Chapitre 1 Introduction

Depuis la parution, il y a maintenant neuf ans, de la première édition du livre de P.A. Muller [Muller 03] chez le même éditeur, de nombreux autres auteurs ont apporté leur contribution à la diffusion du langage UML, avec des points de vue souvent complémentaires, tels que [Kettani 01] et [Soutou 02].

Les « pères » d'UML eux-mêmes ont décrit dans le détail les concepts, les notations et la sémantique du langage dans des ouvrages de référence [UML-UG 05] et [UML-RM 04].

Alors pourquoi un autre livre sur UML ?

Les professionnels de l'informatique cherchent régulièrement des exemples concrets à partir desquels ils pourront élaborer leurs propres projets. C'est à cette préoccupation très pragmatique que souhaite répondre ce livre qui, pour illustrer UML, présente une étude de cas réaliste couvrant toutes les étapes du processus de développement.

Cette approche est retenue depuis longtemps à Valtech[1], dans le cadre de l'offre de formation aux entreprises, en particulier pour les deux cours phares que sont :

- le cours « Modéliser les besoins et analyser avec UML », qui est consacré à la présentation générale d'UML et de son utilisation pour l'expression des besoins et la spécification détaillée ;
- le cours « Analyse et conception avec UML », qui offre un panorama complet de l'utilisation d'UML dans une démarche de développement de type Processus Unifié.

1. L'activité de formation a été filialisée fin 2002 pour donner naissance à la société Valtech Training, dont fait partie Pascal Roques.

Ce livre est construit à partir de ce matériel pédagogique éprouvé et de notre expérience à l'enseigner. Cependant, notre métier de consultant ne consiste pas à répéter les mêmes formations. Dans le cadre de missions d'assistance et de conseil, notre offre se prolonge auprès des équipes de projets. C'est donc cette expérience du terrain ainsi que notre exigence d'aboutir à des réalisations concrètes que nous avons essayé de retranscrire dans ce livre.

En conséquence, vous ne trouverez pas ici de description formelle exhaustive du langage UML, ni de réflexions théoriques alambiquées sur un quelconque aspect de sa dernière version. Mais ce livre vous montrera l'application pratique d'UML à partir d'un développement complet. C'est ainsi que vous apprendrez :

- à utiliser UML pour capturer les besoins des utilisateurs ;
- à analyser ces besoins avec UML ;
- puis à concevoir avec UML et les design patterns en vue d'un développement Java.

En dernier point, nous vous livrons un processus de développement qui, adapté au développement des systèmes Client/Serveur, s'inscrit dans la lignée du « Unified Process ».

Loin de nous prétendre exhaustifs, ou détenteurs d'une vérité absolue, notre unique ambition consiste à présenter notre approche du développement logiciel avec UML, afin que vous puissiez en bénéficier à votre tour.

Nota : cette quatrième édition incorpore des nouveautés de la version 2 d'UML[1], en particulier au niveau des diagrammes de séquence.

Prérequis

Ce livre est en quelque sorte le pendant pratique de la théorie UML ; il a été essentiellement pensé comme le complément utile d'un ouvrage de référence tel que [Fowler 04] .

Pour pouvoir en tirer pleinement profit, nous vous conseillons donc de connaître les rudiments de l'approche objet. Les termes classe, instance, encapsulation, héritage et polymorphisme doivent vous être familiers. Dans la plupart des cas, la pratique d'un langage de programmation objet suffit pour acquérir ces concepts.

1. Le document le plus récent utilisable lors de cette quatrième édition a été le « 06-04-02.pdf » téléchargeable sur le site de L'OMG (www.uml.org). Il s'agit de « UML 2.1 Superstructure Specification ».

En second lieu, il convient d'avoir compris ce qu'est une méthode de développement logiciel qui intègre notamment un support de modélisation. La connaissance d'une méthode de type Unified Process [Jacobson 99], de Merise, ou de tout autre méthode orientée modélisation, vous permettra de mieux situer la démarche que nous mettons en pratique dans cet ouvrage.

Ce livre ne fait que rappeler les rudiments d'UML, dans la mesure où il existe aujourd'hui suffisamment d'ouvrages sur le sujet. Néanmoins, l'annexe B présente rapidement les diagrammes d'UML 2.

Structure de l'ouvrage

Cet ouvrage s'articule autour de l'étude de cas SIVEx, que vous découvrirez en temps utile (chapitre 3). Comme c'est avant tout un guide pratique d'utilisation d'UML dans diverses situations, il ne manque pas une occasion d'utiliser UML. D'ailleurs, pour présenter la structure des chapitres, nous avons utilisé un diagramme d'activité !

Notre processus s'appelle le « 2 Track Unified Process » ou processus en Y ; il est sous-jacent à la structure du livre, comme le montre la figure 1-1. Ce processus est décrit plus en détail au chapitre 2, *Processus et architecture*.

La première partie du livre fait office d'entrée en matière.

- Le chapitre 1, *Introduction*, correspond à la présentation de l'ouvrage.
- Le chapitre 2, *Processus et architecture* vous livre notre vision du processus et de l'architecture, ainsi que l'importance que nous leur accordons. C'est notamment ici que vous trouverez toutes les explications sur le processus en Y.

La seconde partie concerne la modélisation des besoins.

- Le chapitre 3, *Étude préliminaire*, présente le sujet de l'étude de cas SIVEx, et commence la modélisation de son contexte.
- Le chapitre 4, *Capture des besoins fonctionnels*, explique comment identifier les besoins exprimés selon le métier des utilisateurs, les reformuler, les structurer et les documenter avec UML. Il s'appuie pour une large part sur la technique des cas d'utilisation.
- Le chapitre 5, *Capture des besoins techniques*, indique comment identifier les exigences qui ont trait à l'exploitation d'un système logiciel, les reformuler, les structurer et les documenter avec UML. Il s'appuie également sur les cas d'utilisation et introduit la notion importante de découpage en couches.

La troisième partie concerne l'analyse objet.

- Le chapitre 6, *Découpage en catégories*, montre comment organiser la structure des concepts d'un système pour établir une cartographie judicieuse des classes issues des besoins fonctionnels.
- Le chapitre 7, *Développement du modèle statique*, décrit et illustre le travail d'analyse détaillée de la structure des classes.
- Le chapitre 8, *Développement du modèle dynamique*, est consacré au travail d'analyse détaillée du comportement des classes.

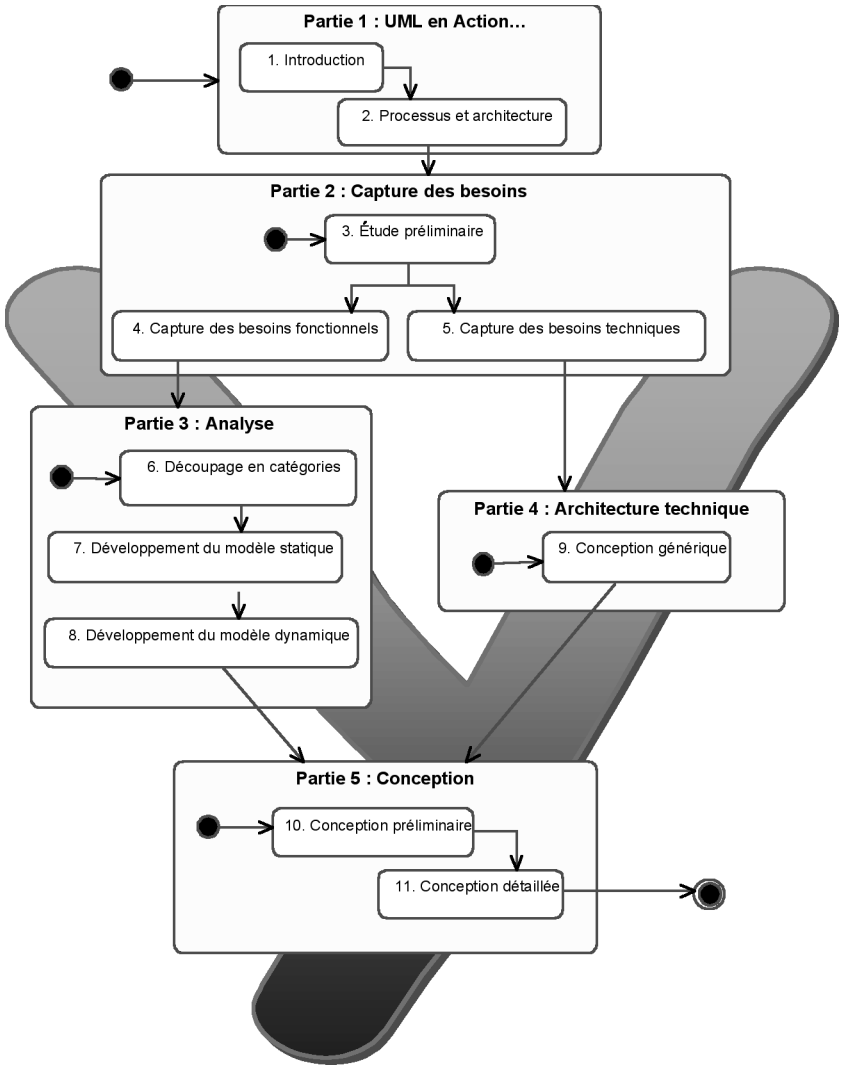


Figure 1-1 : Le processus en Y sous-jacent à la structure du livre

La quatrième partie concerne la conception de l'architecture technique.

- Le chapitre 9, *Conception générique*, explique comment, à partir de l'analyse des besoins techniques, il est possible de commencer à concevoir le système, indépendamment de son contenu fonctionnel.

La cinquième partie concerne la conception objet.

- Le chapitre 10, *Conception préliminaire*, explique comment organiser un modèle de conception au vu des regroupements d'analyse et des couches logicielles d'architecture. Il illustre notamment l'identification des composants métier et techniques d'un système logiciel.
- Le chapitre 11, *Conception détaillée*, illustre la modélisation de solutions Java en appliquant différents design patterns, suivant les couches que l'on désire réaliser.

Ce livre comporte également quatre annexes et un index :

- l'annexe A qui constitue une bibliographie de référence ;
- l'annexe B qui résume la notation des diagrammes d'UML 2 ;
- l'annexe C qui dresse une synthèse des mots-clés UML et des stéréotypes utilisés dans cet ouvrage ;
- l'annexe D qui récapitule les conseils et les pièges à éviter.

Pour que vous puissiez vous repérer aisément dans l'ouvrage, nous avons attribué des titres aux paragraphes traitant du processus, à savoir :

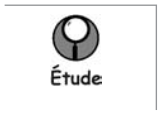
- *Objectif*, qui définit l'exposé du chapitre ;
- *Quand intervient…*, qui vous rappelle la position du chapitre dans le processus en Y et vous restitue le contexte d'intervention ;
- *Éléments mis en jeu*, qui établit la liste des mots-clés, techniques UML, concepts d'architecture et processus du chapitre ;
- *Phases de réalisation*, qui récapitule le détail du processus utilisé dans le chapitre ;

Outre le texte principal, chaque chapitre est ponctué d'insertions facilement repérables grâce aux icônes suivantes :

RAPPEL OU ÉNONCÉ D'UNE DÉFINITION

NOUS VOUS CONSEILLONS DE…

NOUS VOUS DÉCONSEILLONS FORTEMENT DE…



APPROFONDISSEMENT D'UN POINT PARTICULIER

Comment lire UML en Action…

La grille de lecture est en partie dédiée à nos homologues consultants qui ont rarement le temps de lire un ouvrage jusque dans ses moindres détails. Aussi avons-nous établi la spécification fonctionnelle de notre ouvrage… en UML.

Les lecteurs que nous avons identifiés sont représentés dans la nomenclature de la figure suivante.

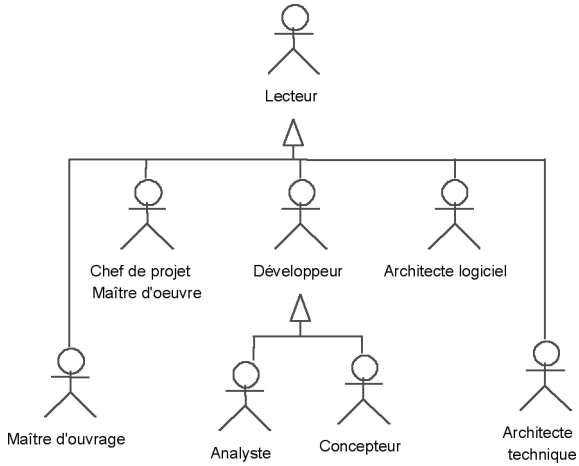


Figure 1-2 : Typologie des lecteurs de UML en action…

Le lecteur « Maître d'ouvrage » est client du système à développer. Il participe à la spécification en représentant le métier des utilisateurs ; il est donc impliqué dans l'analyse objet. Il s'intéresse également au processus de développement afin de s'assurer que les dispositions nécessaires à la réussite du projet ont été prises.

Le lecteur « Chef de projet » appartient à la maîtrise d'œuvre. Il s'intéresse à l'aspect gestion du développement et doit inciter à l'utilisation des meilleures pratiques de développement. C'est pourquoi il est directement concerné par toutes les techniques de modélisation objet présentées dans cet ouvrage.

Le lecteur « Développeur » représente un membre de l'équipe de développement. Il est soit analyste, soit concepteur, soit successivement les deux. En fonction de son rôle, il est intéressé par les techniques de modélisation objet soit pour l'analyse, soit pour la conception.

Le lecteur « Architecte technique » met en œuvre les outils, les langages et les plates-formes nécessaires à l'équipe de développement. Il est concerné par la modélisation d'une conception objet, avec laquelle il va construire les composants techniques requis par les développeurs.

Le lecteur « Architecte logiciel » est chargé de structurer et d'organiser le système informatique. Son intérêt se porte par conséquent sur les modèles d'analyse qu'il doit intégrer au niveau conceptuel et sur les modèles de conception au niveau logiciel. Il trouvera également les techniques permettant de concevoir les briques, réutilisables ou non, avec lesquelles est façonné un système informatique résilient et évolutif.

Des cas d'utilisation structurent le contexte dans lequel l'ouvrage peut être utilisé par les différents lecteurs. C'est encore une bonne occasion d'utiliser UML.

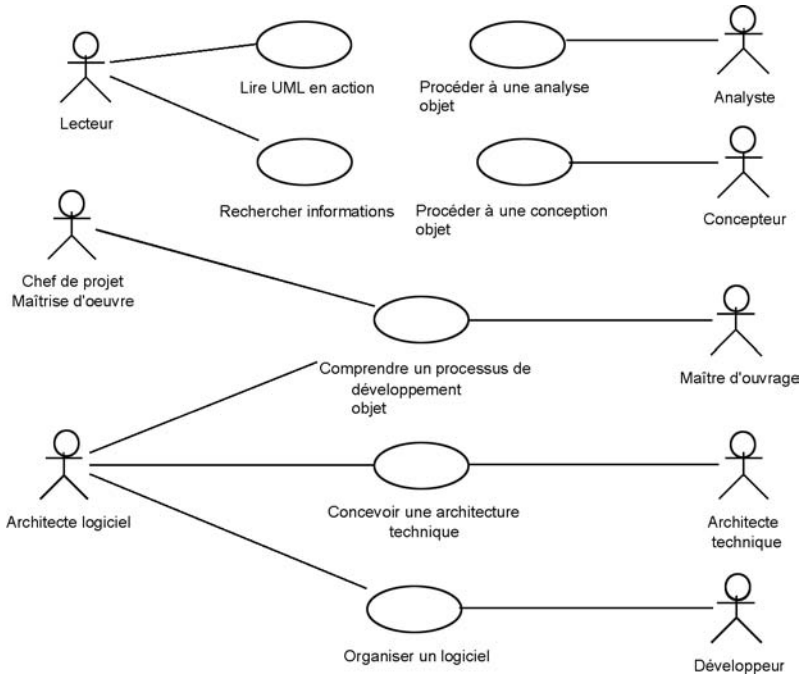


Figure 1-3 : Les cas d'utilisation de UML en Action…

Cette courte analyse des besoins concernant *UML en Action…* aboutit à la grille de lecture ci-dessous.

Qui	Fait quoi	Comment
Le lecteur	lit UML en Action…	de manière séquentielle.
Le lecteur	recherche les informations	grâce à nos différents outils de recherche : l'index, la cartographie du processus, les titres « Éléments mis en jeu » et « Phases de réalisation » de chaque chapitre et enfin la bibliographie.
L'analyste	procède à une analyse objet	doit lire dans l'ordre le chapitre 3, Étude préliminaire, pour comprendre la teneur fonctionnelle de SIVEx, le chapitre 4, Capture des besoins fonctionnels, puis la troisième partie concernant l'analyse objet.
Le concepteur	procède à une conception objet	doit lire le chapitre 9, Conception générique, pour comprendre comment concevoir des composants techniques, puis la cinquième partie dédiée à la conception objet.
Le maître d'ouvrage, le chef de projet, l'architecte	comprennent un processus objet	doivent se référer au chapitre 2, Processus et architecture, puis aux titres «Quand intervient…» et «Phases de réalisation» de chaque chapitre.
Les architectes technique et logiciel	conçoivent une architecture technique	doivent consulter le chapitre 5, Capture des besoins techniques, pour voir comment en acquérir les spécifications, puis la quatrième partie consacrée à l'architecture technique.
L'architecte logiciel	organise un système logiciel	doit lire le chapitre 2, Processus et architecture, introduisant les termes d'architecture, le chapitre 6, Découpage en catégories, pour comprendre l'organisation conceptuelle, le chapitre 9, Conception générique, qui concerne l'organisation technique et logicielle, et le chapitre 10, Conception préliminaire, qui traite de l'organisation en composants.

Remerciements

UML en Action… n'aurait pas pu voir le jour sans l'accord de la société **Val***tech*, dont nous avons exploité l'étude de cas provenant d'une ancienne version de la formation « Modéliser les besoins et analyser avec UML ». C'est donc un premier remerciement à Jean-Yves Hardy et Olivier Cavrel.

Bien que fortement imprégné de nos deux expériences, cet ouvrage est aussi le résultat de discussions, d'échanges, et de documents de capitalisation provenant de l'ensemble des consultants de **Val***tech* à travers le monde. C'est ainsi que vous bénéficiez de l'expérience accumulée sur plus de mille projets dans les secteurs les plus variés. Nous tenons donc à remercier ceux qui ont bien voulu prendre un peu de temps pour nous faire part de leurs remarques et suggestions, en particulier Françoise Caron, Pierre Chouvalidzé, Thibault Cuvillier, Philippe Dubosq, Michel Ezran, Thomas Gil, Jean-Luc Guigné, Patrick Le Go, Yann Le Tanou, Gaël Renault, José Roméro, Christophe

Addinquy, tous de **Val***tech*, et Frédérique Vallée de MATHIX SA (pardon d'avance à ceux que nous avons oubliés !).

Le contenu technique est une chose, mais le travail de présentation qui rend un livre facile à exploiter et répondant bien aux attentes des lecteurs, nous le devons à l'équipe éditoriale d'Eyrolles.

Enfin, un grand merci à nos familles, avec une mention particulière à Pascale et Fabienne, qui nous ont soutenus et encouragés tout au long de cette petite aventure.