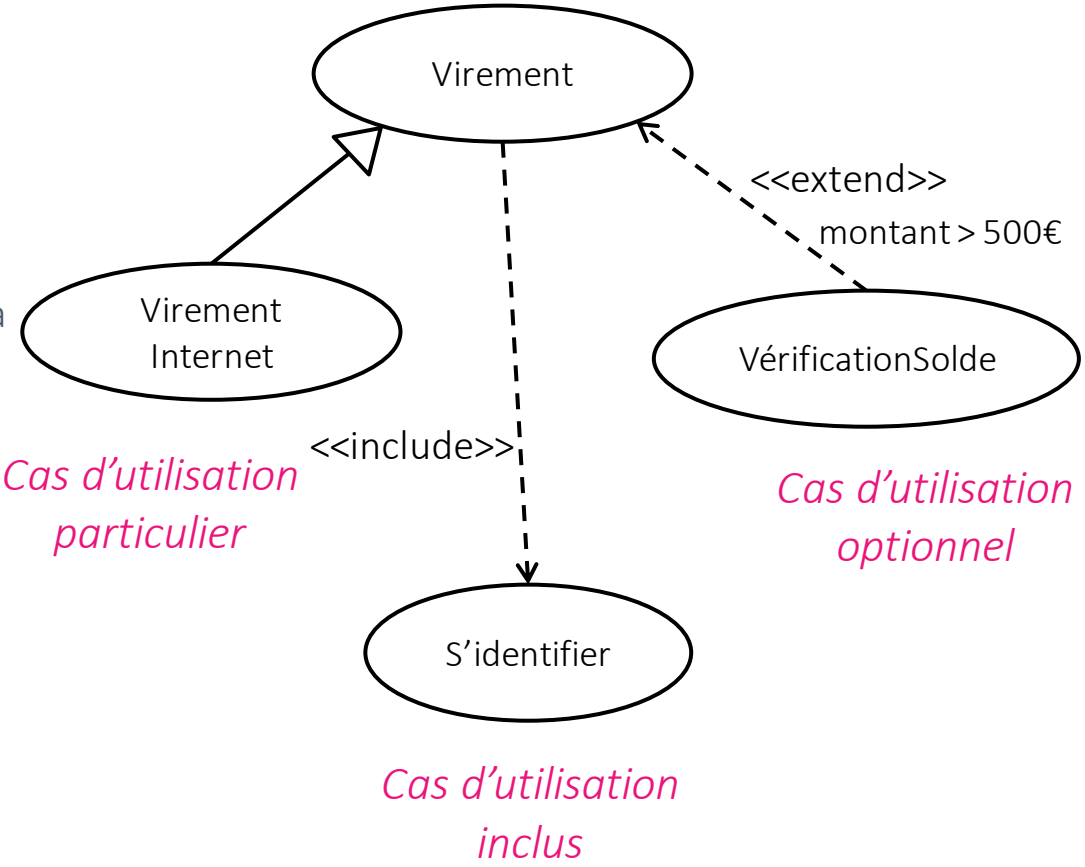
- Exemple :



stephanie.chollet@grenoble-inp.fr

Relations entre cas d'utilisation

- Généralisation : 
  - A utiliser pour spécialiser un comportement général
- Extension : 
  - À utiliser quand il y a un comportement supplémentaire à ajouter pour le comportement défini dans un ou plusieurs cas d'utilisation
- Inclusion : 
  - À utiliser lorsqu'il y a des parties communes de comportement pour deux ou plus cas d'utilisation.



stephanie.chollet@grenoble-inp.fr

Conseils pratiques

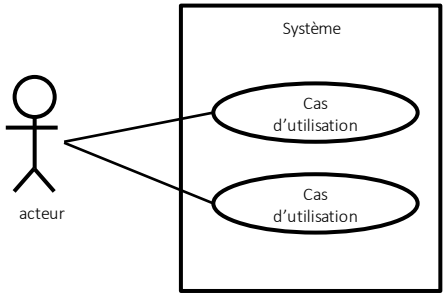
- Il n'y a pas de manière mécanique et totalement objective de repérer les cas d'utilisation :
  - Il faut bien déterminer les limites du système
  - Il faut se placer du point de vue de chaque acteur et déterminer comment il se sert du système
  - Il faut limiter le nombre de cas d'utilisation en choisissant le bon niveau d'abstraction

- Trouver le bon niveau de détail pour les cas d'utilisation est un problème difficile qui nécessite de l'expérience.

stephanie.chollet@grenoble-inp.fr

Extrait du dossier de spécifications – 1/2

1. Diagramme des cas d'utilisation général



   1. Description des acteurs
   2. Description des cas d'utilisation

x nbUseCase

CU100 Cas d'utilisation	
But	
Acteur	
Pré-conditions	
Scénario nominal	
Post-conditions	
Exceptions	

+ diagramme de séquence

2. Description détaillée des cas d'utilisation

x nbSousUseCase

CU100 Cas d'utilisation	
But	
Acteur	
Pré-conditions	
Scénario nominal	
Post-conditions	
Exceptions	

+ diagramme de séquence

3. Récapitulatif des cas d'utilisation

Identifiant	Tâche	Description	Priorité
CUXXX			Must Should May

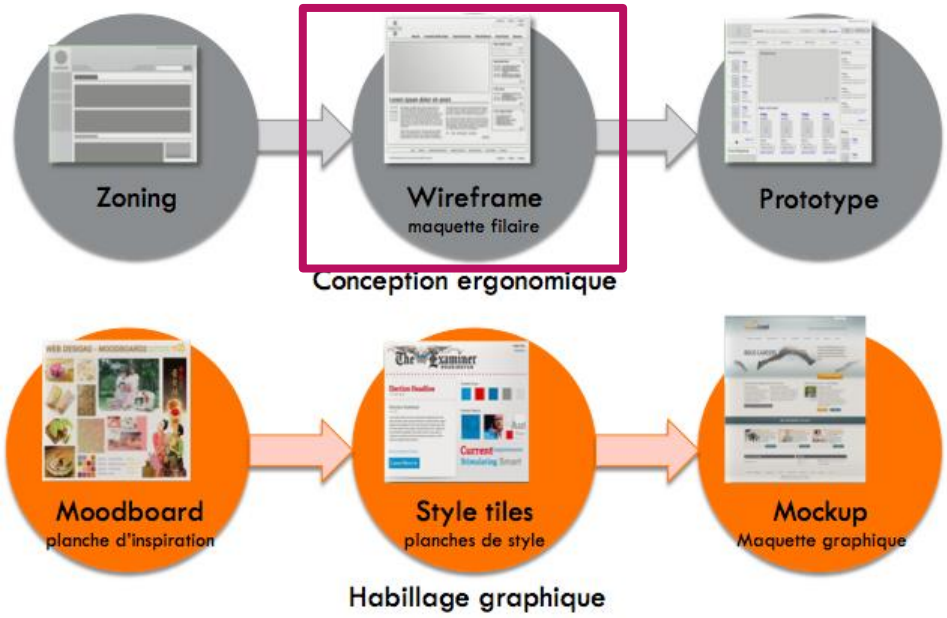
stephanie.chollet@grenoble-inp.fr

Extrait du dossier de spécifications – 2/2

4. Spécification des interfaces de communication

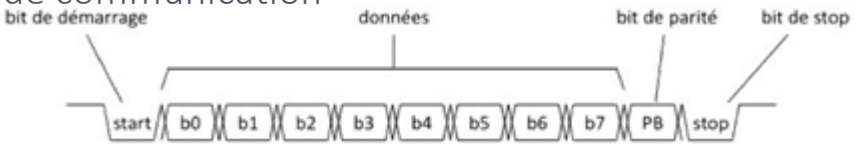
   1. Interfaces Homme/Système (IHM)

   Définition de wireframes

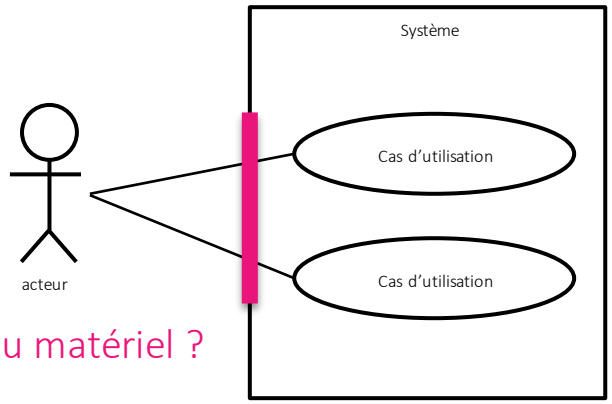


2. Interfaces logicielle (API)

Définition des protocoles et des interfaces de communication




```
"userID" : 1,
"isActive": false,
"identity":
{
        "lastName" : "Doe",
        "firstName" : "John",
        "birthDate" : "01/02/1975",
        "socialSecurityNumber" : "331-80-5651"
},
"address":
{
        "number" : "01",
        "street" : "2068 Thomas Street",
        "city" : "Roselle",
        "zipCode" : "60172",
        "state" : "IL"
},
```



Humain ou matériel ?

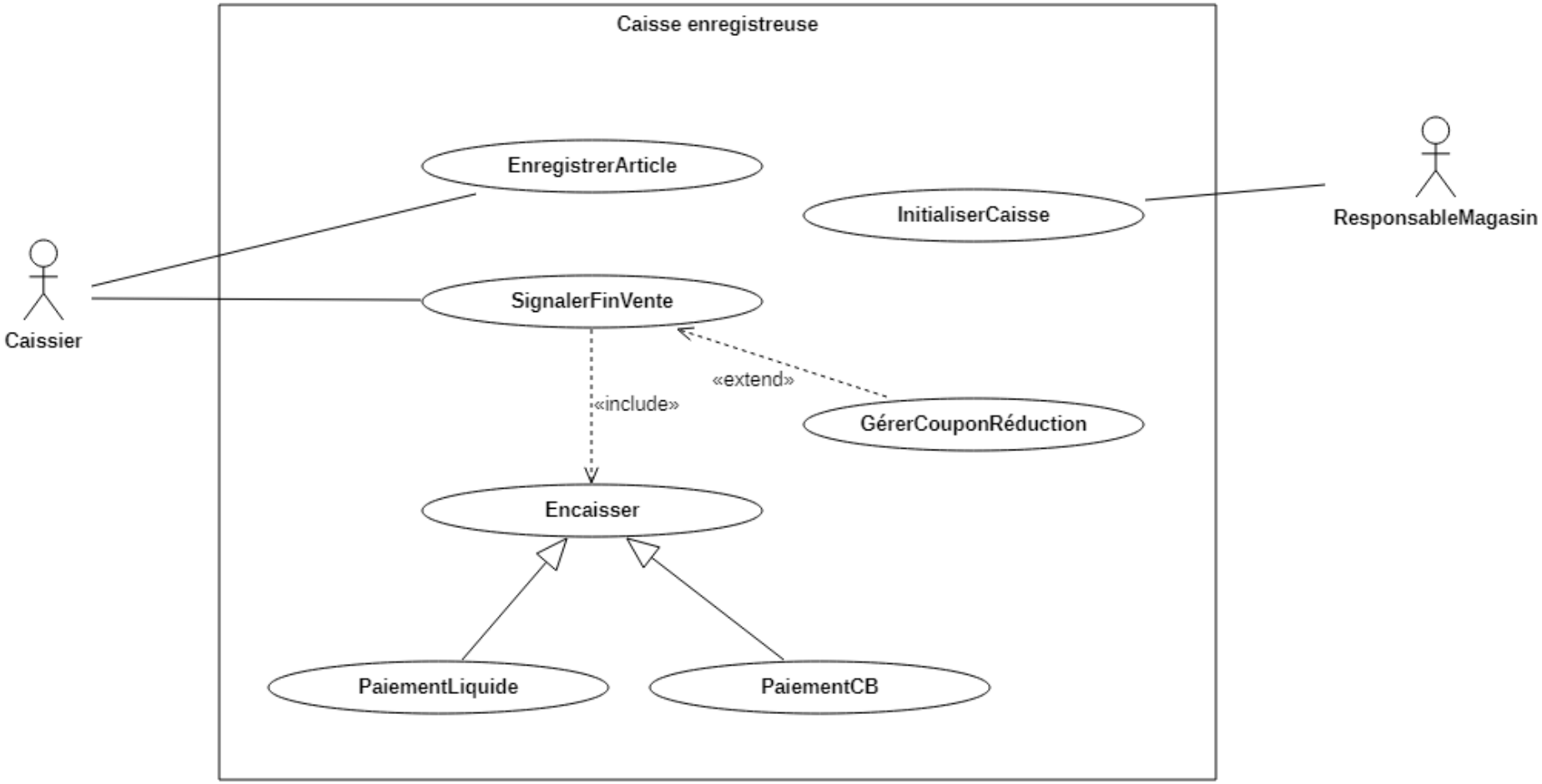
stephanie.chollet@grenoble-inp.fr

Exercice

1. Un client arrive à la caisse avec des articles.
2. Le caissier enregistre le numéro d'identification de chaque article, ainsi que la quantité si celle-ci est supérieure à 1.
3. La caisse affiche le prix de chaque article et son libellé.
4. Lorsque tous les articles ont été enregistrés, le caissier signale la fin de la vente.
5. La caisse affiche le total des achats.
6. Le client choisit son mode de paiement :
    1. Liquide : le caissier encaisse l'argent et la caisse indique le montant éventuel à rendre au client.
    2. Carte de crédit : un terminal bancaire fait partie de la caisse, il transmet la demande à un centre d'autorisation multi-banques.
7. La caisse enregistre la vente et imprime un ticket.
8. Un client peut présenter des coupons de réduction avant le paiement.
9. Lorsque le paiement est terminé, la caisse transmet les informations relatives aux articles vendus au système de gestion des stocks.
10. Tous les matins, le responsable du magasin initialise les caisses pour la journée.

stephanie.chollet@grenoble-inp.fr

Exercice – Diagramme de cas d'utilisation



stephanie.chollet@grenoble-inp.fr

Exercice – Description d'un cas d'utilisation

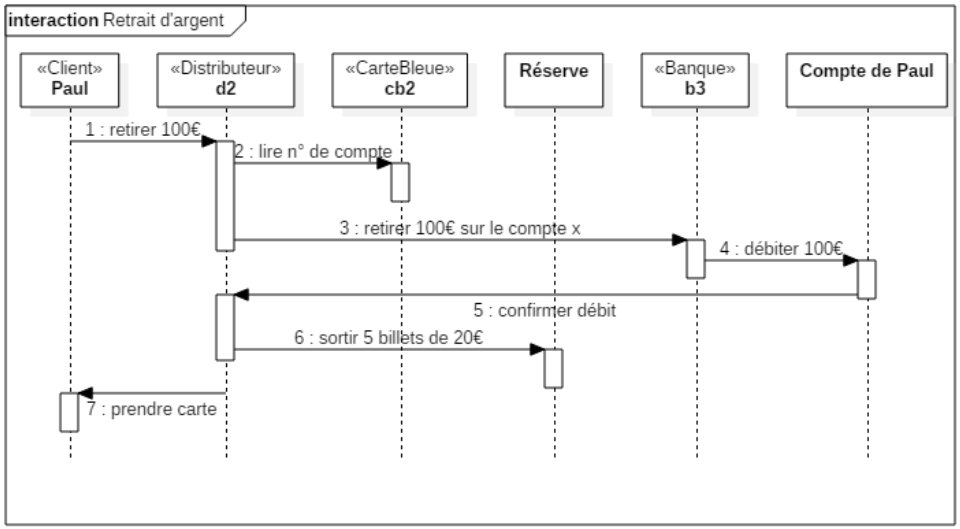
stephanie.chollet@grenoble-inp.fr

Diagramme de séquence

Modélisation des interactions

Diagramme de séquence

- Objectifs :
  - Décrire un scénario en explicitant les interactions entre objets/acteurs sous la forme d'envoi de messages
- Représentation des aspects dynamiques des systèmes d'un point de vue temporel
- A utiliser :
  - Pour la documentation des cas d'utilisation
  - Pour la représentation des interactions entre les objets

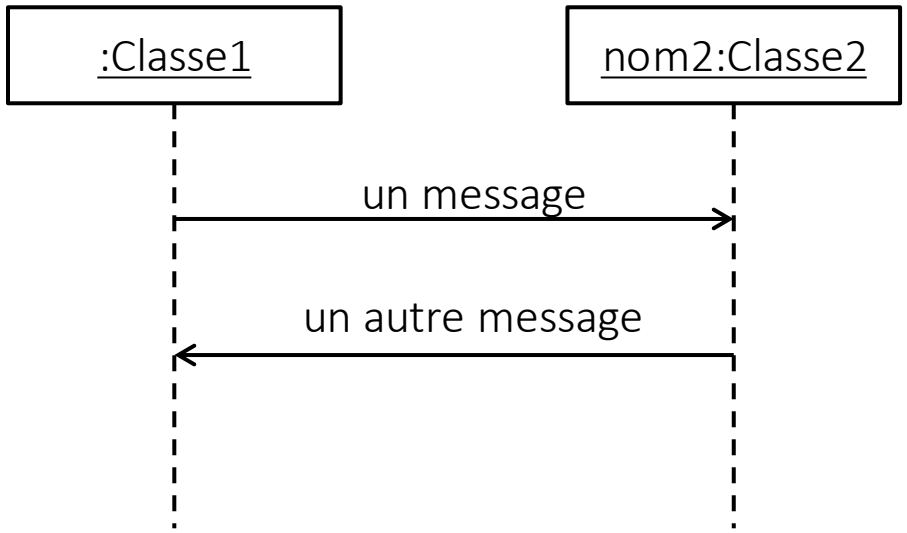


stephanie.chollet@grenoble-inp.fr

Les interactions

- Une interaction modélise un comportement dynamique entre objets. Elle se traduit par l'envoi de message entre objets en insistant sur la chronologie.

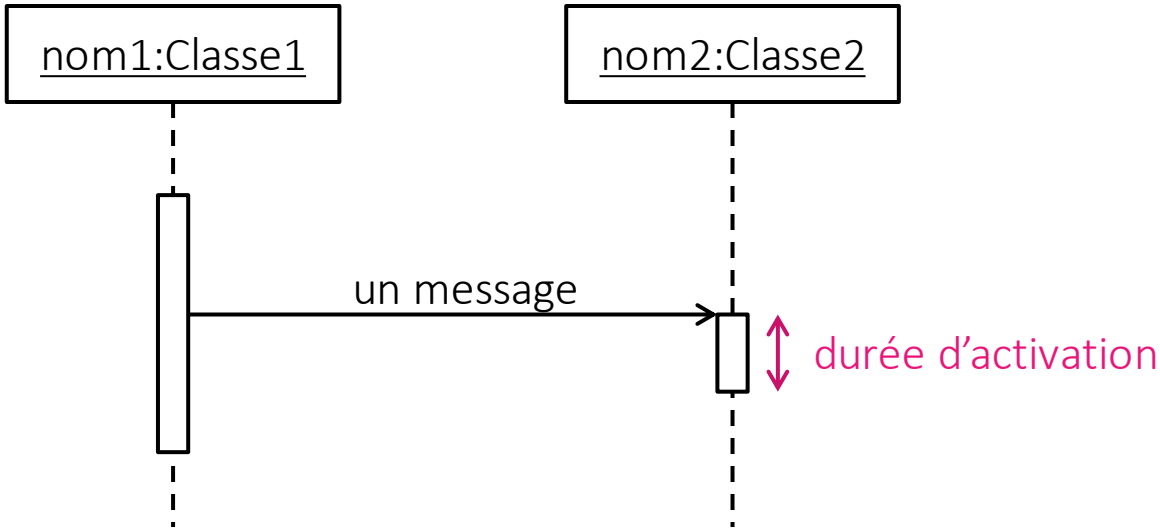
- Représentation :
  - Un objet/acteur est matérialisé par un rectangle et une barre verticale, appelée ligne de vie
  - Les messages sont représentés par une flèche horizontale, orientées de l'émetteur vers le destinataire
  - La dimension verticale représente l'écoulement du temps



stephanie.chollet@grenoble-inp.fr

Les activations

- Une période d'activité correspond au temps pendant lequel un objet effectue une action :
  - soit directement
  - soit par l'intermédiaire d'un autre objet qui lui sert de sous-traitant



stephanie.chollet@grenoble-inp.fr

Conseils pratiques

- Au moins un scénario par cas d'utilisation
- Décomposer les interactions de grande taille en sous-tâches
  - Un diagramme de séquence par sous-tâche
- Un diagramme de séquence par condition d'erreur

stephanie.chollet@grenoble-inp.fr

Diagramme d’états transitions

Diagramme d'états transitions

- Objectif :
  - Modéliser l'évolution d'un objet suite à un événement extérieur

- Equivalent aux automates à états finis/machines à états

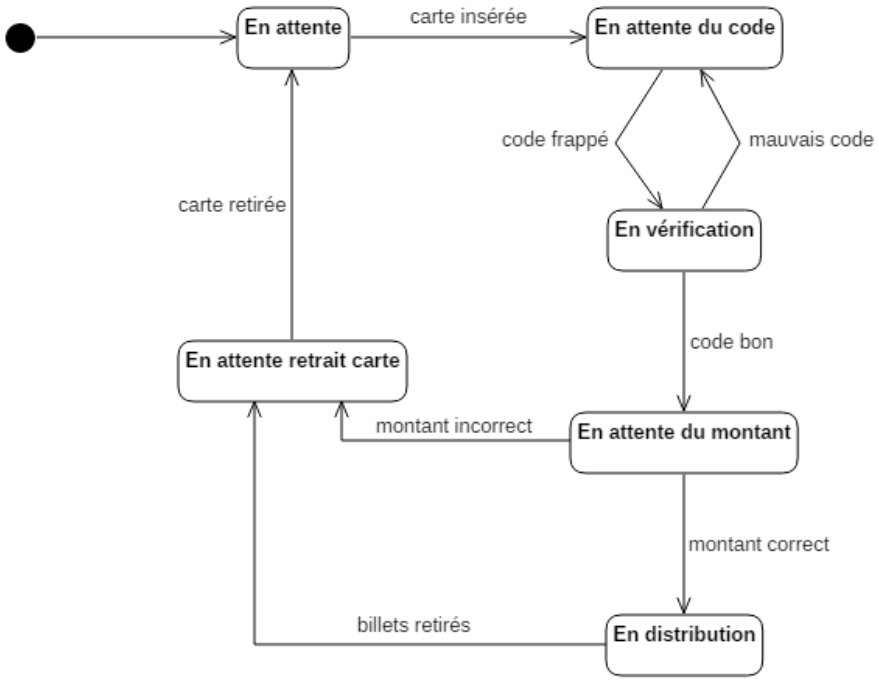
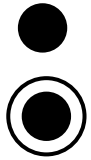


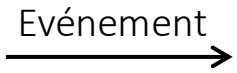
Diagramme d'états transitions

- Deux états particuliers :
  - Initial : représente le point à partir duquel l'objet existe
  - Final : représente l'activité finie



- Etat :
  - Représente l'état d'un objet

- Transition :
  - Représente le passage d'un état à un autre
  - Doit être annoté d'un label précisant l'événement qui déclenche la transition



Exercice

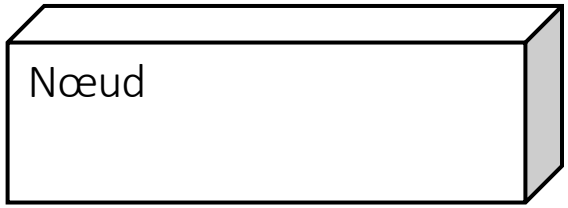
- Modéliser avec un diagramme d'états transitions le fonctionnement d'un téléphone.

Les autres diagrammes

Diagramme de déploiement – 1/2

- Objectif :
  - Représente la disposition physique des différents matériels qui entrent dans la composition du système ainsi que la répartition des composants sur ces dispositifs matériels

- Eléments du diagramme :
  - Nœud : ressource de traitement en cours d'exécution



  - Association : chemin de communication entre des nœuds

Diagramme de déploiement – 2/2

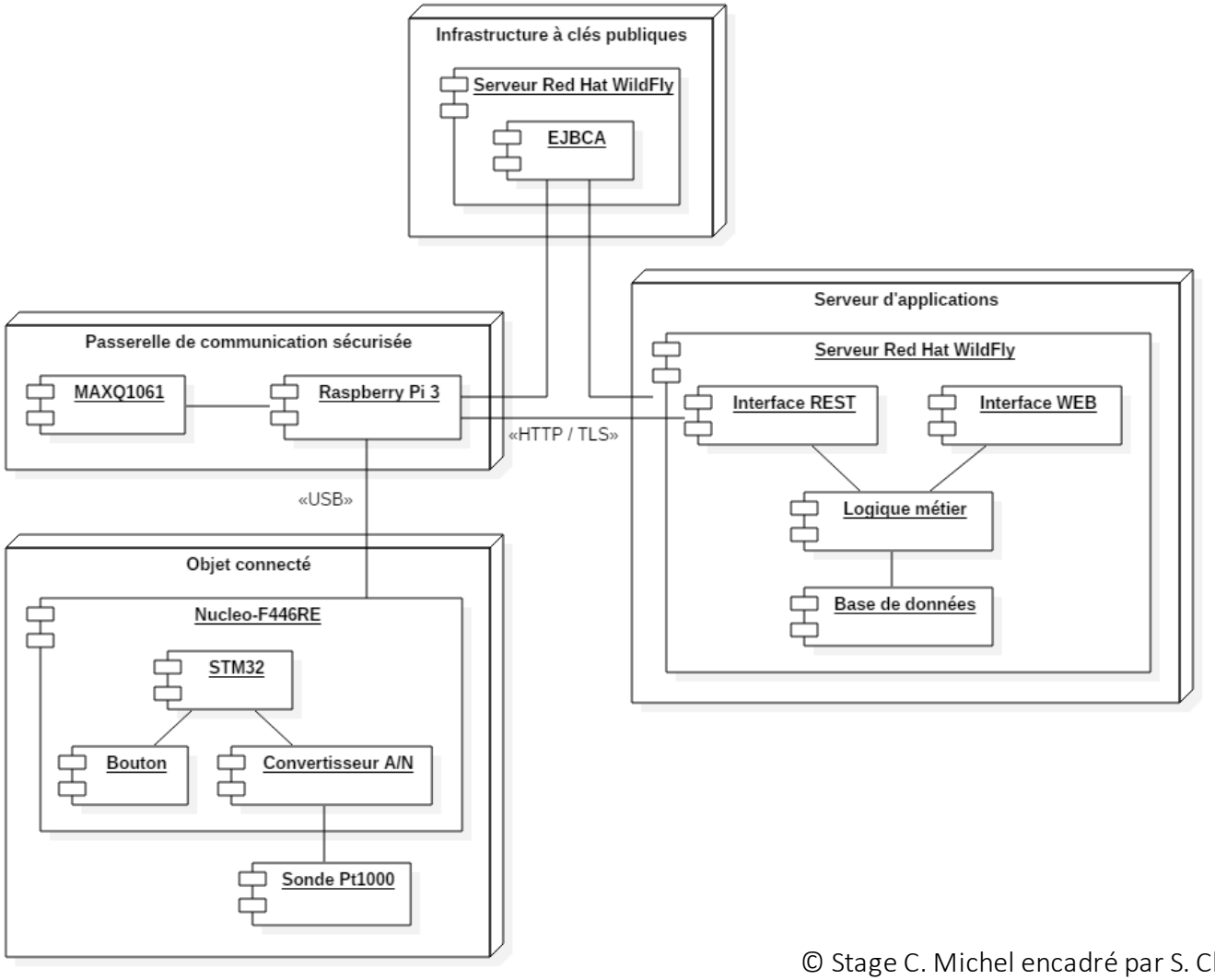
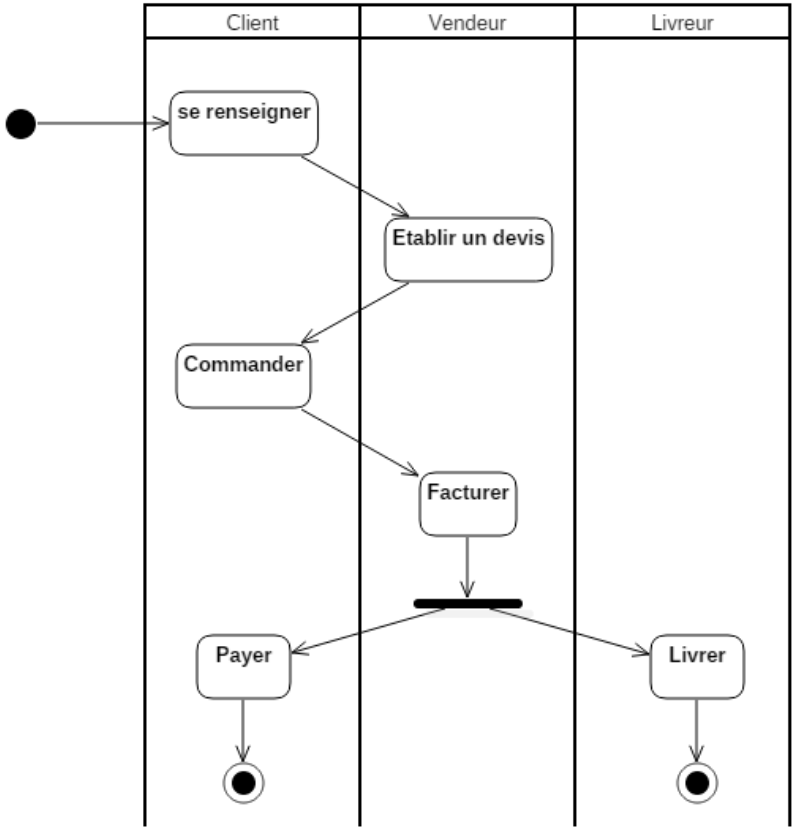


Diagramme d'activités

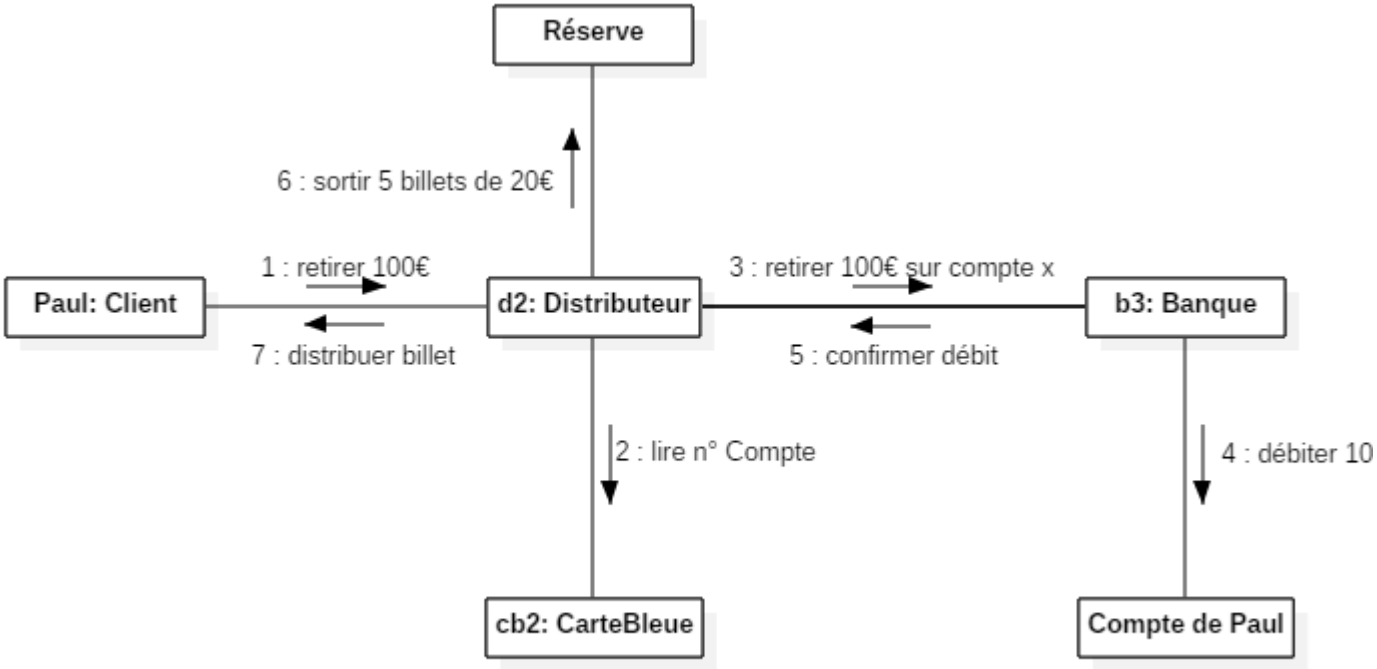
- Représente le comportement d'une méthode ou d'un cas d'utilisation ou d'un processus métier



stephanie.chollet@grenoble-inp.fr

Diagramme de collaboration

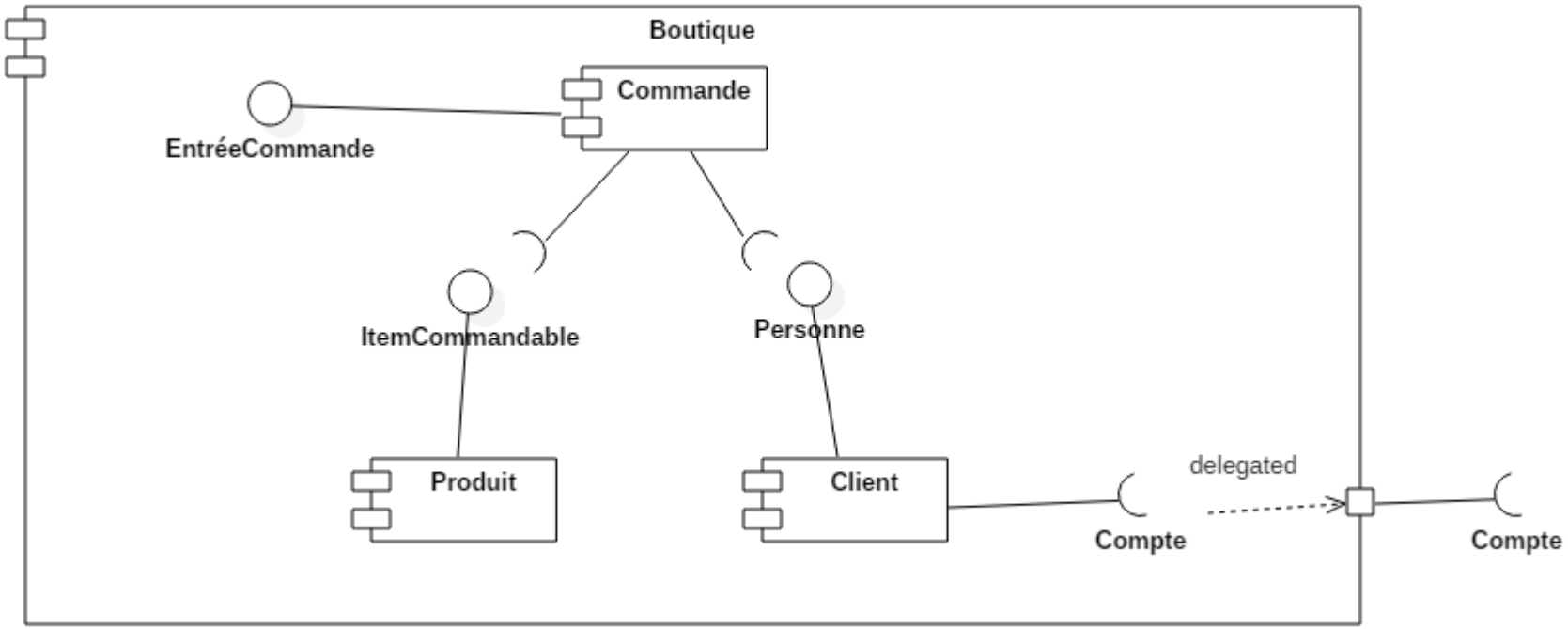
- Objectif :
  - Représentation spatiale des objets, des liens et des interactions



stephanie.chollet@grenoble-inp.fr

Diagramme de composants

- Objectif :
  - Représente les composants (ex : EJB, OSGi…) d'une application



stephanie.chollet@grenoble-inp.fr