

Analyse et modélisation

Séance 7: Étude préliminaire et capture des besoins fonctionnels

420-A56-GG

Fortement inspiré du livre « UML 2 en action »

Objectifs

- Faire une étude préliminaire

Comment modéliser avec UML ?

- UML est un langage qui permet de représenter des modèles, il ne définit pas le processus d'élaboration des modèles !
- Dans le cadre de la modélisation d'une application informatique, les auteurs d'UML préconisent d'utiliser une démarche :
 - itérative et incrémentale,
 - guidée par les besoins des utilisateurs du système,
 - centrée sur l'architecture logicielle.

Démarche itérative et incrémentale ?

- L'idée est simple : pour modéliser (comprendre et représenter) un système complexe, il vaut mieux s'y prendre en plusieurs fois, en affinant son analyse par étapes.
- Cette démarche devrait aussi s'appliquer au cycle de développement dans son ensemble, en favorisant le prototypage.
- Le but est de mieux maîtriser la part d'inconnu et d'incertitudes qui caractérisent les systèmes complexes

Une démarche pilotée par les utilisateurs ?

- Avec UML, les utilisateurs définissent ce que doit être le système.
 - le périmètre du système à modéliser est défini par les besoins des utilisateurs
- Les utilisateurs sont les clients du système
 - Le but du système à modéliser est de répondre aux besoins de ses utilisateurs.
- Les besoins des utilisateurs servent aussi de fil rouge, tout au long du cycle :
 - A chaque itération de la phase d'analyse, de la phase de conception et de réalisation, de la phase de test

Une démarche centrée sur l'architecture ?

- Une architecture adaptée est la clé du succès d'un développement.
- Elle décrit des choix stratégiques qui déterminent en grande partie les qualités du logiciel (adaptabilité, performances, fiabilité...).

En pratique alors ?

- UML n'est pas un processus... mais
 - il facilite une démarche d'analyse itérative et incrémentale, basée sur les niveaux d'abstraction.
- Les niveaux d'abstraction permettent de structurer les modèles.
 - Un micro-processus régit les itérations à niveau d'abstraction constant.
 - Un macro-processus régit le passage de niveau à niveau.
- La démarche incrémentale consiste à construire les modèles de spécification et de conception en plusieurs étapes.

Processus de développement en Y

- Un système d'information est généralement conditionné par la définition :
 - Un volet fonctionnel
 - Un volet technique
- Ces deux axes sont indépendants pour ce qui concerne les premières étapes du développement
- Ensuite, on les fusionne pour définir le système souhaité

Besoins fonctionnels

- C'est la définition des fonctions attendues et souhaitées pour le système
- Si nouveau système : spécifications du cahier des charges
 - Ex. : gestion de l'emploi du temps au Cégep
- Cas d'un système existant à modifier : spécifications des modifications ou nouvelles contraintes
 - Ex. : modification du calcul ; prise en compte de nouveaux paramètres dans une BD ; enrichissement des fonctionnalités, ...

Besoins techniques

- Nouveau système : choix des orientations techniques à privilégier (fonctionnement sur un intranet, application Client/Serveur en local...)
 - QUELQUE SOIT les applications
- Système existant : MàJ avec prise en compte des évolutions techniques
 - Ex. : accepter des commandes par le Web . Rien ne change fonctionnellement, mais l'architecture technique va évoluer
 - Ex.: ajout d'une fonctionnalité qui nécessite une synchronisation entre un fournisseur et son client.

Rappels UML

Étapes de l'axe fonctionnel

- 1. **Capture des besoins fonctionnels**, focalisés sur le métier des utilisateurs
- 2. **Analyse** (étude précise des besoins fonctionnels)

Étapes de l'axe technique

- 1. **Capture des besoins techniques** : choix, contraintes matérielles, contraintes d'intégration
- 2. **Conception générique** (indépendante des besoins fonctionnels): définition des composants nécessaires à l'architecture technique;

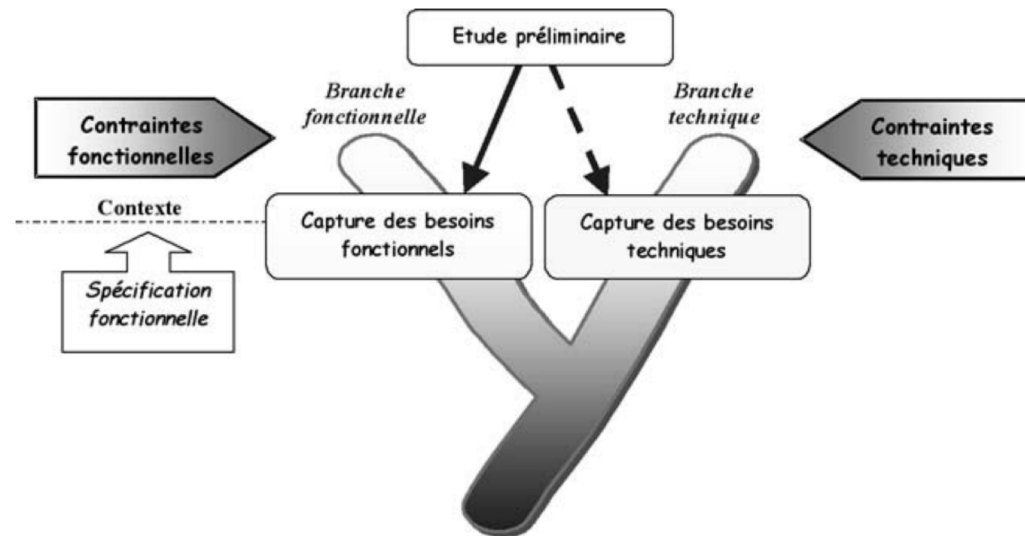
Étapes de l'axe final commun

- 3. **Conception préliminaire** : intégration du modèle d'analyse dans l'architecture technique
 - identification des composants du systèmes à développer
- 4. **Conception détaillée** : comment réaliser chaque composant
- 5. **Codage et tests**
- 6. **Recette** : valider les fonctions du système développé

Étude préliminaire

- Aussi appelée pré-étude, première étape de notre processus de développement
- Objectifs:
 - Premier repérage des besoins fonctionnels et opérationnels: textes et des diagrammes très simples (Système = boîte noire)
 - Modéliser le contexte après identification des acteurs et des interactions (messages) systèmes-acteurs.

- UML en action (P.46)



Étude préliminaire

Résultats de l'étude préliminaire

- un Cahier des Charges préliminaire
- la liste des acteurs
- une description des messages entre les acteurs et le système
- une modélisation du contexte dynamique, qui montre quels messages sont échangés entre les acteurs et le système
- Éventuellement : contexte statique, pour les cardinalités

Étude de cas

<https://doc.lagout.org/programmation/Databases/SQL/UML.pdf> (P.59)

- VExpress est une société dont l'activité principale est la messagerie. Cette activité consiste en l'enlèvement, le transport et la livraison de colis.
- VExpress possède 800 véhicules répartis sur 70 agences. La société traite un volume moyen de 40 000 colis par jour (enlèvements et livraisons) et compte 3 000 employés.
- VExpress souhaite se doter d'un système informatique performant afin de :
 - maîtriser au plus près l'acheminement des colis par la connaissance de leur localisation et de leur état,
 - suivre la réalisation des commandes, ainsi que la gestion comptable des factures et des règlements,
 - offrir aux clients la possibilité de suivre l'acheminement de leurs colis via une connexion Internet.
- La durée de vie du nouveau système, appelé SIVEx (Système d'Information de VExpress), est estimée à 5 ans.

Étude de cas

- Société de messagerie SIVExpress : enlèvement, transport et livraison de colis
- 70 agences, 3000 employés et 40 000 colis traités par jour
- Souhaite s'informatiser : un syst. qui permette de (besoins fonctionnels) :
 - localiser et connaître l'état des colis en TR
 - suivre les commandes + gestion comptable
 - autoriser les clients à suivre l'acheminement de leur commande via Internet
- Besoins techniques : UML / archi. 3-tiers / Internet / JAVA /SGBDR (cf CdesCh)

Étude de cas

- L'identification des acteurs
- L'identification des messages
- La réalisation du diagramme de contexte

Identifier les acteurs

Qu'est ce qu'un acteur?

- Représente l'abstraction d'un rôle joué par des entités externes qui interagissent directement avec le système étudié.
- un acteur peut être un humain ou un constituant informatique, un dispositif matériel ou un autre système : néanmoins c'est une entité externe
- Identification des acteurs : processus itératif

Identifier les acteurs

- Utilisateurs humains : identifier tous les profils possibles
 - opérateur de maintenance, administrateur...
- Autres systèmes connexes qui interagissent avec le système
 - vérifier qu'ils agissent bien directement avec le système, et pas par le biais d'un acteur déjà répertorié !
- Vérifier que les acteurs sont bien extérieurs au système, que ce ne sont pas des composants.
- Vérifier aussi qu'il a une autonomie de décision, ça ne doit pas être un matériel passif

Identifier les acteurs

Liste des acteurs de SIVEx

- Réceptionniste
 - Saisie et annule les cdes client
- Client
 - Consulte ses cdes via internet, reçoit confirmation cde par courriel ou fax
- Progiciel de comptabilité
- Comptable
 - Fait le point sur les cdes, établit factures/avoirs; recouvrements
- Répartiteur
 - Crée les différentes missions en fonction cdes et ressources;

Identifier les acteurs

Liste des acteurs de SIVEx

- Chauffeur
 - Assure les missions; réceptionne/livre les colis; informe de ses arrêts le système
- Opérateur de quai
 - Identifie et pèse les colis provenant d'un enlèvement; pointe le passage des colis en départ et arrivée d'agence; inventaires de quai.
- Responsable logistique
 - Définit le réseau des agences; stratégie de transport
- Administrateur système
 - Gère les profils utilisateurs du système, et mots de passe; archivages

Identifier les messages

Qu'est qu'un message?

- message = communication entre acteur et système
 - le message transporte de l'information
 - le message est envoyé avec l'intention de déclencher une activité chez le récepteur
 - la réception d'un message est considérée comme un événement
- Pour chaque acteur, se demander quels sont les messages envoyés par les acteurs qui déclenchent un comportement attendu, dans le cadre de leur activité
- Pour le système, se demander quels sont les messages envoyés vers chaque acteur

Identifier les messages

- Pas de messages entre acteurs !
- Construire le diagramme de contexte dynamique, qui ordonne et classe les messages.
 - Système au centre, comme une boîte noire
 - Acteurs autour Mentionner les messages et le sens de la communication : de l'acteur vers le système ou l'inverse (cf SIVEX)
- Liste des messages de l'étude de cas ?

Identifier les messages

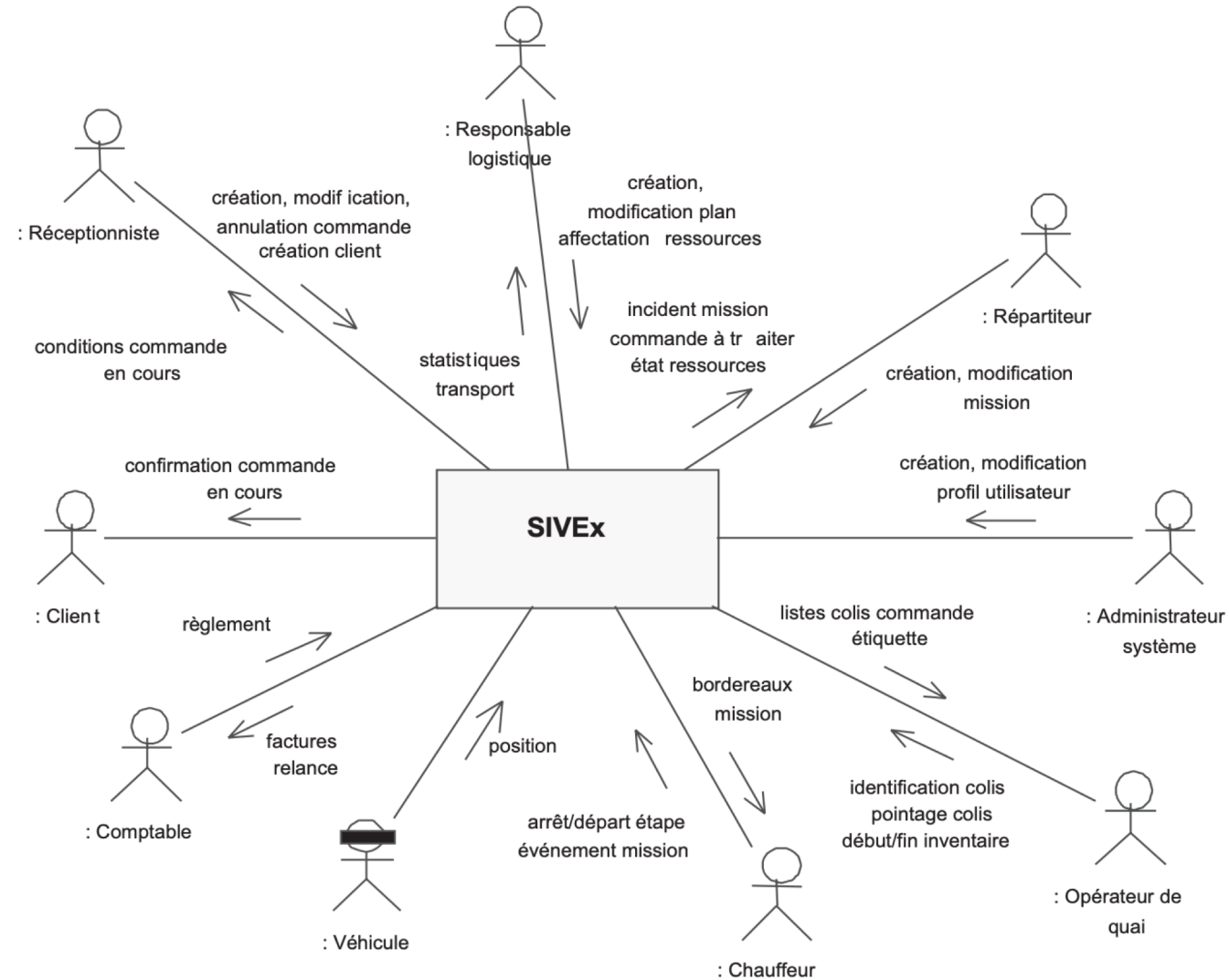
Liste des messages de l'étude de cas ?

- Le système SIVEx émet (entre autres) :
 - les statistiques de transport pour le responsable logistique,
 - les confirmations de commande pour le client,
 - les incidents de mission pour le répartiteur.
- Le système SIVEx reçoit (entre autres) :
 - les créations, modifications, ou annulations de commande du réceptionniste,
 - les règlements de facture du comptable,
 - les créations de mission du répartiteur,
 - les informations de suivi de mission du chauffeur,
 - les positions des véhicules,
 - l'identification et le pointage des colis de l'opérateur de quai.

Modéliser le contexte

- Tous les messages (système - acteurs) identifiés peuvent être représentés de façon synthétique sur un diagramme, que l'on peut qualifier de diagramme de contexte dynamique.

Modéliser le contexte



Résumé de l'étude préliminaire

- Établissement d'un recueil initial des besoins fonctionnels et opérationnels
- Modéliser le contexte du système, considéré comme une boîte noire:
 - identifier les entités externes du systèmes qui interagissent avec lui: les acteurs
 - répertorier les interactions (émission/réception des messages) entre ces acteurs et le système
 - représenter l'ensemble des interactions sur un modèle de contexte dynamique, éventuellement sur un modèle de contexte statique, ou décomposé pour faire apparaître les principaux sous système fonctionnels.

Résumé de l'étude préliminaire

