



UML

Dr Khalifa SYLLA

Séquence 1: INTRODUCTION GENERALE

INTRODUCTION GENERALE

1.1. Les systèmes d'information (SI)

Définition

Exemples de SI :

- Logiciel des gestions d'étudiants
- Site web de vente de matériels informatiques
- Application de gestion d'un restaurant

« Un système d'information est un ensemble des ressources (matériel, logiciel, personnel, données, procédures,...) permettant d'acquérir, traiter, stocker l'information (sous forme de données, textes, sons, images, ...) dans et entre les organisations. » Robert Reix

Matériel	→	machines, supports
Logiciel	→	programmes, méthodes
Personnel	→	utilisateurs
Données	→	connaissances, modèles
Procédures	→	méthodes

Mais qui sont ces organisations qui manipulent les systèmes d'information ? Plus simplement, il s'agit d'un ensemble de personnes. Dès qu'une activité dépasse une seule personne, on parle d'organisation. Les différents acteurs de l'organisation partagent les mêmes objectifs et ont chacun un rôle bien défini.

Si des individus se constituent en organisation, c'est parce qu'ils ont un objectif de **résultat**.

Exemples de résultat :

- Produire un logiciel
- Mettre en ligne un commerce

Pour atteindre cet objectif, le travail (le projet) est divisé en plusieurs tâches de manière cohérente et chaque acteur de l'organisation possède une tâche bien déterminée.

Les principales fonctions d'un SI sont :

- Collecter des informations
- Traiter et transmettre des informations
- Mémoriser des informations

Quel est le lien avec l'informatique ? L'informatique facilite la gestion d'un SI. L'informatisation nécessite la mise en place de supports matériels et logiciels de stockage des données.

1.2. Pourquoi modéliser ?

Qu'est ce qu'un modèle ?

C'est une représentation **abstraite** (ou tout simplement une abstraction de la réalité) ou une vue subjective mais pertinente de la réalité.

L'**abstraction** est un des piliers de l'**approche objet**. Il s'agit d'un processus qui consiste à identifier les **caractéristiques intéressantes** d'une entité, en vue d'une utilisation précise.

L'abstraction désigne aussi le **résultat** de ce processus, c'est à dire l'ensemble des caractéristiques essentielles d'une entité, retenues par un observateur.

A quoi sert un modèle ?

L'abstraction dans un modèle permet de :

- Faciliter la compréhension du système étudié par la réduction de sa complexité
- Simuler le système étudié à partir de la représentation de ce système et de la reproduction de ses comportements.

Ainsi un modèle réduit (décompose) la réalité, dans le but de disposer d'éléments de travail exploitables par des moyens mathématiques ou informatiques, grâce à l'abstraction qui permet de cerner les caractéristiques fondamentales du système étudié, suivant l'optique du modélisateur.

La modélisation

Pour construire un bâtiment, l'architecte s'appuie sur un plan conçu selon les spécificités de la parcelle et les besoins du client. De la même manière, pour concevoir un système d'information, l'informaticien s'appuie sur un **modèle**, on parle de **modélisation**.

Définition : la modélisation, c'est tout processus de mise en oeuvre d'un modèle d'un système donné.

La modélisation permet de comprendre le SI en développement. Les objectifs sont :

- Visualiser un système tel qu'il est ou tel que nous voudrions qu'il soit
- Préciser la structure ou le comportement d'un système
- Fournir un canevas qui guide la construction d'un système
- Documenter les décisions prises

Le développement de logiciels suit des règles encadrant la conception et permettant le travail en groupe (organisation). La modélisation n'est qu'une étape du travail de conception.

La **modélisation fonctionnelle** est centrée sur l'identification :

- Des données (attributs, variables) du système;
- Des fonctions (service ou comportements) du système. Elles manipulent les données du système;
- Un canevas qui guide la construction d'un système ;
- L'interdépendance des fonctions.

Avantages de la modélisation

L'approche fonctionnelle reste la plus intuitive pour nos esprits cartésiens. En effet:

- Il s'agit de répondre à la question de savoir: Que fait le système ? Il suffit de déterminer les fonctionnalités du système sans s'occuper de leur(s) origine(s) dans le système.
- Question beaucoup plus simple que celle de savoir: Qu'est-ce qui, dans le système, fait quoi ? Il faut déterminer les fonctionnalités du système, ainsi que leur(s) origine(s) dans le système.

Inconvénients de la modélisation

On peut noter entre autres:

- La séparation des données et des traitements qui sécurise moins les données.
- L'approche fonctionnelle n'est pas adaptée au développement d'applications qui évoluent sans cesse et dont la complexité croît continuellement.
- L'absence d'Environnement de Développement Intégré (EDI) efficace à la hauteur d'UML, quand à la génération automatique de code source.

1.3. La modélisation orientée objet

Pourquoi l'approche objet ? Quelle est la raison qui rend l'approche objet tellement attractive?

Cette question, les adeptes de l'objet répondent invariablement que les avantages de l'approche objet sont:

- la stabilité de la modélisation pour rapport aux entités du monde réel
- la construction itérative facilitée par le couplage faible entre composants
- la possibilité de réutiliser des éléments d'un développement à un autre
- la simplicité du modèle qui fait appel qu'à cinq concepts fondateurs (les objets, les messages, les classes, l'héritage et le polymorphisme) pour exprimer de manière uniforme l'analyse
- la conception et la réalisation d'une approche informatique

L'approche objet a été inventée pour faciliter l'évolution des applications. Elle considère une application comme une société d'objets qui coopèrent.

L'achèvement d'une tâche par une application repose sur la communication entre tous ou certains de ses objets. L'unité de communication entre objets étant le message.

La MOO est basée essentiellement sur trois concepts que sont:

- L'encapsulation
- L'héritage
- Le polymorphisme

Ne pas confondre MOO et LOO (langage orientée objet). Un langage orientée objet (LOO) est un langage de programmation permettant d'implémenter un MOO.

C'est tout langage qui supporte les mécanismes d'héritage, d'encapsulation et de polymorphisme.

Chaque LOO a ses spécificités. ^[1]_{SEP} Il existe à présent plusieurs LOO parmi lesquels: C++, Java, ^[1]_{SEP} C#, etc.

1.4. UML

UML (Unified Modeling Language), traduit en français « langage de modélisation objet unifié » est né en 1995 de la fusion des trois méthodes qui ont le plus influencé la modélisation objet : OMT, Booch et OOSE.

Il convient à tous les langages objets (C++ (Héritage multiple, Template), Java (Interface), C#, . . .)

UML est normalisé par l'OMG (Object Management Group), en fin 1997. OMG regroupe plus de 80 des plus grandes Entreprises du monde (HP, IBM, SONY, ERICSON, MICROSOFT,...)

UML est un langage de modélisation orientée objet : toutes les entités modélisées sont des objets ou se rapporte à des objets. C'est n'est pas une méthode, cependant il a été adopté par toutes les méthodes orientées objet.

Avantages d'UML

UML présente plusieurs avantages :

- Transcender les contraintes d'implémentation liées aux langages et aux systèmes;
- Représenter des concepts abstraits (graphiquement);
- Limiter les ambiguïtés (parler un langage commun, au vocabulaire ^[1]précis, indépendant des langages orientés objet);
- Faciliter l'analyse (simplifier la comparaison et l'évaluation de solutions);
- Ne pas effectuer une analyse fonctionnelle et se contenter d'une implémentation objet, mais penser objet dès le départ,
- Définir les vues qui permettent de décrire tous les aspects d'un système avec des concepts objets.
- Sécuriser d'avantage les données (par l'encapsulation)
- Réutiliser efficacement l'existant (par l'héritage et le polymorphisme)
- Une panoplie d'outils et de langages performants pour le développement.
- Bases de données orientées objet (O2, ObjectStore, Versant...)

Inconvénients d'UML

Cependant, UML présente quelques inconvénients. En effet, comme le langage est précis, il nécessite un effort d'apprentissage. De plus, pour un projet donné, il est souvent difficile de naviguer dans l'abstrait, donc difficulté d'identifier les objets.