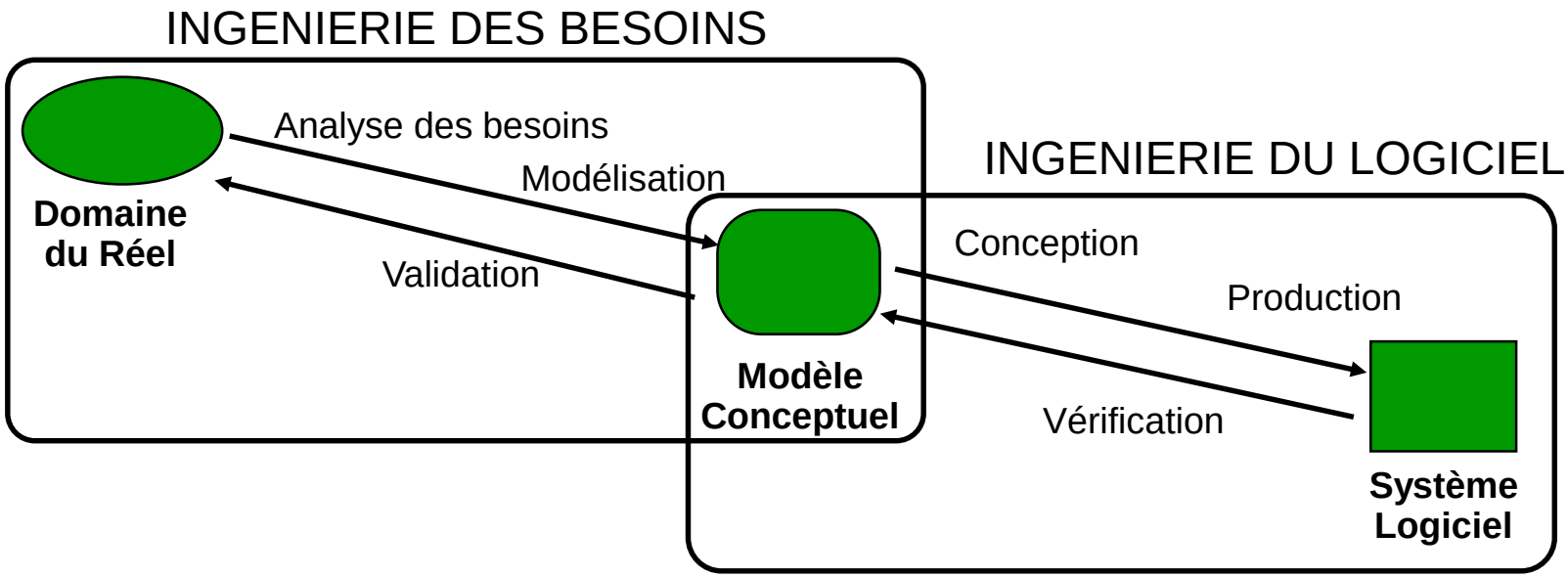
Introduction

L' Ingénierie des Systèmes vise à transformer les besoins des utilisateurs en spécifications formalisés d'une future application



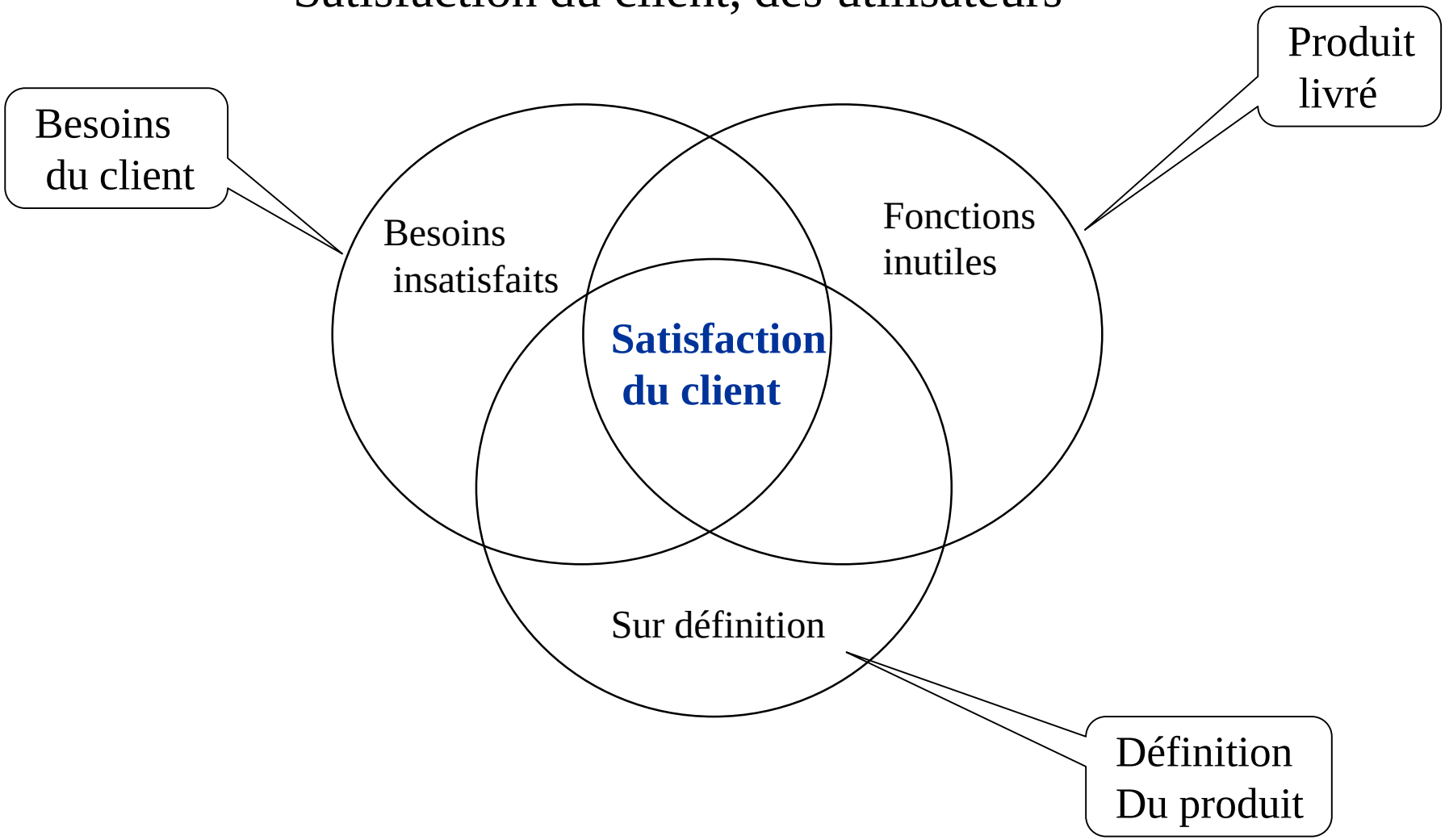
- La réalisation d'un logiciel est un processus long, complexe et coûteux

- Particularité de ce produit : un objet immatériel, malléable, ne s'use pas, peut devenir obsolète par rapport au contexte technique, aux concurrents, etc. Enfin, les défaillances proviennent d'erreurs humaines (peu prévisibles).

Introduction

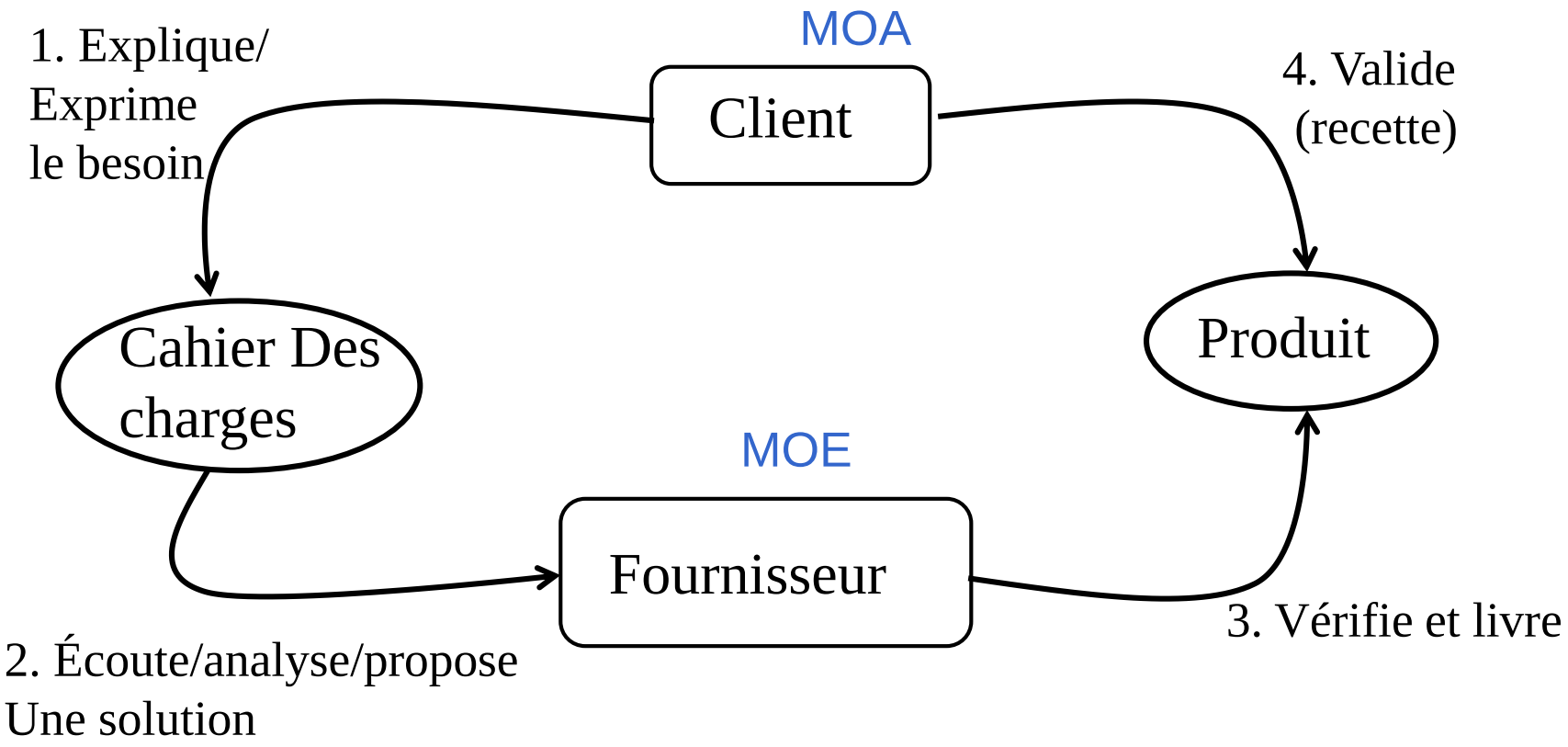
Enjeu Principal

Satisfaction du client, des utilisateurs



Introduction

Les exigences au cœur de la relation Client- Fournisseur



Plan

- Contexte
- Les approches d'I.E.
- Le processus I.E.
- Cas d'application : APB

Contexte

Ingénierie de logiciel/

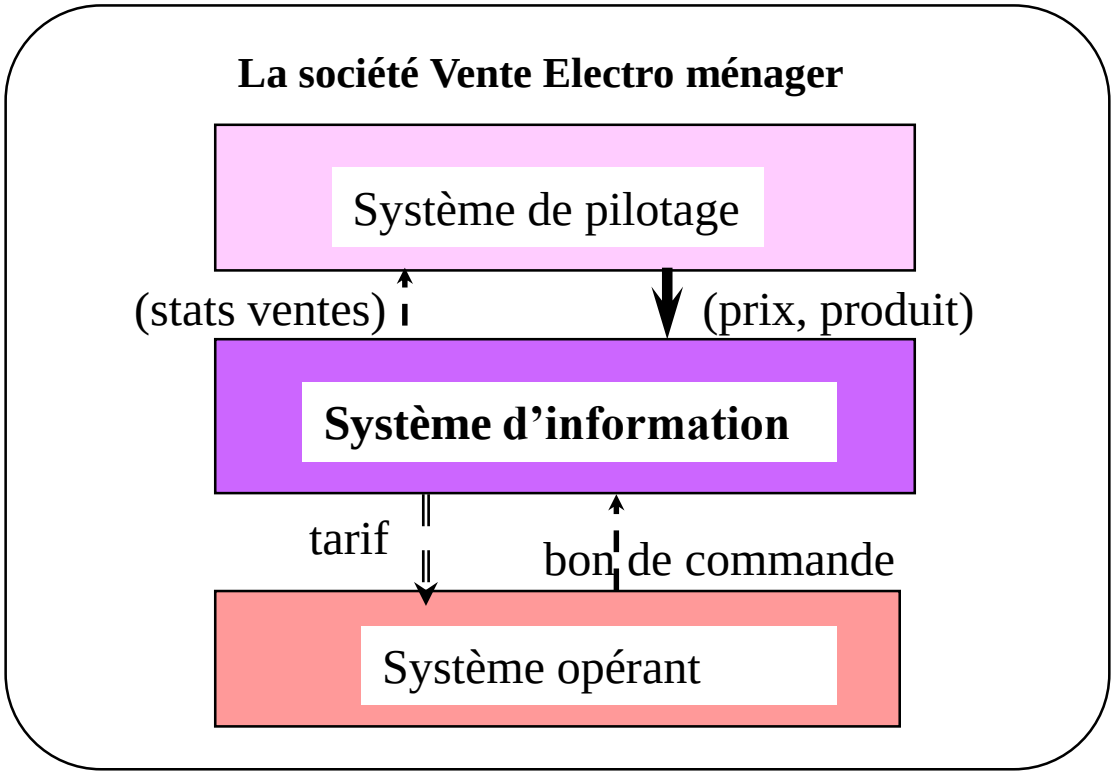
Le logiciel : un ensemble des programmes et procédures nécessaires au fonctionnement d'un système informatique

Le génie logiciel (Software Engineering en anglais) :
ensemble des méthodes, techniques et outils pour la production, l'exploitation et la maintenance de logiciel

Le produit Logiciel : des codes sources et exécutables, des programmes tests, des fichiers de configuration, des documents internes et externes à destination de l'équipe de développement et des utilisateurs

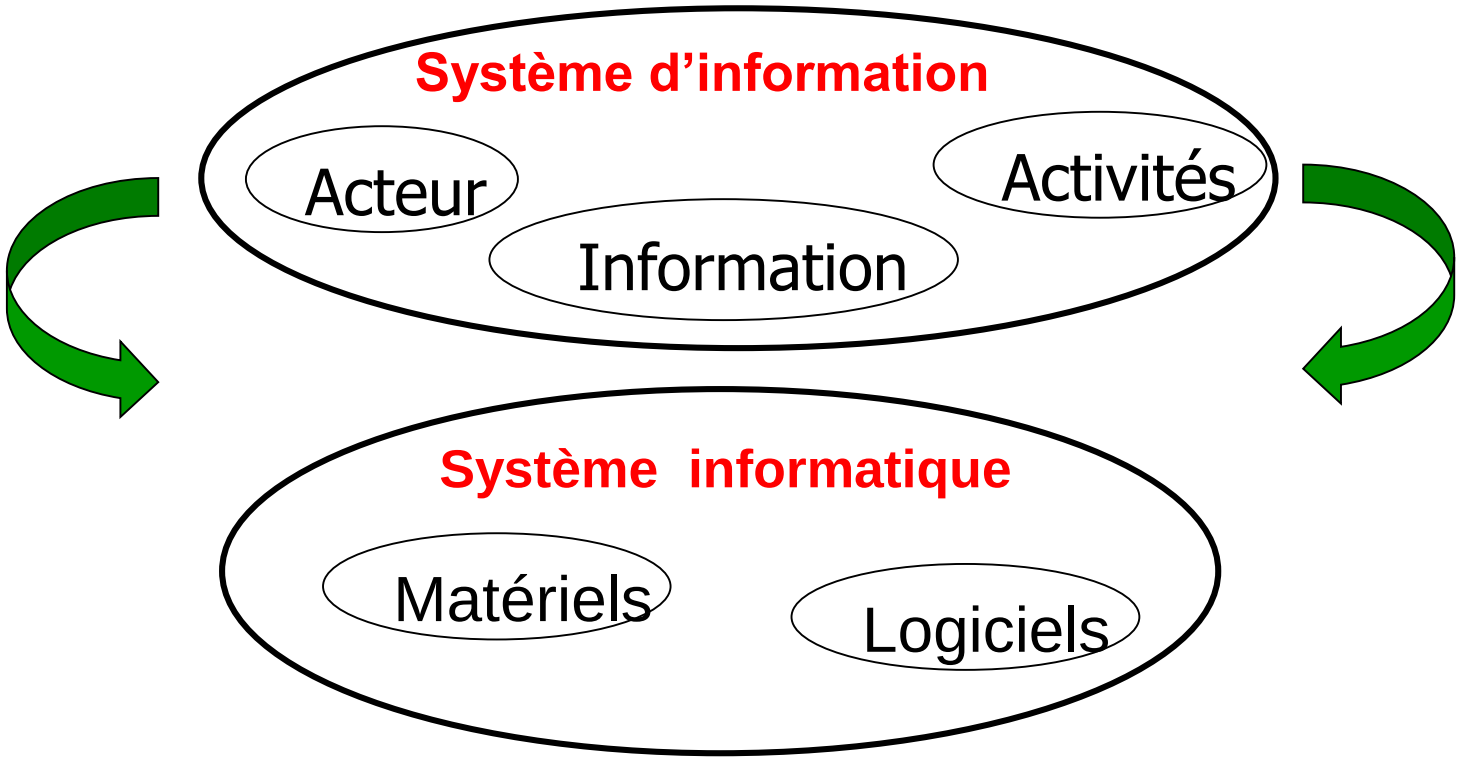
Contexte

- Définition: Un S.I. est un ensemble de ressources (humaines, organisationnelles, matérielles et logicielles) permettant de gérer (saisir, stocker, traiter, restituer) les informations utiles aux décideurs et opérationnels d'une organisation (Rex)



Contexte

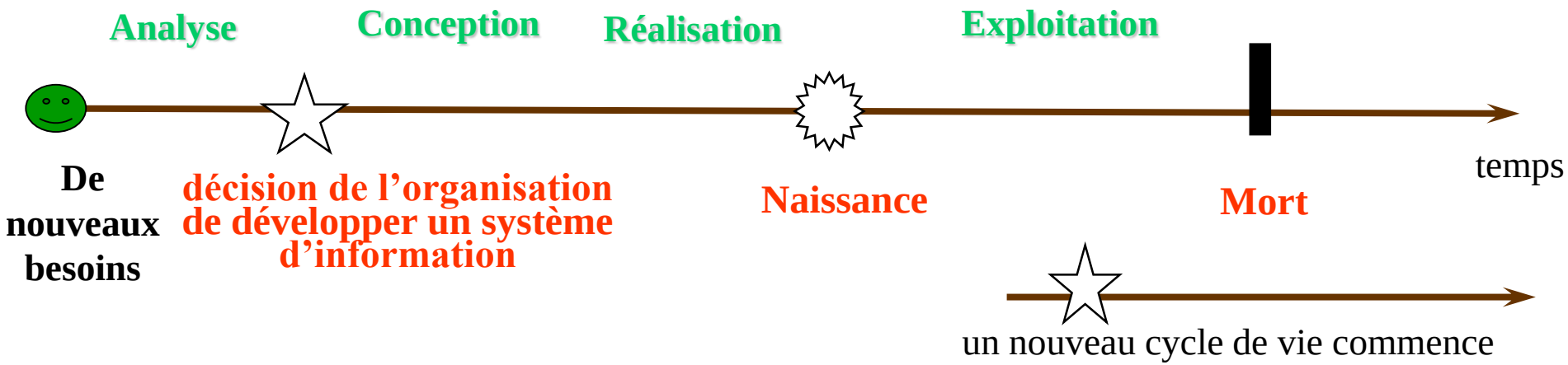
De l'ingénierie du logiciel … à l'ingénierie de système



Un système est un ensemble de composants matériel, logiciel et humain qui coopèrent de façon organisée dans le but d'atteindre un objectif commun

Contexte

Il a son propre cycle de vie au sein de l'organisation

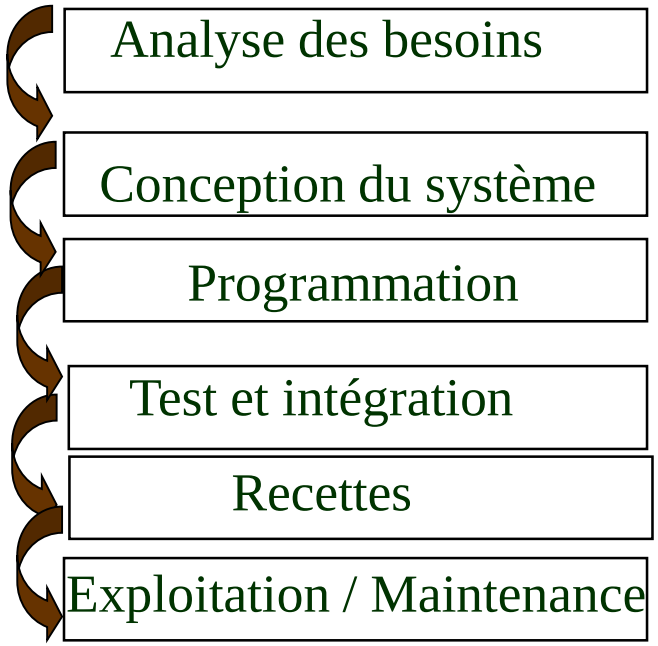


Les besoins correspondent à de **nouveaux objectifs**
Les besoins résultent **de problèmes, de disfonctionnements**
Les besoins peuvent venir **d'exemples, de démonstrations de produits**
Les besoins sont associés **à des innovations techniques**
Les besoins proviennent **de changements de règlementations, de lois…**

Contexte

Ingénierie de Système/ Ingénierie d'Exigence

L'organisation des étapes



- Chaque étape conduit à élaborer des résultats précis (documents, diagrammes, modèles…) ou produits du développement

- Chaque étape comporte un ensemble de phases, d'activités, de tâches

- Chaque étape utilise des concepts et des règles appropriées (formalisme)

Problèmes de L'I.L.

Fin des années 60, l'expression « Crise du logiciel » est apparue
évoqué en 1968 par McIlroy ('Mass Produced Software Components')

Classement du développent : succès, échec, mitigé
(les 1eres résultats édifiants : 5%, 19% , 76% en 1980)

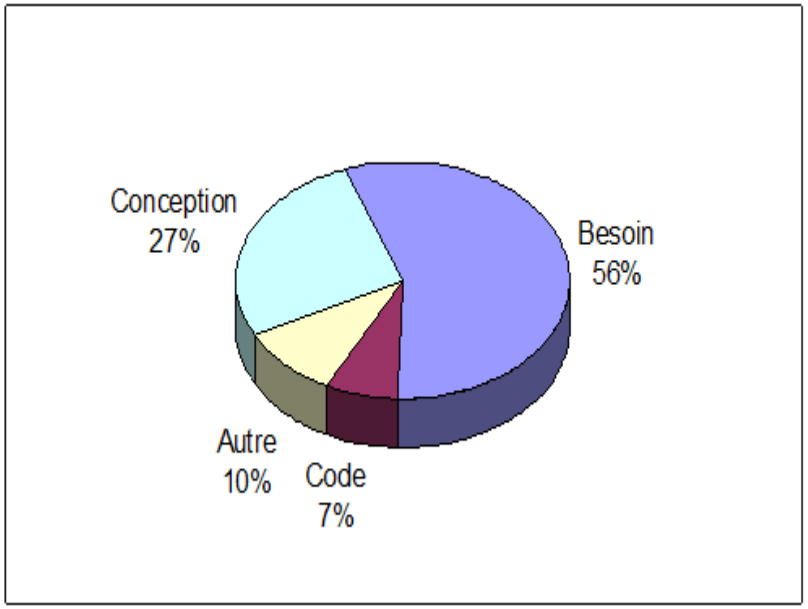
Enquête : études annuelles du standish group (Chaos Report)

Année	Succès	Mitigé	Echec
1995	16%	53%	31%
2000	28%	49%	23%
2006	35%	46%	19%
2010	37%	42%	21%
2013	39%	43%	19%

http://www.standishgroup.com.chaos.html

Complexité des applications ↗ et exigences des utilisateurs aussi

Problèmes de L'I.L.



Source Martin et Leffinel

D'autres études montrent que plus de la moitié des erreurs détectées pendant la vie d'un projet sont attribuables à un problème de définition des besoins

Coût de la correction

Les erreurs d'analyse : Dangereuses (Ariane 5, Boeing 737), tenaces

PHASE	COÛT RELATIF DE CORRECTION
Ingénierie des exigences	1
Conception	5
Réalisation	10
Vérification & Validation	20
Maintenance	200

Problèmes de L'I.E.

Succès ou échec d'un développement se situe lors des phases amont du développement du logiciel

- *Causes liées à la difficulté de comprendre les besoins des clients et des utilisateurs :*
  - *Pas assez d'informations des parties prenantes (13%)*
  - *Les exigences sont incomplètes (12%)*
  - *Faible implication des parties prenantes dans le projet (8%)*
  - *Les attentes sont irréalistes (6%)*
  - *Inévitable évolutivité des besoins (11%)*
  - *...etc*

Problèmes de L'I.E. : Pourquoi

• Activité conceptuelle cruciale, complexe, efforts importants de mise en œuvre

• Difficultés culturelles : plusieurs parties prenantes impliquant conflits de points de vue

- communication entre clients et développeurs
- Client pas toujours conscients de leurs besoins
- développeurs n'ont pas la connaissance du domaine d'application

L'I.E. : Qui, Quoi, Pourquoi

1) Partie prenante (Stakeholder) : est une personne physique ou morale ayant une influence ou un intérêt direct dans le projet

C'est une source principale d'exigences

- Personne qui interagit avec le système
- Entités juridiques
- Institutions
- Organisations

Exemple : dans APB

Ministère de l'éducation, Rectorat, lycée, Parents, élèves, Concepteur, développeur, etc.

L'I.E. : Qui, Quoi, Pourquoi

- **La question du Why**

  Pourquoi le système a t il besoin d'être développé?

- **La question WHAT**

  Ce que le système doit faire?

« Requirements definition must say

– why a system is needed, based on current or foreseen conditions,

– what system features will satisfy this context,

…….» (Ross77)

"Something that the product must do or a quality that the product must have" (Robertson99)

L'I.E. : buts, fonctions et contraintes des systèmes et de leur liens avec les spécifications du logiciel

Le système Post Bac

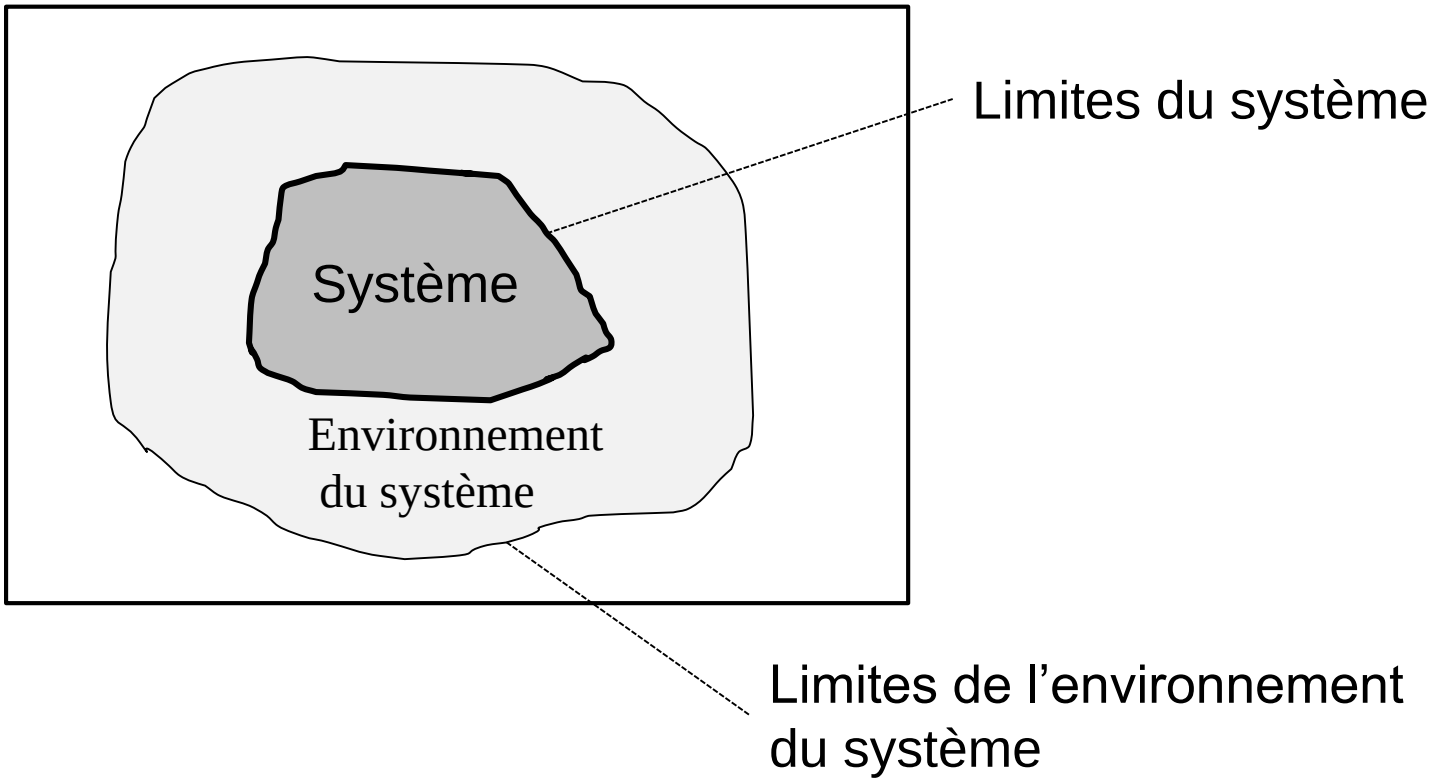
- **La question du WHY??**
  - Simplifier les procédures en regroupant sur un seul site l'ensemble des formations de l'enseignement supérieur?
  - Optimiser l'affectation des places offertes?
  - Aider à l'orientation ?

- **La question du WHAT??**
  - se renseigner sur les établissements (liens avec leur site)
  - émettre des vœux de poursuite d'études
  - classifier les vœux des lycéens
  - Les vœux ne doivent pas être connus des établissements
  - suivre son dossier
  - le système doit inclure un vérificateur d 'orthographe

L’ingénierie des exigences/Besoins

Définition

" L'ingénierie des exigences est l'activité qui transforme une idée floue en une spécification précise de besoins, souhaits, exigences exprimés par une communauté d'usagers et donc définit la relation existante entre un système et son environnement "



Les exigences

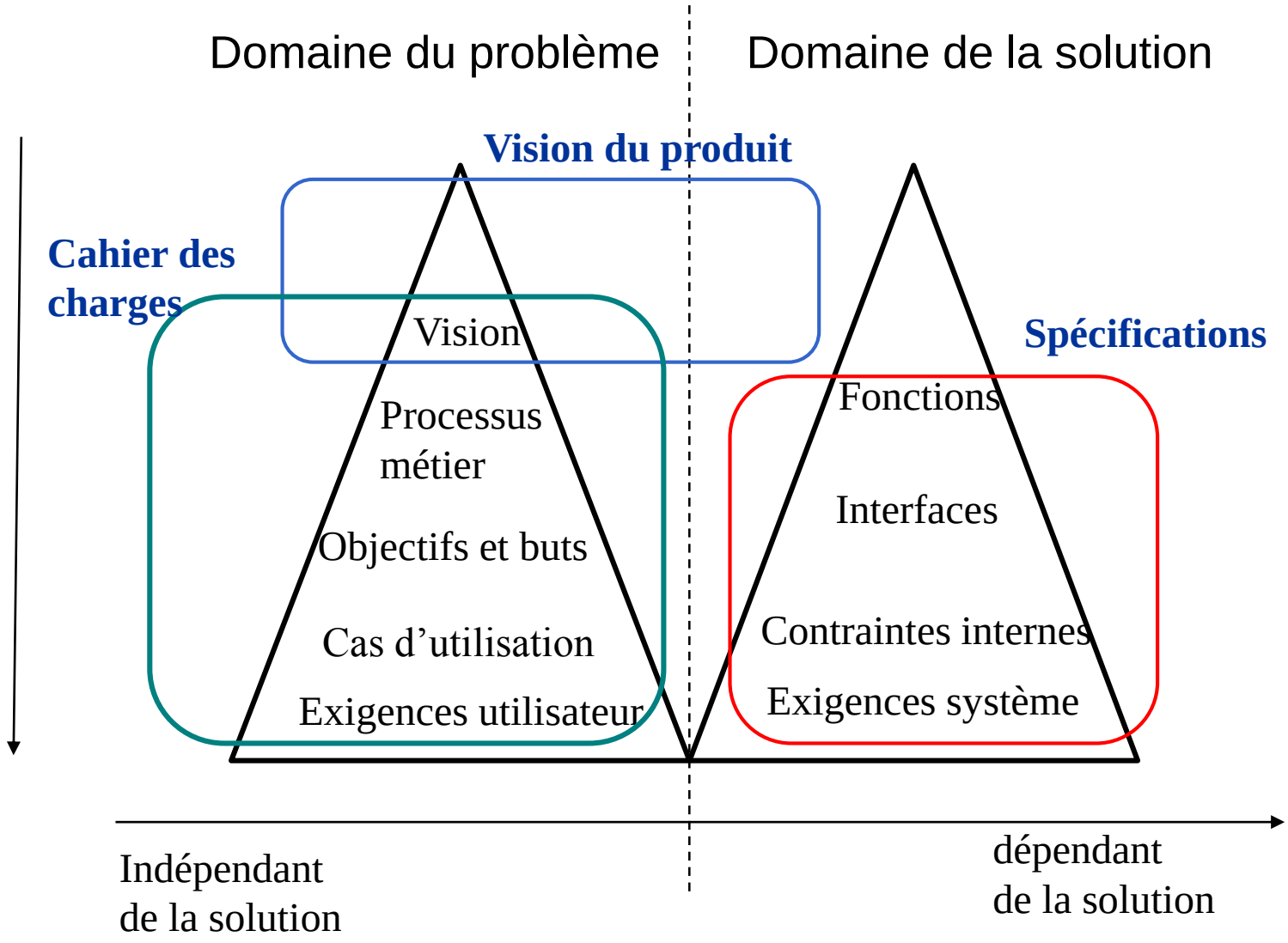
Plusieurs typologies

- Fonctionnelles (1) versus non fonctionnelles (2)
  (1) ce que le système doit faire
  (2) sous quelle contrainte le système doit le faire

- Produit (1) versus processus (2)
  (1) Fonctionnelles & non fonctionnelles
  (2) processus de développement

➢ Quoi exprimer? Comment l'exprimer ?
Problème /Solution, Choix , Détails techniques
Langage naturel, formel, uses cases, scénarios

Ingénierie des exigences



Expression des besoins = modèle abstrait du futur système

Ingénierie des exigences

Modélisation comme activité de base

Questions

- Comment modéliser les exigences (Why-What) ?
- Comment définir le modèle?
- Comment raisonner avec le modèle?

Quelques repères historiques

- L'orientation fonctionnelle (1970) :
  Méthodes Fonctionnelles : SADT, Jackson, etc
- L'orientation conceptuelle (1980)
  Méthodes conceptuelles : Merise
- L'orientation objet (1990) :
  Méthodes Objet : UML, SYSML

Limites des approches conventionnelles

Raisons :

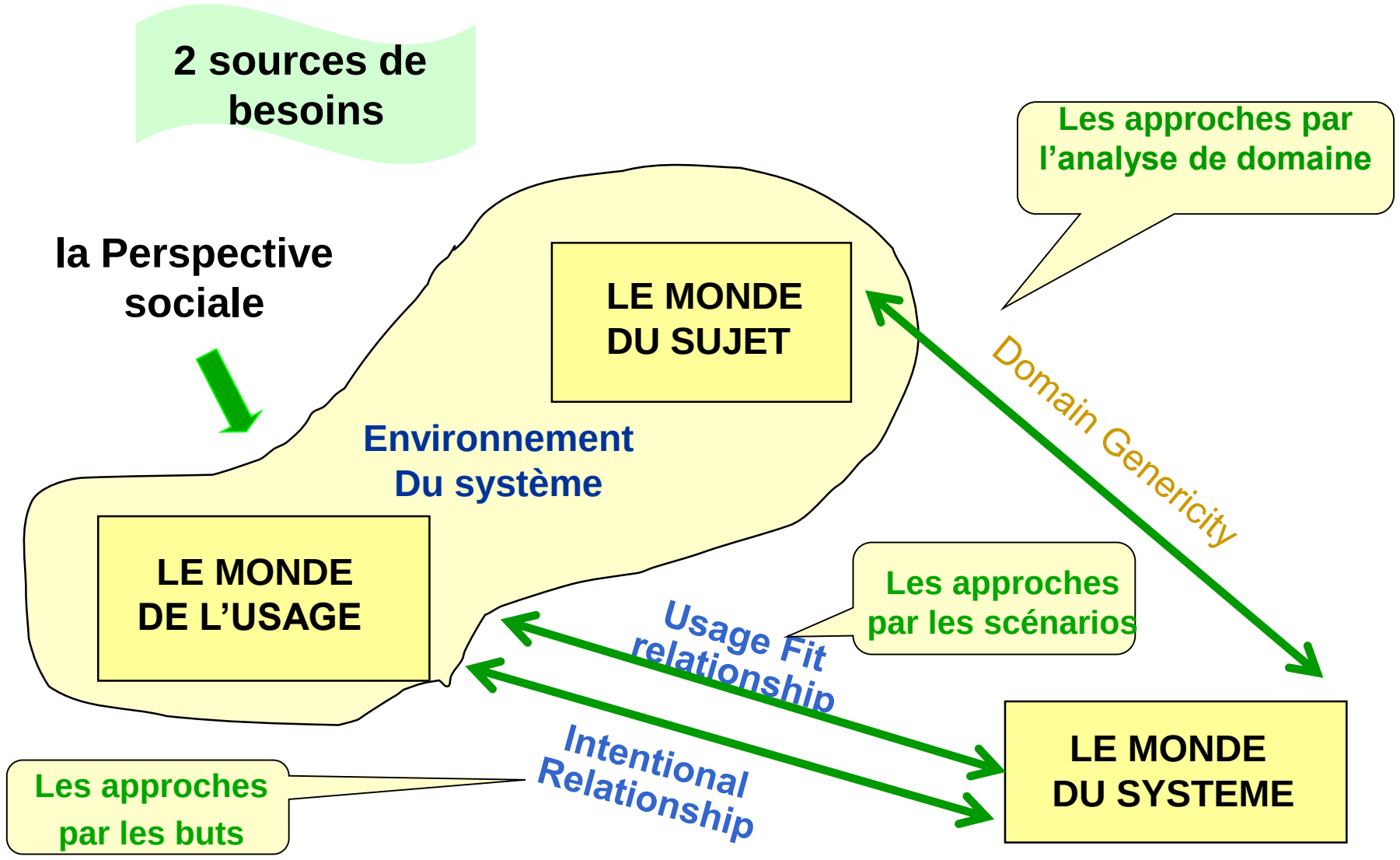
- Besoins supposés connus et stables
- Les utilisateurs ont juste besoin d'être questionnés
- Processus centré sur 'la solution technologique'
- Notations trop abstraites
- Maîtrise d'ouvrage trop informaticienne

Cadre de référence

Redéfinir l'ingénierie des besoins

- Elargir le champ de la modélisation : de E/R (Entité Relationship) à EM (Enterprise Modelling)
- Comprendre les problèmes
- Fixer le vocabulaire
- Repenser le processus d'I.E. :
- Les besoins se découvrent, s'élucident , se cultivent!
- Détecter les besoins manquants, ambigus
- Détecter les incohérences

Cadre de référence



D’autres approches d’I.E.

- Approche basée sur les buts
  - But : un objectif que le système doit atteindre par la coopération d’agents
    - Raffinements de buts : structures de graphes ET/OU
    - Notion de conflits entre buts
    - Lien d’opérationalisation
    - La méthode KAOS – l’outil Objectiver

- Approche basée sur les scénarios
  - Scénario: suite d’événements entre le système et son environnement
  - Lien Scénarios – But
  - Scénarios partiels (problème de couverture), procéduraux (risque de surspécification)

D’autres approches d’I.E.

- Travaux de Jackson, Parnas, Zave
  - Distinction entre propriétés de domaine et besoins (lois physiques de l’environnement ne sont pas de besoins)
  - Distinction entre besoins et spécifications
    - besoins en terme d’objets et notion de phénomène contrôlés par le logiciel
    - Spécifications en terme d’objets manipulés par le logiciel accessible par le développeur
  - Distinction entre besoins et hypothèses