



THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par *l'Institut National Polytechnique de Toulouse*

Discipline ou spécialité : *Systèmes Industriels*

Présentée et soutenue par *Romain DEGUIL*

Le *24 novembre 2008*

Titre : *Mapping entre un référentiel d'exigences et un modèle de maturité : application à l'industrie pharmaceutique*

JURY

Mme. Aline CAUVIN – Rapporteur

M. Jean-Pierre BOUREY – Rapporteur

M. Dominique CARRASCO – Examineur (directeur industriel)

M. Didier GOURC – Examineur (directeur académique)

M. François KIEFER – Examineur

M. Hervé PINGAUD – Examineur

Ecole doctorale : *Systèmes*

Unité de recherche : *Centre de Génie Industriel de l'ENSTIMAC*

Directeur(s) de Thèse : *Didier GOURC*

Rapporteurs : *Aline CAUVIN & Jean-Pierre BOUREY*

REMERCIEMENTS

Je ne peux que commencer par remercier chaleureusement mes trois encadrants qui ont été les véritables pierres angulaires de ces travaux. Ainsi, je remercie Didier GOURC, qui a su m'accompagner tout au long de ces trois dernières années en démontrant de formidables qualités humaines et professionnelles. Je remercie également Hervé PINGAUD, dont la stature n'a d'égale que l'intelligence et le charisme. Je remercie Dominique CARRASCO pour m'avoir donné une chance de m'épanouir dans un environnement industriel à travers des missions diverses et variées, qui possède une intelligence stratégique exceptionnelle.

Je remercie évidemment Aline CAUVIN et Jean-Pierre BOUREY qui ont accepté de rapporter cette thèse. Je suis, également très sensible, à l'honneur que m'ont fait François KIEFER et Lionel DUPONT, qui ont été respectivement examinateur et président du jury.

Je remercie le Centre de Génie Industriel de l'Ecole des Mines d'Albi Carmaux de m'avoir accepté durant ces trois années. Mes pensées vont plus particulièrement vers Isabelle FOURNIER pour la relecture du manuscrit et le support logistique durant ces trois ans ! Je ne peux pas citer tous les thésards et enseignants chercheurs du centre mais ils ont tous participé, sur le plan humain ou professionnel, à la réussite de cette aventure.

La Direction du Système d'Information des Laboratoires Pierre Fabre m'a donné la possibilité de réaliser ces recherches en gardant toujours à l'esprit l'intérêt industriel. Je remercie donc : José GARCIA, Catherine MARC-ANTOINE, Jean-Claude THIBOUT, Eric ESNEULT, Josette SALAS, Bruno KELLER, Béatrice GALLEE-LABORIE et tous les autres avec qui j'ai été en contact et avec qui j'ai eu plaisir à travailler.

Je remercie également Nicolas BAGUELIN, Gil GALLEZ et Vincent COLLIER pour l'aide qu'ils m'ont apportée sur la réglementation pharmaceutique.

Sur un plan plus personnel, je tiens à remercier mes deux sœurs et Annie qui m'ont toujours soutenu ainsi que ma mère, qui a accepté de relire et de corriger mes travaux. Elles ont été la clé de ma réussite.

Enfin, je remercie Cindy, Caroline, Julie, Docteur Mathieu, Sylvain, Mathieu, Gabriel pour le soutien psychologique du samedi soir !

TABLE DES MATIERES

REMERCIEMENTS 1

TABLE DES MATIERES 1

LISTE DES ILLUSTRATIONS 4

LISTE DES TABLEAUX 7

INTRODUCTION GENERALE ET PROBLEMATIQUE 8

1. L'industrie pharmaceutique 9

2. La qualité en informatique et le contexte de la DSI 13

3. Problématique des travaux et plan du document 15

CHAPITRE I. ETAT DE L'ART SUR LES MORPHISMES DE MODELES 17

1. Modèle, méta-modèle et méta-méta-modèle 18

 1.1. *Modèle* 18

 1.2. *Méta-modèle* 22

 1.3. *Méta-méta-modèle* 23

2. Opérations sur les modèles 26

 2.1. *Notion de morphisme* 26

 2.2. *Les morphismes « non altérants » : les mappings* 27

 2.3. *Les morphismes « altérants » : transformation de modèles* 34

3. Synthèse 37

CHAPITRE II. LES REFERENTIELS DE BONNES PRATIQUES INFORMATIQUES 38

1. Notions clés 41

 1.1. *Qualité et roue de Deming* 41

 1.2. *Processus et processus logiciel* 42

 1.3. *Capacité, maturité et modèle de maturité* 45

 1.4. *Origine des modèles de maturité : la QMMG* 50

2. Capability Maturity Model Integration (CMMI) 53

2.1. <i>L'origine du CMMI : le Capability Maturity Model (CMM)</i>	53
2.2. <i>Le CMMI</i>	54
3. La série des normes ISO 9000	65
3.1. <i>Historique</i>	65
3.2. <i>Composition de l'ISO 9000 : 1994</i>	65
3.3. <i>La composition de l'ISO 9000 : 2000 et le modèle de processus</i>	66
4. Le couple ISO 20000 / ITIL.....	71
4.1. <i>L'ISO 20000</i>	71
4.2. <i>L'ITIL : la facette « métier » de la norme ISO 20000</i>	73
5. Synthèse.....	76
6. Conclusion	83

**CHAPITRE III. EXIGENCES REGLEMENTAIRES PHARMACEUTIQUES
APPLICABLES AUX SYSTEMES INFORMATISES 84**

1. Les autorités réglementaires	86
1.1. <i>L'autorité américaine : la Food and Drug Administration (FDA)</i>	86
1.2. <i>L'autorité européenne : European MEDicines Agency (EMA) et l'autorité française : Agence Française de Sécurité Sanitaire des Produits de Santé (AFSSAPS)</i> . 87	
2. Périmètre	90
3. La réglementation pharmaceutique applicable aux systèmes informatisés.....	91
3.1. <i>Textes américains</i>	91
3.2. <i>Textes européens</i>	92
3.3. <i>Le GAMP et le Volume 5</i>	94
4. Vers un référentiel d'exigences.....	95
4.1. <i>Qu'est-ce qu'une exigence ?</i>	95
4.2. <i>L'élaboration du référentiel d'exigences à partir des textes réglementaires</i>	98
5. Synthèse.....	105

CHAPITRE IV. LA MISE EN RELATION DES REFERENTIELS 106

1. Présentation générale du mapping réalisé.....	107
1.1. <i>Etat de l'art sur les mappings entre référentiels de bonnes pratiques informatiques existants</i>	107
1.2. <i>Notre mapping</i>	109
2. Mise en relation	114
2.1. <i>Notations, variables et indices</i>	114
2.2. <i>Deux méthodes de résolution</i>	117
3. Synthèse et choix.....	121

CHAPITRE V. APPLICATION INDUSTRIELLE ET RESULTATS OBTENUS	123
1. La construction des relations et la réalisation du solveur.....	125
1.1. <i>Construction des relations</i>	125
1.2. <i>Réalisation du solveur</i>	126
2. Résultats du mapping	128
2.1. <i>Mapping entre les pratiques du CMMI et les exigences réglementaires</i>	128
2.2. <i>Résultats du mapping global</i>	132
3. Exploitation des résultats du mapping et du travail d'analyse des exigences réglementaires....	134
3.1. <i>Plan Qualité de la DSI</i>	134
3.2. <i>Exploitation du référentiel d'exigences réglementaires : construction des diagrammes d'Ishikawa</i>	136
 CONCLUSION ET PERSPECTIVES.....	 140
 LISTE DES ACRONYMES UTILISES	 142
 BIBLIOGRAPHIE	 146
 ANNEXES.....	 164
La QMMG	165
Le CMM.....	168
Les plans des deux livres de la norme ISO 20000.....	172
La validation pharmaceutique.....	173
<i>Qualification de Conception (QC)</i>	173
<i>Qualification d'Installation (QI)</i>	174
<i>Qualification Opérationnelle (QO)</i>	174
<i>Qualification de Performance (QP)</i>	175
<i>Classification GAMP</i>	175
<i>Cycle en V et Validation</i>	176
Diagrammes d'Ishikawa	177

LISTE DES ILLUSTRATIONS

Figure 0-1 : cycle de vie du médicament et réglementation [Laboratoires Pierre Fabre, 2004].....	11
Figure 0-2 : cycle de vie applicatif et référentiels de bonnes pratiques (inspiré de [Villeneuve <i>et al.</i> , 2001])	14
Figure 0-3 : architecture du document.....	16
Figure I-1 : les usages du terme « modèle »	19
Figure I-2 : modèles et points de vue	21
Figure I-3 : exemple d'un système étudié et de modèles associés (inspiré de [Bézivin <i>et al.</i> , 2004])	21
Figure I-4 : un système et un point de vue peuvent être modélisés dans différents langages.....	22
Figure I-5 : architecture à quatre niveaux (inspirée de [OMG , 2002])	24
Figure I-6 : formalisme habituellement utilisé pour représenter les morphismes.....	26
Figure I-7 : mapping vertical et mapping horizontal.....	28
Figure I-8 : mapping endogène : $m_1(s)/f_1$ et $m_2(s)/f_1$	29
Figure I-9 : mapping exogène : $m_1(s)/f_1$ et $m_2(s)/f_2$	30
Figure I-10 : règles de mises en relation.....	30
Figure I-11 : relations entre mapping et formalisme [Caplat <i>et al.</i> , 2003].....	31
Figure I-12 : illustration de la notion de cardinalité des mappings (avec $m=n=2$) (inspirée de [Lopes <i>et al.</i> , 2006]).....	32
Figure I-13 : mapping unidirectionnel ou bidirectionnel.....	32
Figure I-14 : diagramme de classes en UML de la notion de mapping	33
Figure I-15 : le processus de transformation de modèles [Touzi, 2007]	35
Figure I-16 : transformation de modèles [Lemesle, 1998].....	36
Figure II-1 : les différentes « couches » du système d'information.....	38
Figure II-2 : roue de Deming.....	42
Figure II-3 : vue partielle du méta-modèle SPEM : le processus [Combemale <i>et al.</i> , 2006]	43
Figure II-4 : représentation schématique d'un processus	44
Figure II-5 : évolution de l'espèce humaine au cours des âges	48
Figure II-6 : lien entre maturité et capacité	49
Figure II-7 : extrait du CMMI	55
Figure II-8 : bonnes pratiques	56
Figure II-9 : pratique et activité [Miler <i>et al.</i> , 2004]	57
Figure II-10 : les principaux composants du modèle CMMI.....	60
Figure II-11 : correspondances entre niveaux de maturité et niveaux de capacité du CMMI [Chandelon, 2004]	63
Figure II-12 : noyau structurel du CMMI en UML.....	64
Figure II-13 : cartographie de l'ISO 9000 : 1994 [Mathieu, 2000]	66
Figure II-14 : exemple d'annexe de la norme ISO 9000 [AFNOR, 2000].....	67
Figure II-15 : l'enchaînement de la série ISO 9000 [Mathieu, 2002]	69
Figure II-16 : modèle conceptuel de l'ISO 9000 : 2000.....	70
Figure II-17 : modélisation UML de l'ISO 20000.....	72

Figure II-18 : les modules ITIL.....	73
Figure II-19 : positionnement du modèle conceptuel générique des référentiels de bonnes pratiques dans l'architecture de l'OMG	76
Figure II-20 : modèle conceptuel en UML des référentiels de bonnes pratiques.....	80
Figure II-21 : exemples d'instances d'objets	81
Figure III-1 : cycle de vie simplifié du médicament et textes réglementaires.....	89
Figure III-2 : les BPL applicables aux systèmes informatisés.....	92
Figure III-3 : les BPF applicables aux systèmes informatisés (Annexe 11)	93
Figure III-4 : qui peut émettre des exigences ? Qui peut y être soumis ?	97
Figure III-5 : exigence et processus.....	97
Figure III-6 : travail d'analyse, de réorganisation et de découpage des textes réglementaires	99
Figure III-7 : l'élaboration du référentiel d'exigences unifié	100
Figure III-8 : répartition des exigences par catégories	102
Figure III-9 : positionnement du modèle conceptuel de référentiel d'exigences réglementaires dans l'architecture de l'OMG	103
Figure III-10 : modèle conceptuel du référentiel d'exigences.....	104
Figure IV-1 : positionnement des modèles dans l'architecture de l'OMG	109
Figure IV-2 : exemple de combinaisons d'éléments de Ref ₂ (P _i) équivalentes à un élément de Ref ₁ (E _i).....	115
Figure IV-3 : représentation logique de l'exemple de la Figure IV-2.....	115
Figure IV-4 : FCD de l'expression de la Figure IV-2 – forme littérale	115
Figure IV-5 : FCD de l'expression de la Figure IV-2 – forme algébrique	115
Figure IV-6 : illustration de la notation utilisée sur une FCD – notation algébrique (exemple fictif)	117
Figure IV-7 : exemple de FCD pour la seconde méthode	119
Figure V-1 : plan de la partie	124
Figure V-2 : activités pour identifier les relations.....	125
Figure V-3 : méthodes de résolution	127
Figure V-4 : répartition des PA et SP par niveaux	128
Figure V-5 : couverture des exigences par les pratiques du CMMI.....	128
Figure V-6 : répartition des exigences couvertes par niveaux de maturité du CMMI.....	129
Figure V-7 : les pratiques du niveau 2 mises en jeu par le mapping	129
Figure V-8 : les pratiques du niveau 3 mises en jeu par le mapping	129
Figure V-9 : les pratiques mises en jeu par PA	130
Figure V-10 : nombre de références par PA	131
Figure V-11 : répartition des non-couvertures par catégories	131
Figure V-12 : cadre qualité.....	133
Figure V-13 : les dimensions de l'amélioration.....	134
Figure V-14 : le système documentaire de la DSI [Carrasco, 2006].....	135
Figure V-15 : diagramme d'Ishikawa et 5M	137
Figure V-16 : exemple de diagramme d'Ishikawa.....	139
Figure A-1 : représentation schématique de l'évolution des processus d'une organisation avec l'augmentation de la maturité	169

Figure A-2 : les secteurs clés couverts par chacun des 5 niveaux du CMM [CRAM, 1993]	170
Figure A-3 : instanciation pour une pratique clé [CRAM,1993].....	171
Figure A-4 : le cycle en V et la validation [Cabanne, 2001].....	176
Figure A-5 : diagramme d'Ishikawa pour relatif au « groupe de risque » 2.....	178
Figure A-6 : diagramme d'Ishikawa pour relatif au « groupe de risque » 3.....	179
Figure A-7 : diagramme d'Ishikawa – zoom sur le système d'assurance de la qualité	180
Figure A-8 : diagramme d'Ishikawa – zoom sur les bonnes pratiques de tests.....	181
Figure A-9 : diagrammes d'Ishikawa – zoom sur Gestion des compétences et documents.....	182

LISTE DES TABLEAUX

Tableau II.1 : exemple de capacité.....	47
Tableau II-2 : extrait de la QMMG de Crosby [Fraser <i>et al.</i> , 2002].....	51
Tableau II-3 : organisation des secteurs clés par niveau de maturité et catégories.....	62
Tableau II-4 : le module « soutien des services informatiques » [Eurotil, 2004].....	74
Tableau II-5 : le domaine « fourniture de services informatiques » [Eurotil, 2004]	74
Tableau II-6 : taxonomie des composants des référentiels de bonnes pratiques.....	77
Tableau IV-1 : synthèse des travaux existants sur les mappings CMMI	108
Tableau IV-2 : correspondances entre exigences et bonnes pratiques	111
Tableau IV-3 : les opérateurs et leur représentation	112
Tableau IV-4 : comparaison des deux méthodes envisagées	122
Tableau V-1 : les 5M et les risques.....	138
Tableau A-1 : la QMMG	165
Tableau A-2 : structure des deux livres de la norme ISO 20000 (traduction personnelle) [ISO, 2005a] et [ISO, 2005b].....	172
Tableau A-3 : classification GAMP [ISPE, 2001a].....	175

INTRODUCTION GENERALE ET PROBLEMATIQUE

1. L'industrie pharmaceutique

L'industrie pharmaceutique est le secteur économique qui regroupe les activités de recherche, de fabrication et de commercialisation des médicaments à usage humain ou animal. Elle est caractérisée par des perspectives de croissance forte et par une importante internationalisation, dominée par des groupes à dimension mondiale.

Les entreprises françaises ont su s'imposer sur ce marché, ce qui fait de ce secteur une des branches dynamiques de notre économie. Le secteur pharmaceutique français se situe au troisième rang mondial pour les exportations de médicaments, ainsi qu'en nombre de médicaments « découverts ». Son chiffre d'affaires croît de 6,4 % par an depuis 1990 et de près de 8 % par an depuis 1995. En 2007, il s'est élevé à près de quarante cinq milliards d'euros [<http://www.leem.org>] !

Cependant, il ne faut pas considérer le médicament comme un produit de consommation « classique ». Quelques caractéristiques lui confèrent un statut particulier :

- il a une **vocation de santé publique** : c'est un produit réglementé, qui n'est pas soumis aux mêmes lois de l'offre et de la demande qu'un produit de consommation courante,
- c'est un **produit actif** nécessaire à la santé, mais qui peut comporter des **risques** : c'est pourquoi la totalité du cycle (production, dispensation, récupération) du médicament est très étroitement encadrée et confiée à la responsabilité de pharmaciens,
- il a un **mode de financement spécifique** : dans le cadre de la solidarité collective, les organismes de protection sociale peuvent prendre en charge une partie ou la totalité de la dépense pharmaceutique,
- c'est un **bien industriel** : il est fabriqué par des entreprises dont la rentabilité doit assumer une recherche de très haut niveau et coûteuse.

C'est aussi un secteur soumis à des interrogations sur le futur et les défis à relever :

- les récentes vagues de **déremboursement** des médicaments : elles vont nécessairement avoir un impact sur les ventes des médicaments, car elles pointent du doigt un ensemble de produits jugés comme ayant un Service Médical Rendu (SMR) insuffisant,
- l'apparition des **médicaments génériques** : elle remet en question l'avenir des visiteurs médicaux,

- les **coûts liés à la recherche et au développement** de nouveaux médicaments qui explosent : sortir un nouveau médicament coûte en moyenne huit cent millions de dollars [DiMasi *et al.*, 2003]. Ce chiffre double tous les cinq ans depuis les années 1970,
- la **pression réglementaire** qui augmente (protectionnisme, augmentation de périmètre, *etc.*).

Cette réglementation impose l'utilisation de certaines méthodes, le respect de certaines pratiques, *etc.*, afin de s'assurer que le médicament produit ne présente aucun danger pour les patients et que toutes ses caractéristiques sont « maîtrisées ». Comme il est impossible de définir une réglementation unique, valable pour toutes les étapes du cycle de vie du médicament, les instances réglementaires proposent des textes adaptés à chaque phase du cycle de vie. C'est ce que nous illustrons avec la Figure 0-1.

Comme le montre la Figure 0-1, le cycle de vie d'un médicament est composé de quatre phases principales :

- **la recherche** : il s'agit de concevoir différentes molécules et d'en optimiser certaines afin d'obtenir un ou des principes actifs [<http://www.leem.org>],
- **le développement** : il s'agit de réaliser les études précliniques et cliniques. Les études précliniques consistent à analyser les effets d'une molécule injectée à un animal, essais d'efficacité sur des systèmes moléculaires inertes, tests toxicologiques... L'objectif des études cliniques est d'analyser les effets des principes actifs chez l'homme. Les réglementations qui s'appliquent à cette phase sont les « Bonnes Pratiques de Laboratoires » et les « Bonnes Pratiques Cliniques »,
- **l'autorisation**. Elle consiste à déposer le dossier d'enregistrement du médicament dans l'agence adéquate, qui délivrera éventuellement l'Autorisation de Mise sur le Marché¹ (AMM),
- **la production et la distribution** sont réglementées par les « Bonnes Pratiques de Fabrication » et par les « Bonnes Pratiques de Distribution », qui encadrent les activités de pharmacovigilance.

¹ « L'Autorisation de mise sur le marché (AMM) est attribuée à un produit pour permettre sa commercialisation sous forme de spécialité pharmaceutique » (Spécialité : tout médicament préparé à l'avance présenté sous un conditionnement caractérisé par une dénomination spéciale) [Ferry, 2000].

RECHERCHE		DEVELOPPEMENT				AUTORISATION	PRODUCTION DISTRIBUTION	
1 à 2 ans		1 à 2 ans	1 à 2 ans	6 à 8 ans			1 à 2 ans	vie
Recherche des points de départ chimique Conception Synthèse	Optimisation Puissance Sélectivité Passage <i>per os</i> ¹ Durée d'action	Evaluation pré-clinique <i>in vitro</i> et chez l'animal Activité Caractérisation Stabilité Sécurité Devenir dans l'organisme	Evaluation clinique chez l'homme			Dépôt du dossier d'enregistrement du médicament	Commercialisation	
			Phase 1 Tolérance et pharmacocinétique	Phase 2 Activité biologique et recherche d'un effet thérapeutique	Phase 3 Confirmation de l'effet thérapeutique et de la tolérance		Phase 4 Pharmacovigilance	
1000 molécules		50 molécules	10 molécules	3 molécules	1 molécule			
		Réglementation : Bonnes Pratiques de Laboratoires	Réglementation : Bonnes Pratiques Cliniques				Réglementation : Bonnes Pratiques de Fabrication Bonnes Pratiques de Distribution	

Figure 0-1 : cycle de vie du médicament et réglementation [Laboratoires Pierre Fabre, 2004]

¹ Voie d'administration orale

Toutes les activités, qui peuvent avoir un impact, direct ou indirect, sur la santé de l'homme par l'intermédiaire des médicaments, sont concernées par la réglementation. De plus, c'est une évidence de dire que l'informatique prend de plus en plus d'importance dans toutes les industries, et ce, à une vitesse redoutable ! L'industrie pharmaceutique ne fait pas exception. Les systèmes de gestion de données, les outils de travail collaboratif, les systèmes de gestion documentaires et de publication, les applications Internet et Intranet, les systèmes de calcul, de résolution de problèmes, les outils utilisés en laboratoire, les outils de contrôle de processus de production, *etc.*, sont légion [Basavapathruni, 2007]. Il n'est donc pas étonnant que les textes réglementaires s'intéressent aux cas des « systèmes informatisés » utilisés dans les différentes phases du cycle de vie du médicament, et donc aux activités de conception, de réalisation et de maintenance de ces systèmes.

Ce sont précisément ces activités qui vont nous intéresser. En effet, nous travaillons avec la Direction des Systèmes d'Information (DSI) des Laboratoires Pierre Fabre. Ce groupe est le deuxième laboratoire pharmaceutique français indépendant. La DSI est donc soumise à la réglementation pharmaceutique et se doit donc de la respecter, tout en étant conforme aux « règles du métier de la production de logiciels ».

Elle a pour mission « d'aligner le système d'information (des Laboratoires Pierre Fabre) sur la stratégie validée par le Comité Informatique Groupe¹, et les besoins opérationnels des métiers du Groupe Pierre Fabre, en France comme à l'International, en apportant des solutions fiables, ergonomiques et compétitives » [Carrasco, 2006]. Concrètement, la DSI est en charge de toutes les activités liées à la conception, au choix, à la réalisation, au paramétrage, à la maintenance et au support des diverses applications informatiques utilisées dans le Groupe Pierre Fabre.

¹ « Le CIG a pour mandat de valider la stratégie informatique du groupe, de suivre la mise en œuvre de cette stratégie, de valider le plan à cinq ans et le budget annuel, d'arbitrer sur les priorités des grands programmes, et les conflits sur les engagements humains et financiers, de valider les grandes orientations organisationnelles et techniques » [Carrasco, 2006].

2. La qualité en informatique et le contexte de la DSI

Produire toujours mieux, toujours moins cher est un leitmotiv commun à toutes les entreprises, quel que soit le domaine d'activité examiné. Les entreprises œuvrant dans le domaine des systèmes d'information ne sont donc pas épargnées. L'amélioration du niveau de qualité de leurs prestations et de leurs produits, la maîtrise de leurs processus et donc des coûts associés peuvent se faire en utilisant plusieurs dizaines de méthodes différentes. Celles qui vont nous intéresser ici sont celles que nous désignons par l'expression « référentiels de bonnes pratiques ».

Globalement, il s'agit de normes, de modèles de maturité du processus logiciel et de recueil de bonnes pratiques. C'est un monde immense. Le terme de « référentiel » sous-entend qu'il s'agit de documents « références », c'est-à-dire de documents validés et approuvés par des organismes compétents, qui font foi dans les domaines considérés. Certains référentiels couvrent plusieurs phases du cycle de vie applicatif, d'autres sont très spécialisés, d'autres, encore, ne concernent qu'un processus transverse. Certains sont dédiés aux petites et moyennes entreprises, alors que d'autres sont universels !

Il est intéressant d'essayer d'obtenir une macro-cartographie de ces référentiels, en les calquant sur un cycle de vie applicatif. C'est ce que nous proposons dans la Figure 0-2. Le cycle de vie générique utilisé est inspiré d'un travail proposé par [Villeneuve *et al.*, 2001]. La Figure 0-2 n'a pas la prétention d'être exhaustive, son but est juste d'illustrer la diversité des référentiels existants. Chaque référentiel est précédé du signe « ► ».

La DSI des Laboratoires Pierre Fabre a mis en œuvre un de ces référentiels : le référentiel CMMI (« Capability Maturity Model Integration »), qui guide l'analyse, la conception, la réalisation et la maintenance de ses applications. Son système de management de la qualité est, lui, basé sur la famille des normes ISO 9000 : 2000. De plus, elle n'assure pas elle-même l'exploitation des systèmes qu'elle conçoit et réalise : c'est une activité sous-traitée à un fournisseur (DVO) qui, lui, a mis en place une démarche qualité basée sur l'ISO 20000.

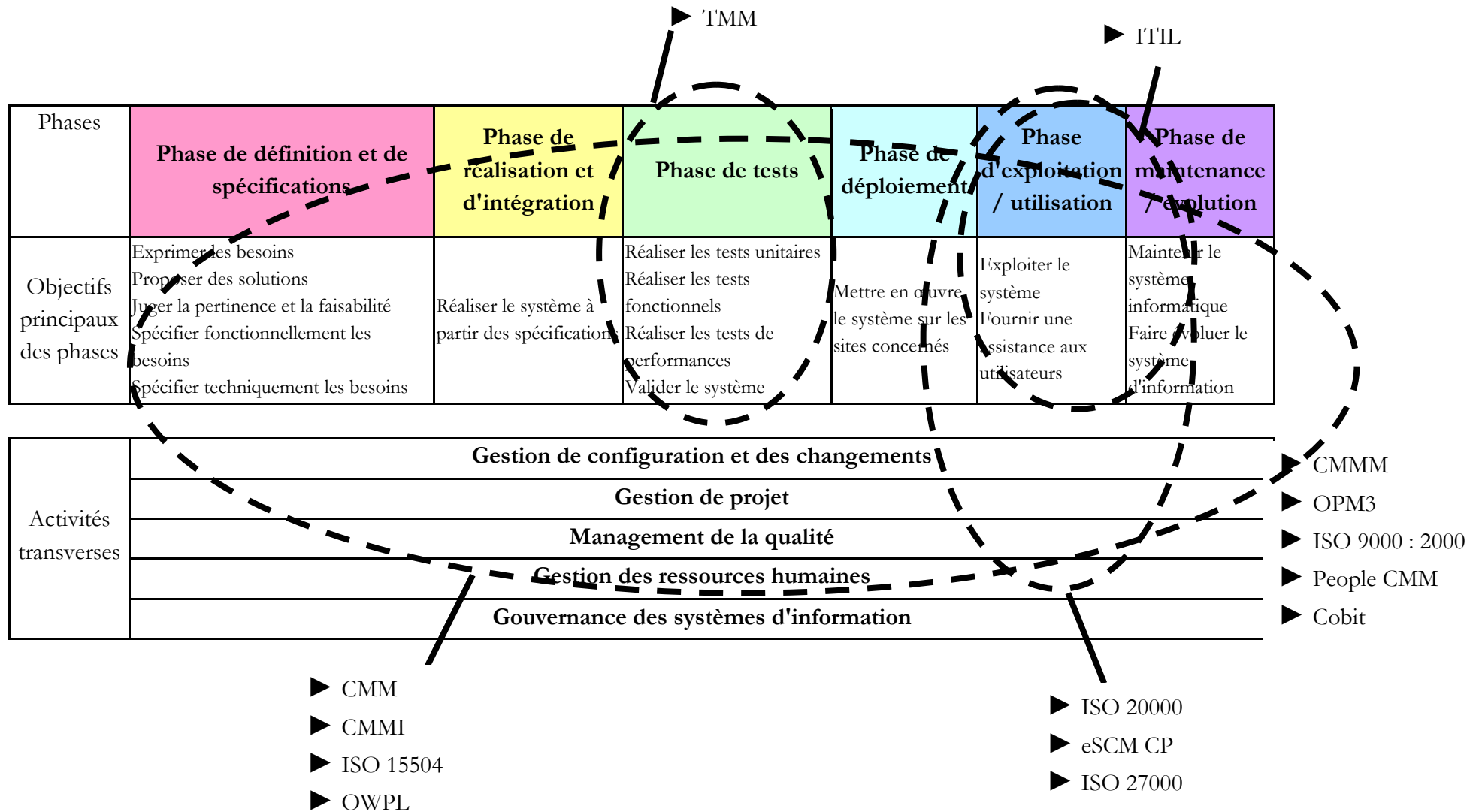


Figure 0-2 : cycle de vie applicatif et référentiels de bonnes pratiques (inspiré de [Villeneuve *et al.*, 2001])

3. Problématique des travaux et plan du document

Nous avons donc, d'un côté, la réglementation pharmaceutique, qui impose aux organisations pharmaceutiques certaines méthodes et démarches pour obtenir des médicaments de qualité. Ces contraintes s'appliquent également aux DSI, en tant qu'organisme « support ». De l'autre côté, ces DSI peuvent avoir mis en place, volontairement, des référentiels de bonnes pratiques informatiques qui proposent des orientations concernant les méthodologies de travail, pour l'industrie du logiciel.

Il apparaît donc légitime de s'interroger sur la similarité entre les attendus des référentiels de bonnes pratiques informatiques et ceux de la réglementation pharmaceutique. En d'autres termes, une DSI qui a mis en place une démarche qualité basée sur un ou des référentiel(s) de bonnes pratiques, répond-elle aux exigences réglementaires ? C'est cette question qui est à l'origine de nos travaux.

Sur le plan scientifique, ce type de problématique n'est pas nouveau. D'une part les modèles de maturité et les référentiels de bonnes pratiques sont très largement manipulés par des études récentes [Ferchichi, 2008], [Gonzales-Ramirez, 2006] et d'autre part la comparaison de référentiels entre eux est également abordée. [Mutafelija *et al.*, 2003a], par exemple, compare le CMMI et la norme ISO 9000 : 2000. L'originalité de notre question vient donc de la nécessité d'intégrer des contraintes relatives au secteur d'activité dans lequel nous évoluons : le monde pharmaceutique.

Afin de présenter notre problématique et proposer notre approche de résolution, nous avons conçu un mémoire en cinq chapitres. Ils sont précédés par cette introduction générale qui nous permet de présenter le contexte et la problématique de nos travaux, ainsi que le plan du document.

Avant de pouvoir comparer les référentiels de bonnes pratiques informatiques et la réglementation pharmaceutique, nous nous interrogeons sur la faisabilité technique de la comparaison de deux modèles afin de répondre à la problématique. Quels sont les outils qui existent ? Comment procéder ? Nous nous sommes rattachés au domaine de l'Ingénierie Dirigée par les Modèles (IDM) afin de trouver des réponses. Ainsi, l'utilisation des morphismes de modèles sera l'objet du **premier chapitre**.

Nous nous sommes ensuite concentrés sur l'idée de créer un modèle d'informations, décrivant le contenu de chacun des référentiels relatif à notre contexte industriel.

Une présentation structurée des différents référentiels de bonnes pratiques mis en jeu dans « l'environnement de la DSI des Laboratoires Pierre Fabre » sera donc réalisée dans le **second chapitre**. Elle nous permettra d'obtenir un modèle conceptuel de ces référentiels.

Le **troisième chapitre** sera dédié à la réglementation pharmaceutique applicable aux systèmes informatisés. Une présentation des instances réglementaires et des textes auxquels la DSI est soumise nous permettra d'établir un modèle conceptuel relatif aux exigences réglementaires pharmaceutiques.

Le **quatrième chapitre** présentera la méthodologie de comparaison et la réponse apportée au problème de génie industriel que nous avons rencontré lors de la mise en relation des référentiels.

Les résultats industriels obtenus, en appliquant la méthodologie présentée au chapitre quatre, seront l'objet du **chapitre cinq**.

En fin de mémoire nous proposons une conclusion. Nous présenterons également les principales perspectives de recherche identifiées.

La logique de ce plan est présentée sur la Figure 0-3.

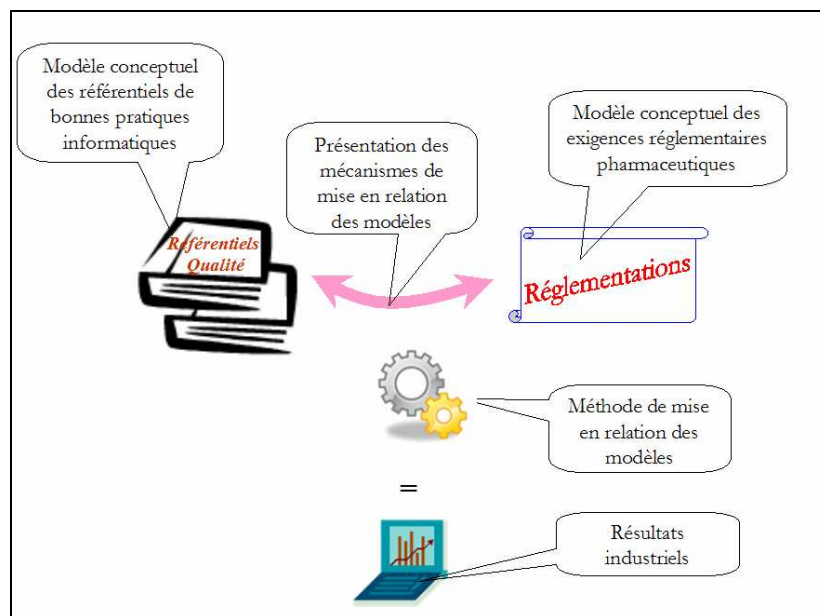


Figure 0-3 : architecture du document

En plus de ces chapitres, ce mémoire propose (en fin de document) :

- un résumé des nombreux acronymes utilisés,
- une liste des références bibliographiques citées,
- des annexes, qui apportent des compléments d'information sur quelques sujets que nous n'avons pas détaillés dans le corps du mémoire.

Chapitre I.

ETAT DE L'ART SUR LES MORPHISMES DE MODELES

Les modèles tiennent une place de choix dans l'ingénierie des systèmes d'information. Au niveau des développements informatiques, le passage de la technologie procédurale à la technologie objet a rendu, par exemple, quasi-obligatoire de faire précéder la phase de réalisation technique d'une phase de modélisation. De même, en ingénierie des processus, les modèles sont souvent considérés comme la brique élémentaire sur laquelle il faut s'appuyer afin de comprendre et améliorer les processus.

Une nouvelle branche de l'ingénierie a donc logiquement vu le jour, l'Ingénierie Dirigée par les Modèles (IDM). L'IDM « est le domaine de l'informatique mettant à disposition des outils, concepts et langages pour créer et transformer des modèles » [Wikipedia, 2008a]. Cette discipline propose de nouvelles approches qui se décomposent généralement en deux phases distinctes :

- la construction (élaboration) d'un ou de plusieurs modèle(s),
- la manipulation de ces derniers.

Derrière le terme « manipulation » se cachent plusieurs opérations, qui tirent leurs origines des mathématiques. Elles sont également désignées par l'expression « morphismes de modèles ». L'étude de ces derniers est l'objet de ce chapitre.

Nous nous attachons tout d'abord à définir rigoureusement les termes clés qui seront utilisés tout au long de notre étude : modèle, méta-modèle et méta-méta-modèle. Une fois ces définitions établies et illustrées, nous détaillons la notion de « morphisme », en adoptant successivement un point de vue mathématique et un point de vue IDM. Un des objectifs de ce chapitre est d'identifier les caractéristiques portées par une famille particulière de morphismes de modèles : les « **mappings** ». Ces caractéristiques seront détaillées et illustrées par de nombreux exemples.

1. Modèle, méta-modèle et méta-méta-modèle

Nous étudions donc dans les parties suivantes les termes : modèle, méta-modèle et méta-méta-modèle. Notre objectif est de réaliser un état de l'art quant à leurs définitions, ainsi que de préciser les positions adoptées. Cette partie nous permet également de présenter l'architecture à quatre niveaux, proposée par l'OMG¹ (Object Management Group), qui positionne ces différents « niveaux » de modèles les uns par rapport aux autres.

1.1. Modèle

Que ce soit dans les sciences physiques, les sciences de l'ingénieur, les sciences humaines... le concept de « modèle » est très largement utilisé. Pourtant, ce « dénominateur commun » n'a encore fait l'objet d'aucun consensus général quant à sa définition : une définition transversale n'existe pas (et en déterminer une n'est pas l'objet de cette partie). Ainsi, « modèle » n'a pas la même signification dans les sciences humaines ou dans le monde artistique. Or, comme nous utilisons cette notion à maintes reprises dans nos travaux, il est primordial de préciser ce que nous entendons par ce terme. Intéressons-nous, tout d'abord, aux « sens communs » du terme, puis aux définitions propres à notre domaine : les systèmes d'information.

Les dictionnaires contemporains différencient principalement cinq usages différents [Armatte, 2005] concernant le terme « modèle » :

- (a) le modèle en tant que référent, par exemple, en peinture : un modèle est « ce qu'il faut reproduire »,
- (b) le modèle en tant que maquette d'un dispositif réel : c'est le cas en architecture où le modèle est une représentation, à échelle réduite, d'un ouvrage,
- (c) le modèle en tant qu'idéal absolu : il s'agit par exemple d'un modèle de beauté, de bonté...),
- (d) le modèle en tant que dispositif mécanique représentant une idée abstraite. En ce sens, la notion de modèle s'apparente à une « métaphore », puisqu'il permet de « concrétiser » une idée abstraite : le modèle mécanique de Faraday pour les flux de force électromagnétique, ou celui de Maxwell relatif à l'électrodynamique [Paty, 1994] en sont des illustrations,
- (e) le formalisme logico-mathématique, par exemple, l'équation de Schrödinger, qui décrit l'évolution de l'état d'une particule en mécanique quantique.

¹ « Object Management Group est une association « américaine à but non-lucratif créée en 1989 dont l'objectif est de standardiser et promouvoir le modèle objet sous toutes ses formes » [Wikipedia, 2007].

Les significations (a) (b) et (c) présentent alternativement le modèle soit comme étant la réalité à reproduire (a), soit comme en étant une représentation de son futur¹ (b), ou soit comme la *référence parfaite* (c). Dans les trois cas, le modèle a pour objectif de « guider » une réalisation. La bibliographie utilise également les termes « modèle descriptif » (« backward-looking model ») lorsqu'un modèle décrit un *système existant*, et « modèle prescriptif » (« forward-looking model ») lorsqu'un modèle décrit un *système envisagé* [Gonzalo-Perez *et al.*, 2007], [Ludewig, 2003].

Les significations (d) et (e) le placent plutôt comme une abstraction de la réalité. Le modèle sert alors d'interprétation ou de support d'anticipation [Terrasse *et al.*, 2005]. La démarche d'élaboration d'un modèle vise alors à la compréhension, au diagnostic d'une réalité, ou à la simulation du « comportement d'un système » (simulation statistique, numérique, stochastique...) [Wikipedia, 2008b].

La Figure I-1 illustre ces différentes significations par rapport à la réalité.

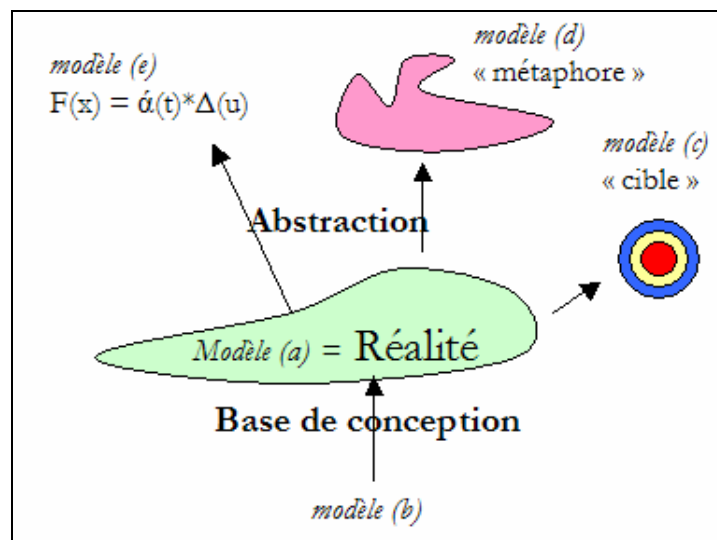


Figure I-1 : les usages du terme « modèle »

Ainsi, selon le sens (a), le modèle est la réalité. Le sens (b) place le modèle comme « base de conception » pour réaliser un ouvrage. Les sens (d) et (e) sont issus d'un processus d'abstraction de la réalité.

Cette notion de modèle en tant qu'abstraction n'est pas nouvelle, puisqu'elle était déjà énoncée par Boltzmann au début du vingtième siècle : « nos pensées sont aux choses dans le même rapport que les modèles sont aux objets qu'ils représentent » [Boltzmann, 1902] !

Dans nos disciplines, les définitions académiques les plus utilisées sont en accord avec cette vision. Vernadat [Vernadat, 99] définit le modèle comme :

¹ En architecture ou en fonderie par exemple, le modèle est construit avant l'ouvrage.

« une représentation d'une abstraction d'une partie du monde réel, exprimée dans un langage de représentation ».

C'est cette définition que nous prendrons comme référence par la suite. La vision moderne rajoute donc la notion de « langage de représentation », qui peut être soit formel (c'est-à-dire ayant une syntaxe bien définie) ou informel (langage naturel).

Doumeingts [Doumeingts, 1990] définit ce langage de représentation (*i.e.*: langage de modélisation) comme étant :

« une représentation de la connaissance à communiquer sans ambiguïté. Il permet de construire des modèles selon des composants associés ».

L'activité de représenter un système dans un langage de représentation est un acte de modélisation. La modélisation est définie par [Le Moigne, 1990] comme :

« l'élaboration et la construction intentionnelle par composition de symboles, de modèles susceptibles de rendre intelligible un phénomène perçu complexe, et d'amplifier le raisonnement de l'acteur projetant une intention délibérée au sein du phénomène ».

Un modèle est donc indissociable de la syntaxe utilisée pour le représenter. Celle-ci dépend de *l'intention* de la personne qui modélise. Modéliser un système, dans un langage donné, ne permet pas (et ce n'est pas le but d'une telle démarche) de représenter toutes les facettes de ce système (statique / dynamique, fonctionnelle / structurelle...). Puisque modéliser est une activité d'abstraction, un modèle donné ne concernera simultanément qu'une ou plusieurs facette(s) du système, en fonction de l'objectif recherché, c'est-à-dire en fonction du point de vue adopté¹. Un système pourra donc être décrit par un ensemble de modèles. C'est ce que nous illustrons, très schématiquement, par la Figure I-2. Le nombre de modèles et de points de vue nécessaires pour appréhender et comprendre un système est directement lié à sa complexité. Plus le système sera complexe, plus le nombre de modèles nécessaires sera élevé. De même, plus le besoin à l'origine d'une modélisation est vaste, plus le nombre de modèles nécessaires sera élevé.

¹ Cette notion de « point de vue » telle que nous la présentons ici s'écarte donc de celle utilisée en modélisation d'entreprise.

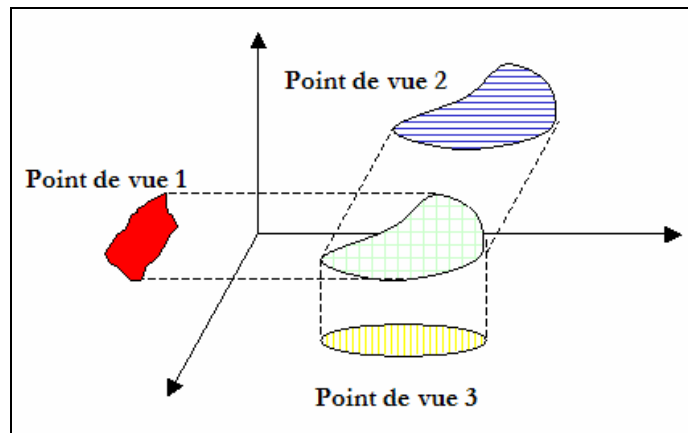


Figure I-2 : modèles et points de vue

En fonction du point de vue choisi (1, 2 ou 3 sur la Figure I-2), le résultat de l'activité de modélisation d'un système étudié sera tout à fait différent. Nous illustrons également ceci par la Figure I-3, sur la base d'un exemple plus concret.

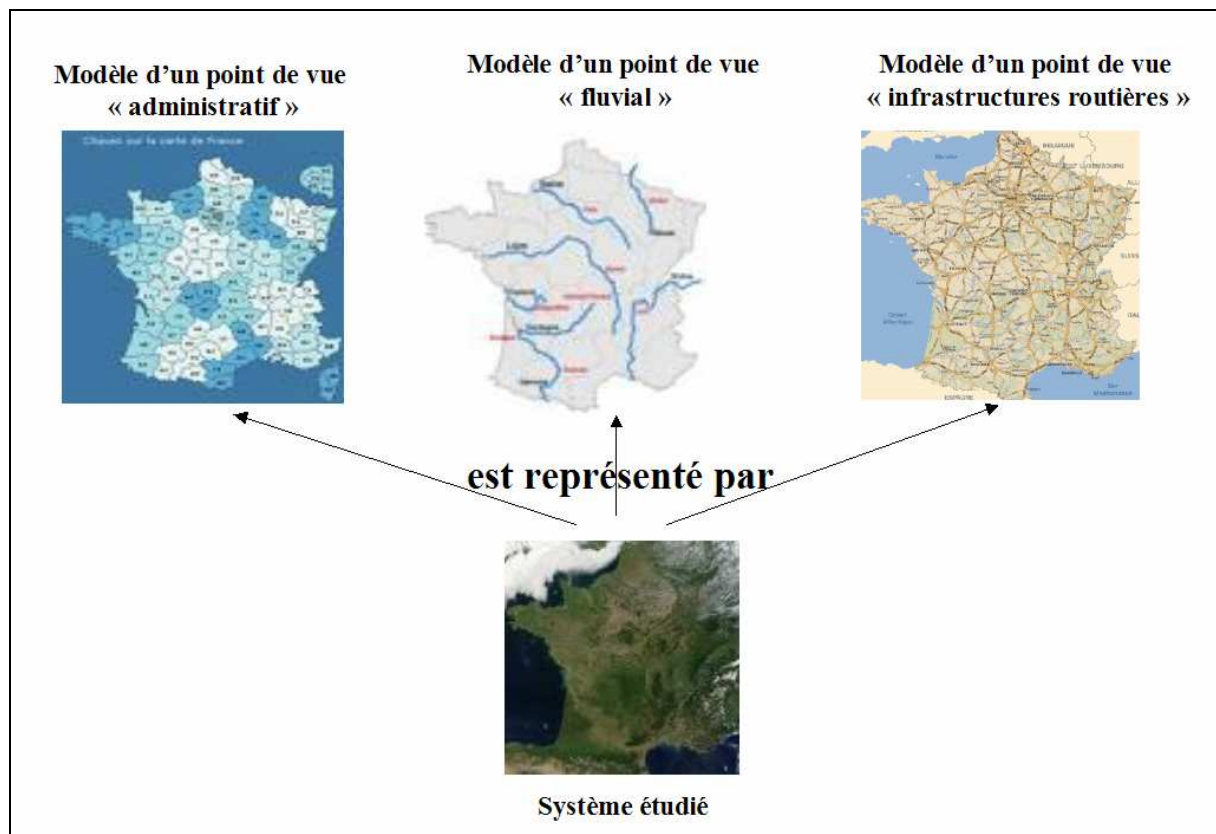


Figure I-3 : exemple d'un système étudié et de modèles associés [Bézivin *et al.*, 2004]

Le système étudié est « la France métropolitaine ». Trois points de vue différents sont adoptés successivement : « administratif », « fluvial » et « infrastructures routières ». Les trois modèles obtenus ne présentent pas, par conséquent, la même information, bien qu'ils soient tous des modèles du même système.

1.2. Méta-modèle

Pour un point de vue et un système donnés, une multitude de modèles différents sont envisageables (ceci est illustré par la Figure I-4).

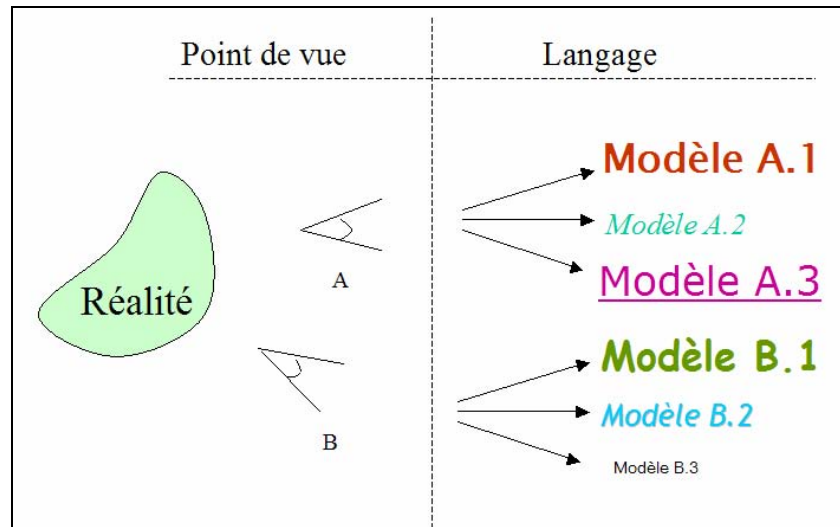


Figure I-4 : un système et un point de vue peuvent être modélisés dans différents langages

Le langage utilisé, pour représenter le système étudié, aura un impact direct sur la forme du résultat obtenu par l'acte de modélisation. Sur l'exemple de la Figure I-4, chaque « modèle A.x » porte la même connaissance, mais pas de la même manière. C'est le langage utilisé qui crée cette différence.

Ce langage est défini par une sémantique associée à chacun des éléments qui le composent (à quoi correspond un rond, un carré... ?), ainsi que par sa syntaxe (c'est-à-dire la manière de mettre en relation les composants qu'il contient). La définition de ces éléments permet de s'assurer qu'un modèle sera compris de la même manière par l'ensemble des utilisateurs, c'est-à-dire que tout le monde sache ce que représente un rond ou un carré et le lien qui peut les associer.

Ainsi, l'utilisation d'un méta-modèle s'impose progressivement dans un objectif de « cadrer », de définir les langages.

Un méta-modèle est défini par l'OMG comme étant [OMG, 2002] :

« un modèle qui définit le langage utilisé pour exprimer un modèle ».

Il s'agit donc d'un modèle de langage de modélisation. Plus pragmatiquement, un méta-modèle est « une description de tous les concepts d'un langage. La méta-modélisation permet de définir directement les concepts manipulés, leur sémantique et la syntaxe du langage liée à l'utilisation de ces concepts » [Chapron, 2006].

Alors que la relation qui unit le système étudié à son modèle pourrait être exprimée par « **est représenté par** », la relation qui unit un modèle à son méta-modèle pourrait, elle, être exprimée par « **est conforme à** » [Bézivin *et al.*, 2004].

Notons également que [Bataille *et al.*, 2001] ajoute à la définition précédente de « méta-modèle » la définition suivante : « un méta-modèle est un modèle dont les instances sont des modèles ». Ces deux définitions ne positionnent pas le méta-modèle de la même manière. La première associe un méta-modèle au langage de représentation de modèles, alors que la seconde le présente comme le fruit d'une opération d'abstraction (l'opération « inverse » de l'instanciation), ce qui semble plus adapté à la « technologie objet ». Nous faisons donc le choix d'adopter la première définition comme référence pour la suite, bien que la seconde soit tout autant valable.

Pour qu'un utilisateur puisse comprendre sans ambiguïté et manipuler un modèle, il convient donc de préciser le méta-modèle associé. Dans le cas de la Figure I-3, le méta-modèle peut se matérialiser par une légende qui devrait préciser la sémantique des objets utilisés (comment se représente un département ?) et la syntaxe associée. Il s'agit donc de la légende des cartes.

La carte devra être **conforme à** sa légende, qui est donc son méta-modèle.

1.3. Méta-méta-modèle

Ce principe d'abstraction peut être reproduit une nouvelle fois, en définissant un méta-méta-modèle. La relation qui lie un modèle à son méta-modèle est la même que celle qui lie un méta-modèle à son méta-méta-modèle (« **est conforme à** »).

L'OMG propose donc la définition suivante [OMG, 2002] :

un méta-méta-modèle est « un modèle qui définit le langage utilisé pour exprimer un méta-modèle ».

Dans le cadre de l'approche MDA¹ (Model Driven Architecture), qui est une des approches préconisées par l'IDM, l'OMG a défini un méta-méta-modèle appelé le MOF (Meta Object Facility). Synthétiquement, le MOF est une spécification qui définit plus d'une vingtaine de concepts destinés à être utilisés pour définir des méta-modèles. Ainsi, il introduit, par exemple, les notions de [Marvie, 2003] :

- classe : « une classe permet de définir un concept au sein d'un méta-modèle. Une classe peut contenir des attributs et des opérations »,
- association : « une association permet de définir une relation entre deux classes »,

¹ Nous détaillons les concepts de l'approche MDA par la suite (Cf. Chapitre I.2.3.1).

- package : « un package permet de regrouper des classes et associations étant liées » ...

Le MOF est réflexif : il se décrit lui-même (il n'y a donc pas besoin d'un méta-méta-méta-modèle).

L'OMG représente cette architecture sous la forme d'une pyramide à quatre niveaux (Figure I-5).

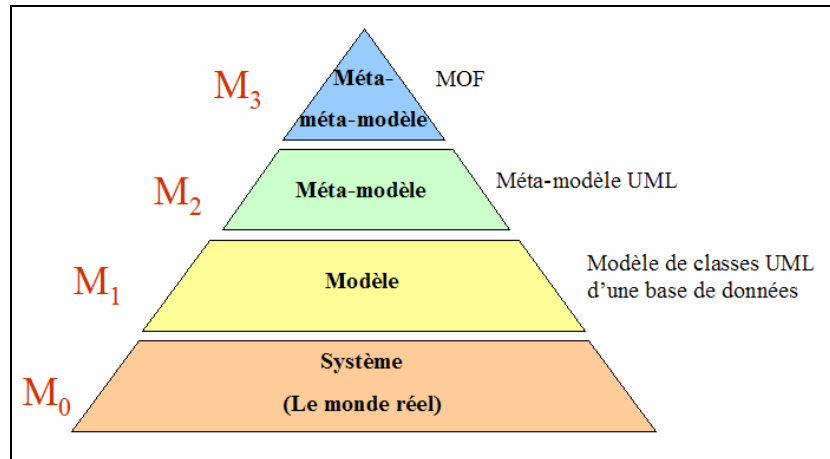


Figure I-5 : architecture à quatre niveaux (inspirée de [OMG, 2002])

Un diagramme de classes en langage UML (Unified Modeling Language, [Muller, 1997]) d'une base de données appartient donc au niveau M_1 . Le langage UML, qui permet donc, entre autres, de modéliser des bases de données, est un méta-modèle et appartient par conséquent au niveau M_2 . Le modèle de classe du niveau M_1 sera conforme au méta-modèle UML. De plus, le méta-modèle UML est conforme à la spécification MOF.

Notons enfin qu'UML n'est pas le seul méta-modèle conforme au MOF. Citons, par exemple, le SPEM (Software Process Engineering Metamodel, littéralement « méta-modèle d'ingénierie des processus logiciels »), qui permet de décrire le processus de production de logiciels [OMG, 2005a].

Ainsi, certains langages ont une ambition de couverture indépendante des domaines d'application (comme UML) : ils se veulent universels. D'autres sont spécifiques à un domaine précis (comme le SPEM). Ils sont désignés par l'acronyme DSL : « Domain Specific Language ». Le passage de l'un à l'autre est possible par des artifices, comme la notion de profil UML, qui permet de spécialiser UML à un domaine particulier.

Dans la vision de l'OMG, « tout est modèle » [Bézivin, 2004]. Ainsi, par la suite, lorsque nous utiliserons le terme « modèle », il pourra indifféremment désigner un modèle de système (niveau M_1), un méta-modèle (niveau M_2), ou un méta-méta-modèle (niveau M_3).

L'utilisation de cette structure à quatre niveaux fait émerger le besoin d'avoir des réponses aux deux questions suivantes :

- si deux modèles représentent des systèmes différents mais « comparables », selon les mêmes méta-modèles ou non, comment est-il possible de les comparer ?
- si deux modèles, situés au même niveau, représentent le même système tout en étant conformes à des méta-modèles différents, est-il possible de passer de l'un à l'autre ?

Nous tenterons d'apporter des réponses à ces questions dans les parties suivantes.

2. Opérations sur les modèles

Ces « mécanismes » de passage d'un modèle à un autre, de modification, de comparaison, *etc.*, s'apparentent à des opérations mathématiques. A ce titre, un modèle peut donc être considéré comme un simple opérande d'une opération. Certains auteurs utilisent le terme « **morphisme** » [D'Antonio, 2005] pour désigner l'ensemble de ces opérations.

2.1. Notion de morphisme

Le terme de « morphisme » a été emprunté à un domaine des mathématiques : l'algèbre. Il désigne alors une application entre deux ensembles. La définition précise que l'image d'un ensemble de départ par un morphisme est un ensemble d'arrivée ayant la **même structure**, c'est-à-dire les mêmes propriétés algébriques¹ [Lipschutz, 1973].

L'analogie entre le monde de l'IDM et les mathématiques va consister à paralléliser les notions de « structure algébrique » et de « modèle ». Le terme « morphisme » a donc été annexé par l'IDM pour finalement englober l'ensemble des fonctions qui permettent de transformer un modèle en un autre **et** l'ensemble des relations qui permettent de tracer les correspondances entre deux modèles. Le caractère « conservateur de structure » des morphismes se justifie en précisant que l'image d'un modèle par un morphisme reste un modèle ayant un certain degré d'analogie avec son antécédent.

D'une manière générale, nous représenterons un morphisme M entre un modèle A et un modèle B par le formalisme suivant (Figure I-6) :

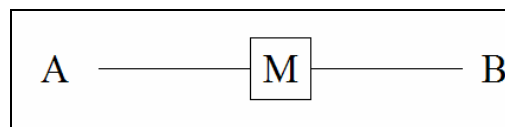


Figure I-6 : formalisme habituellement utilisé pour représenter les morphismes

Ainsi, sur l'exemple de la Figure I-3, un morphisme entre le modèle de la France « administrative » et celui de la France « fluviale » pourrait consister à mettre en relation les fleuves et les départements (par exemple, identifier les correspondances au niveau des noms des départements et des fleuves, ou encore les correspondances entre les courbes des fleuves et les frontières départementales...).

Le type de morphisme décrit par cet exemple est appelé « non-altérant », car il laisse les modèles « intacts ». Il s'agit, en quelque sorte, simplement d'une « liste » d'équivalences : les deux cartes n'ont pas été modifiées. Il existe également des morphismes « altérants ». Nous nous intéresserons à leur cas plus tard. La partie suivante présente la famille des morphismes « non-altérants ».

¹ Par exemple, l'image d'un anneau (au sens algébrique) par un morphisme est un anneau !

2.2. Les morphismes « non altérants » : les mappings

2.2.1. Définition

Les morphismes non-altérants sont une mise en relation d'éléments de modèles, sans altération de ces derniers. La bibliographie les nomme également « mappings ».

D'un point de vue théorique, une définition du terme mapping est proposée dans [Bottoni *et al.*, 2006]. Nous avons légèrement simplifié cette définition afin de la rendre plus « accessible ».

Soit deux modèles A et B , un mapping est une relation m , telle que $m \subseteq Sub(A) \times Sub(B)$, où $Sub(X)$ désigne un sous-ensemble du modèle X .

Le produit cartésien $Sub(A) \times Sub(B)$ est l'ensemble de tous les couples $(X ; Y)$, dont la première composante appartient à $Sub(A)$ et la seconde à $Sub(B)$. La définition précédente précise donc qu'un mapping est un de ces couples $(X ; Y)$.

Le mapping entre l'**ensemble** A , qui est composé de l'ensemble des noms de fleuves français, et l'**ensemble** B , qui est composé de l'ensemble des noms de départements français, nous permet d'illustrer ceci. Ainsi, dans cet exemple, le produit cartésien $A \times B$ décrit l'ensemble des couples (nom de fleuve ; nom de département). Un mapping est donc par exemple le couple (Rhône ; Rhône – 69), mais peut également être le couple (Rhône ; Réunion – 974).

Il apparaît donc le besoin de « noter » la pertinence des couples engendrés par le produit cartésien. C'est pourquoi [D'Antonio, 2005] introduit la notion de **mapping annoté**. Il s'agit de rajouter une composante à la relation m , qui permet d'évaluer sa pertinence (selon des critères à définir). La relation m s'exprime alors de la manière suivante : $m \subseteq Sub(A) \times Sub(B) \times L$, avec $L \in [0,1]$ par exemple.

Ainsi, dans l'exemple que nous avons utilisé précédemment, l'annotation va consister à légitimer le fait qu'un fleuve traverse un département, ce qui va correspondre à la valeur « 1 » lorsque cela est avéré, et à la valeur « 0 » lorsque cela est faux. Le second couple (Rhône ; Réunion – 974) est inadapté, il pourrait être noté par la valeur 0. Nous aurions donc les deux triplets : (Rhône ; Rhône – 69 ; 1) et (Rhône ; Réunion – 974 ; 0).

Par souci de simplification, le terme mapping sera employé par la suite pour désigner indifféremment un couple établi par le produit cartésien de deux modèles ou un sous-ensemble de ce produit cartésien. La notation utilisée ne différencie donc pas une correspondance (un couple) de l'ensemble des correspondances. Notons que certains auteurs désignent une correspondance par le terme « matching » et l'ensemble des correspondances par le terme « mapping ».

2.2.2. Mappings verticaux et mappings horizontaux

En tant qu'ensemble de correspondances entre deux modèles, la bibliographie classe les mappings selon les deux sens envisageables :

- le sens vertical, c'est-à-dire entre un modèle et son méta-modèle (entre les niveaux M_1 et M_2), ou un méta-modèle et son méta-méta-modèle (niveaux M_2 et M_3),
- le sens horizontal, c'est-à-dire entre deux modèles de même niveau (entre deux modèles du niveau M_1 ou deux modèles du niveau M_2 ...).

Les mappings verticaux correspondent donc aux relations générées lors d'une instanciation (du niveau M_N vers le niveau M_{N-1}) ou d'une abstraction (du niveau M_N vers le niveau M_{N+1}) (Figure I-7).

Par exemple, le concept de « département » appartient au méta-modèle de la carte administrative de la France. Le département « Ardèche - 07 » (tout comme les quatre-vingt-dix-huit autres départements) peut y être rattaché, dans le cadre d'un mapping vertical.

Les mappings horizontaux permettent, eux, d'identifier les correspondances entre différents modèles de même niveau dans l'architecture de l'OMG (Figure I-7). L'exemple du mapping entre les noms des fleuves français et les noms des départements français est donc un mapping horizontal. Ces mappings sont l'objet de notre étude et nous détaillerons leurs caractéristiques dans les parties suivantes.

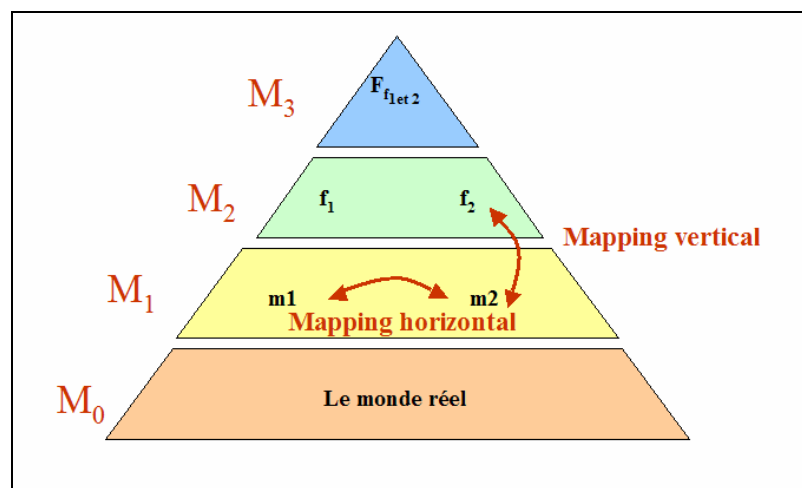


Figure I-7 : mapping vertical et mapping horizontal

Par la suite, quand nous utiliserons le terme « mapping », il faudra donc comprendre par convention « mapping horizontal ».

2.2.3. Mappings endogènes / exogènes

La définition de mapping, en tant qu'un ensemble de mises en relation d'éléments de modèles, ne précise pas si les modèles sont exprimés dans les mêmes langages ou dans des langages différents.

Les deux cas sont envisageables. Si les langages sont les mêmes, il s'agira d'un mapping endogène. Dans le cas contraire, il s'agira d'un mapping exogène [Caplat *et al.*, 2003].

La notation utilisée pour exprimer le fait qu'un modèle « m », d'un système « s », est conforme au méta-modèle f sera : $m(s)/f$.

La Figure I-8 propose une représentation schématique de la notion de mapping endogène.

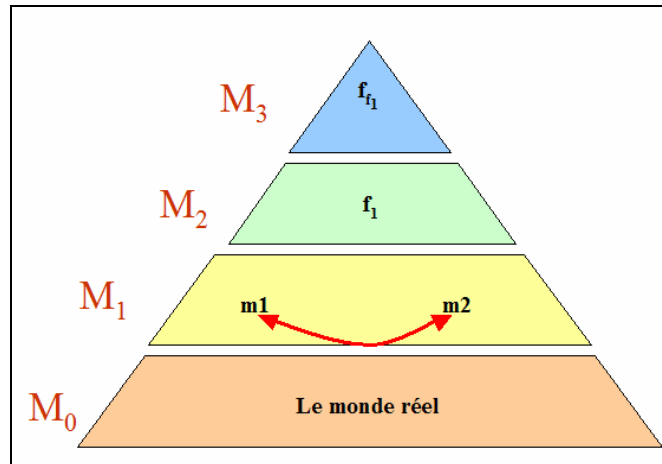


Figure I-8 : mapping endogène : $m_1(s)/f_1$ et $m_2(s)/f_1$

Les deux modèles m_1 et m_2 (niveau M_1) sont conformes au même méta-modèle f_1 . La problématique de réalisation du mapping entre m_1 et m_2 va consister à établir des relations directement entre les éléments de m_1 et les éléments de m_2 . Nous parlerons de « mapping de modèles ».

Un mapping endogène consisterait, par exemple, à mettre en relation des villes jumelées, entre deux pays représentés de « la même manière » : c'est-à-dire, dont les représentations sous forme de « carte géographique » **sont conformes** à la même légende. Entre la carte de l'Italie et celle de la France, une des relations du mapping sera le couple « Rome ; Paris ».

Dans le cas où les méta-modèles des deux modèles sont différents, une représentation, en reprenant le principe de l'architecture à quatre niveaux de l'OMG, pourrait être la Figure I-9.

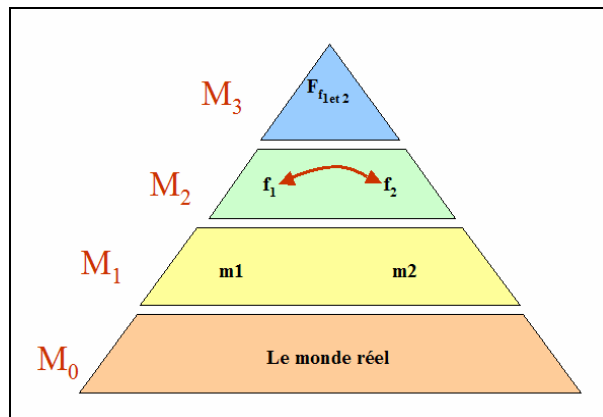


Figure I-9 : mapping exogène : $m_1(s)/f_1$ et $m_2(s)/f_2$

Le modèle m_1 est conforme au méta-modèle f_1 et le modèle m_2 est conforme au méta-modèle f_2 .

Dans un premier temps, le mapping entre les deux méta-modèles va permettre d'établir des « règles de mise en relation », c'est-à-dire qu'il va apporter des réponses à la question : « quel ensemble d'éléments du méta-modèle source est-il possible de mettre en relation à quel ensemble d'éléments du méta-modèle cible ? ». Dans un second temps, le mapping entre les deux modèles va donc consister à générer les liens entre les composants dont la « comparabilité » a été établie par les règles de mise en relation (Cf. Figure I-10).

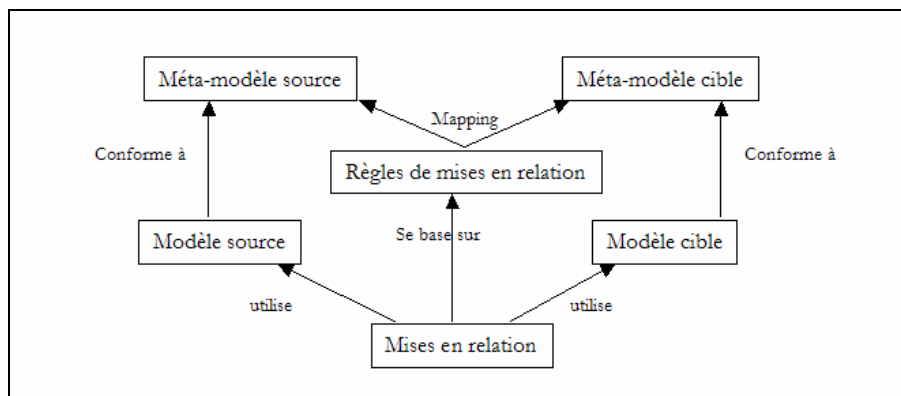


Figure I-10 : règles de mises en relation

Ces mappings vont donc permettre d'identifier les éléments « comparables » (ce qu'il est possible de mettre en relation) au niveau M_N et ensuite de générer les correspondances au niveau M_{N-1} .

[Caplat *et al.*, 2003] proposent un diagramme de classes en UML des relations entre mappings et formalismes. Dans ce schéma, la notion de « formalisme » est à rapprocher de notre notion de « méta-modèle ». Cette vision est intéressante, car elle met en exergue le fait qu'un mapping peut être modélisé en s'appuyant sur un formalisme (Figure I-11).

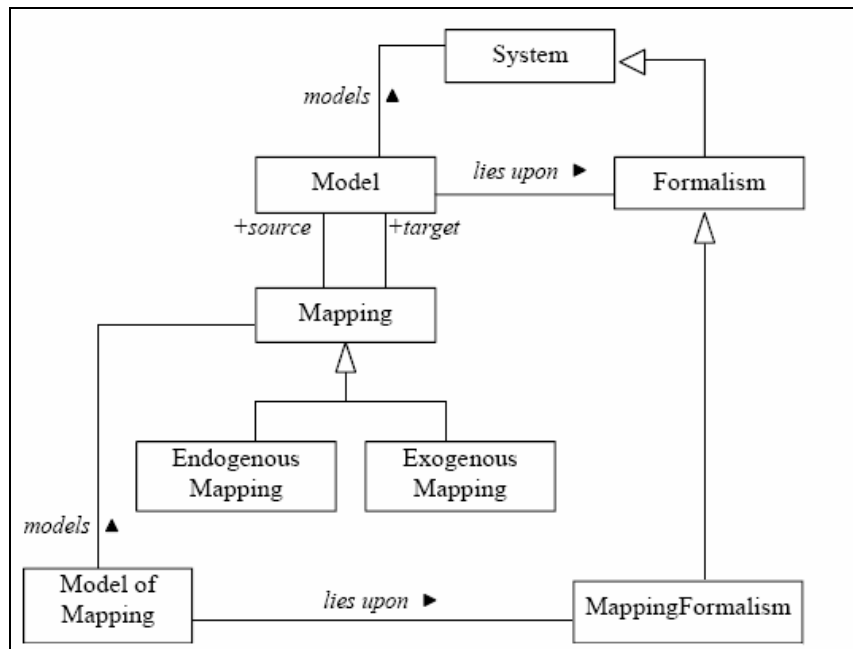


Figure I-11 : relations entre mapping et formalisme [Caplat *et al.*, 2003]

Ainsi, un modèle représente (relation « models » sur la Figure I-11) un système et est lié à un formalisme (méta-modèle). Un mapping consiste à mettre en relation deux modèles : un modèle cible (relation « target » sur la Figure I-11) et un modèle source (relation « source »).

Un mapping est modélisé par un modèle de mapping (« Model of Mapping »), qui est lui-même lié à un formalisme. Ce formalisme de mapping (« MappingFormalism ») hérite de la notion générale de formalisme. Le mapping est spécialisé par les deux notions de « mapping endogène » et « mapping exogène ». Il y a des équivalences entre les concepts de la Figure I-10 et ceux de la Figure I-11. Ainsi, les « Règles de mise en relation » sur la Figure I-10 correspondent au « Model of mapping » de la Figure I-11. De même, les « Mises en relation » correspondent au concept de « Mapping ».

La partie suivante traite des notions qui sont classiques dans la représentation des relations au sein des modèles de données : il s'agit de la cardinalité des relations et de la direction de celles-ci.

2.2.4. Cardinalités et directions des relations

Nous avons établi qu'un mapping est un sous-ensemble d'un produit cartésien. De part la définition même du produit cartésien, il est envisageable que les relations établies par le mapping soient « évoluées », c'est-à-dire qu'un élément du modèle A soit en relation, par exemple, avec un ou plusieurs élément(s) du modèle B.

Cela nous conduit à différencier quatre types de mappings de modèles, en fonction des cardinalités des relations mises en jeu (Figure I-12 – dans les exemples que nous fournissons, le modèle source est la « France fluviale » et le modèle cible est la « France administrative ») :

- les mappings de type « 1 : 1 » (type 1) utilisent des relations qui associent un élément d'un modèle source à un et un seul élément d'un modèle cible. Par exemple, à un fleuve ne peut être associé qu'un seul département qui porte « exactement le même nom »,
- les mappings de type « 1 : n » (type 2) qui peuvent associer un élément du modèle source à plusieurs éléments du modèle cible. Par exemple, plusieurs départements ont un nom qui inclut le nom d'un même fleuve : les noms des départements « Tarn - 81 » et « Tarn et Garonne - 82 » sont tous les deux composés du nom du fleuve « Tarn »,
- les mappings de type « m : 1 » (type 3) peuvent associer plusieurs éléments du modèle source à un seul élément du modèle cible. Par exemple, le département « Saône et Loire - 71 » peut être associé aux fleuves « Saône » et « Loire »,
- les mappings de type « m : n » (type 4) associent plusieurs éléments du modèle source à plusieurs éléments du modèle cible.

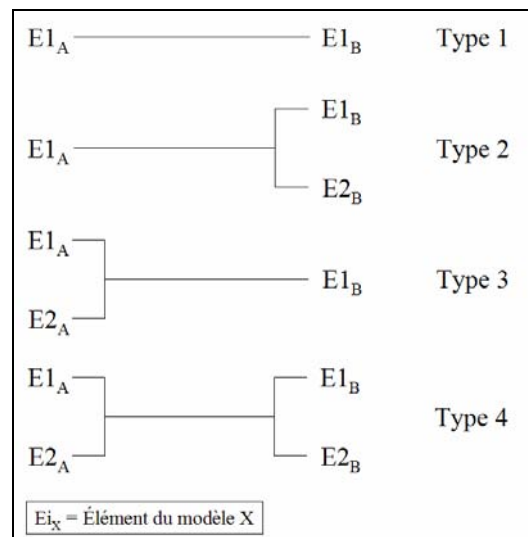


Figure I-12 : illustration de la notion de cardinalité des mappings (avec $m=n=2$) (inspirée de [Lopes *et al.*, 2006])

Il est également possible de définir une notion de « direction » : un mapping peut être unidirectionnel (les relations vont de A vers B) ou bidirectionnel (les relations vont de A vers B et de B vers A). Nous illustrons ceci avec la Figure I-13.

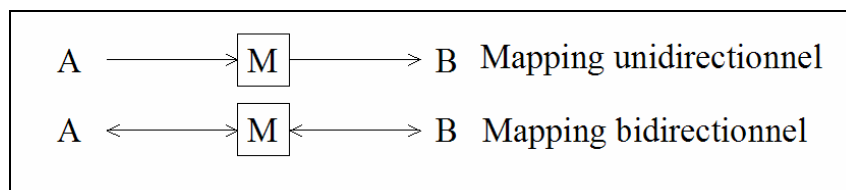


Figure I-13 : mapping unidirectionnel ou bidirectionnel

Le besoin qui est à l'origine de la réalisation d'un mapping justifie le caractère unidirectionnel ou bidirectionnel de celui-ci. En fonction de ce besoin, la sémantique portée par la mise en relation sera établie. Si cette sémantique est « réversible » alors le mapping sera bidirectionnel.

Un exemple de mapping bidirectionnel entre les noms des départements et les noms de fleuves est l'ensemble des relations qui portent la sémantique : « un fleuve X a le même nom qu'un département Y » (puisque c'est équivalent à : « un département Y a le même nom qu'un fleuve X »).

2.2.5. Synthèse et conclusion sur les mappings

Nous proposons dans cette partie de résumer toutes les notions abordées dans les parties précédentes sur un diagramme de classes UML. Nous assimilons le mapping à une classe qui peut être spécialisée en deux classes différentes : « Mapping vertical » et « Mapping horizontal ». La classe « Mapping horizontal » possède trois attributs

Nous avons repris les liens entre les notions de modèles et de mapping proposés par la Figure I-11.

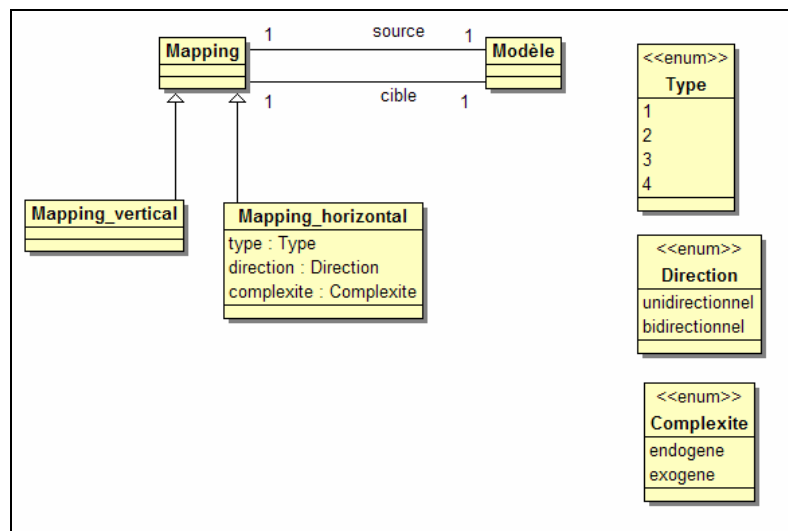


Figure I-14 : diagramme de classes en UML de la notion de mapping

Enfin, certains auteurs proposent une classification des mappings en fonction des approches employées pour les construire (annexe). Notons qu'elle est particulièrement applicable à l'automatisation des mappings d'ontologies [Kalfoglou *et al.*, 2003]. Nous ne présentons pas en détail cette caractéristique, car l'application que nous envisageons ne se situe pas dans cette catégorie. Les mappings peuvent être construits selon des approches sémantiques [Giunchiglia *et al.*, 2003], syntaxiques ou linguistiques (en fonction de la fréquence d'utilisation de certains mots clés).

Dans la partie suivante, nous nous intéresserons aux morphismes altérants, également désignés par le terme transformation de modèles.

2.3. Les morphismes « altérants » : transformation de modèles

La « transformation de modèles » a déjà fait l'objet d'une littérature conséquente [OMG, 2003], [D'Antonio, 2005], [Akehust *et al.*, 2002], [Kleppe *et al.*, 2003]. Il s'agit d'un concept clé de l'approche MDA.

Afin de rendre cette partie plus accessible, nous commençons par présenter l'approche MDA, ce qui nous permettra ensuite de nous intéresser à la transformation de modèles en tant qu'opération mathématique.

2.3.1. Illustration : l'approche MDA

Le MDA est une approche proposée par l'OMG en 2000 [OMG, 2003]. Les objectifs sont multiples et ambitieux : générer le code des logiciels automatiquement, pouvoir passer facilement, de **façon automatisée**, d'un modèle à un autre, d'un niveau d'abstraction à un autre ou d'une application donnée à une autre. L'approche est basée sur plusieurs normes techniques déjà établies (UML, MOF, et surtout des langages de transformation de modèle comme Query View Transformation (QVT)). C'est une spécification de langage de transformation de modèle, qui permet d'adapter un modèle à de nouvelles contraintes, de transformer n'importe quel méta-modèle dédié vers un autre [OMG, 2005b].

L'approche MDA se base sur l'utilisation de quatre modèles principaux :

- **les modèles « Computation Independent Model » (CIM)** : ils montrent le système dans l'environnement organisationnel dans lequel il va être exécuté. Il aide à la compréhension du problème,
- **les modèles « Platform Independent Model » (PIM)** : ce sont des modèles qui n'ont pas de dépendance avec les plateformes techniques. Les PIM représentent par exemple les différentes entités fonctionnelles d'un système avec leurs interactions.
- **les modèles « Platform Description Model » (PDM)** : ils décrivent la plate-forme sur laquelle le système va être exécuté,
- **les modèles « Platform Specific Model » (PSM)** : Ce sont des modèles de l'application qui sont dépendants de la plate-forme choisie.

Le passage du PIM vers le PSM se fait grâce à une transformation de modèles (« Processus de transformation ») sur la Figure I-15. Le PIM est le « Modèle source », qui est conforme au « Méta-modèle source ». Le PSM est le « Modèle cible », conforme au « Méta-modèle cible ». D'autres transformations sont également possibles : de PIM vers PSM, de PIM vers PIM, de PSM vers PSM...

Notons que la transformation s'appuie sur des « Règles de transformation ». La mise en œuvre de ces règles est réalisée par le « processus de transformation ». Cette opération n'est en rien automatique : tout le processus est à spécifier et à outiller avant d'être exécuté.

Nous reviendrons sur ces règles de transformation par la suite.

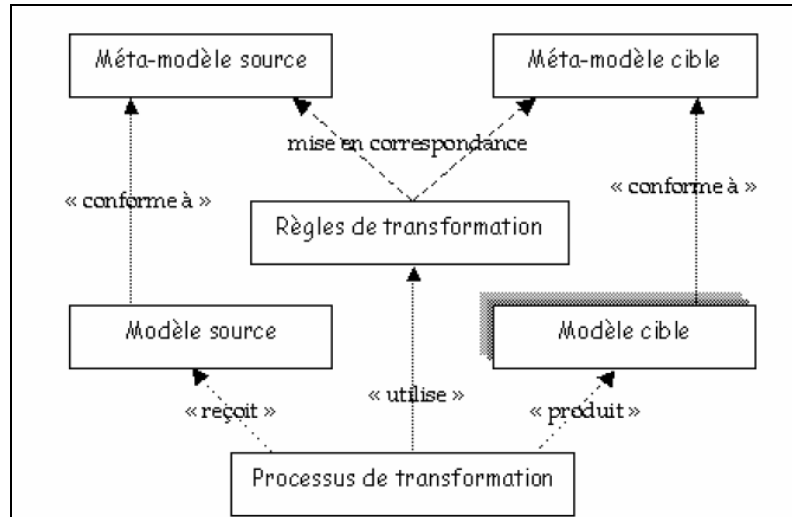


Figure I-15 : le processus de transformation de modèles [Touzi, 2007]

2.3.2. Définition formelle

D'un point de vue formel, la transformation a été définie de la manière suivante [D'Antonio, 2005] :

soit A un modèle et $t : \text{MOD} \rightarrow \text{MOD}$ une fonction (MOD désigne l'ensemble des modèles), t est alors une fonction de transformation de modèle. $B = t(A)$ est alors le modèle A transformé par t (mathématiquement, B est l'image de A par la fonction t).

Plus pragmatiquement, [Jouault, 2006] définit une transformation de modèles comme « une opération qui crée automatiquement un ensemble de modèles cible à partir d'un ensemble de modèles source ».

2.3.3. Développement

L'opération consiste à transformer un modèle source A d'un système s , conforme à un méta-modèle f_1 en un modèle cible B , conforme à un méta-modèle « cible » f_2 du même système s . La notation est la suivante (utilisée dans [Caplat et al, 2003]) : $A(s) / f_1 \rightarrow B(s) / f_2$.

Il s'agit d'un morphisme « altérant » car le modèle source (A) va être transformé en un modèle cible (B). Une transformation est donc une **fonction**, alors qu'un **mapping** est une relation plus large [D'Antonio, 2005].

La transformation d'un modèle en un autre est possible grâce à la définition de « règles de transformation » [Panetto, 2006]. Dans [Levendovski *et al.*, 2002], cet ensemble de règles correspond à un « mapping » (« ensemble des règles de transformation de modèles permettant de traduire les instances d'un méta-modèle source en des instances d'un méta-modèle cible »). Selon les informations présentées précédemment, ces règles sont obtenues grâce à un mapping exogène (Figure I-16).

De façon pragmatique, le mapping des méta-modèles fixe les correspondances entre les concepts instanciés dans les modèles. Les règles de transformation sont donc bien les relations qui composent le mapping. Par exemple, le mapping de méta-modèles va permettre d'établir qu'un concept A appartenant à un méta-modèle source M_A est équivalent à un concept B d'un méta-modèle cible M_B . La transformation d'un élément du modèle source m_A conforme au concept A sera donc un élément du modèle cible m_B , conforme au concept B.

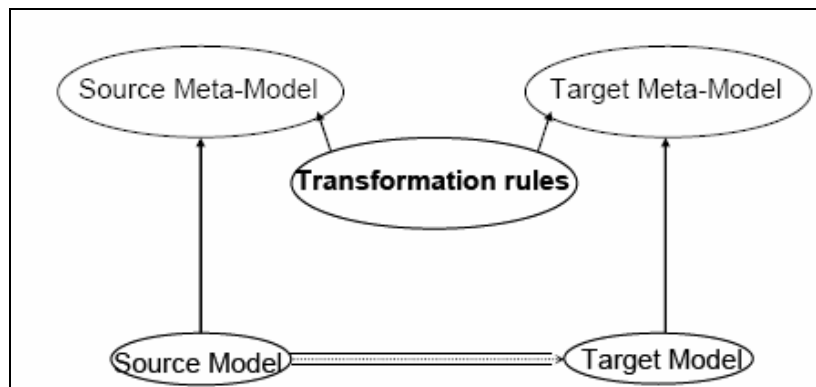


Figure I-16 : transformation de modèles [Lemesle, 1998]

3. Synthèse

Dans ce premier chapitre, nous avons donc :

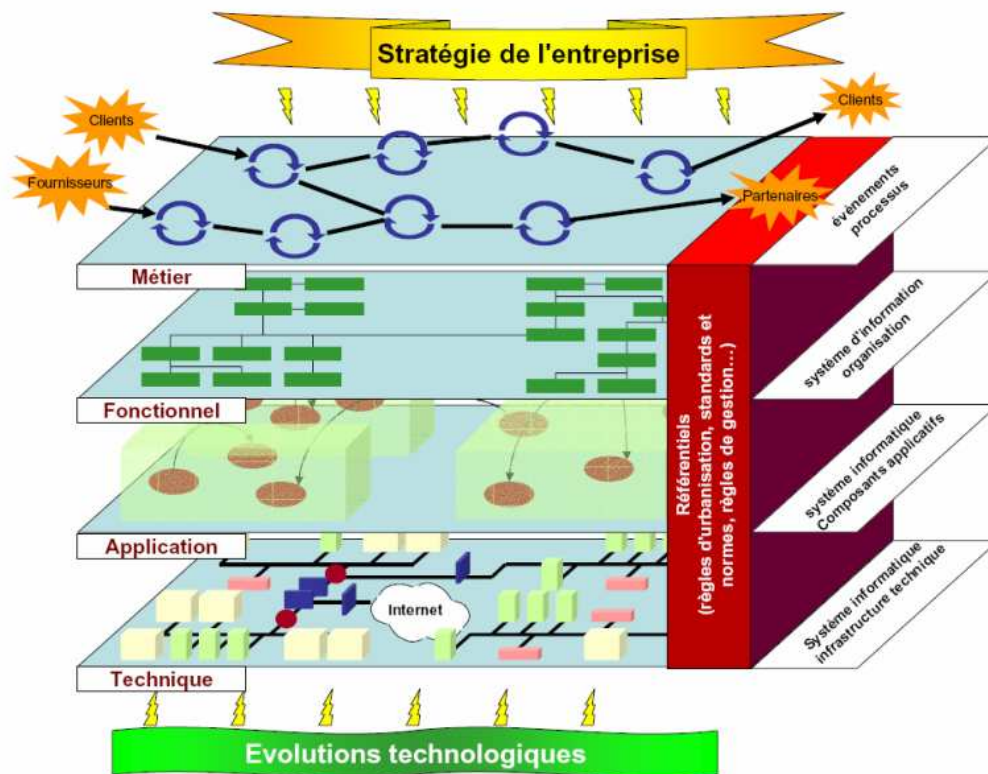
- défini les notions clés de modèle, méta-modèle, méta-méta-modèle,
- abordé les définitions mathématiques de morphisme, et présenté l'analogie avec les morphismes de modèles de l'IDM,
- présenté la famille des morphismes non-altérants et celle des morphismes altérants.

Ce dernier point nous a permis de mettre en valeur le fait que mapping de modèles et transformation de modèles sont deux opérations qui n'ont pas les mêmes objectifs, ni les mêmes pré-requis. Ainsi, pour réaliser un mapping, il faut posséder deux modèles et connaître l'objectif de la mise en relation (ce qui doit être mis en valeur par la construction des correspondances). Pour réaliser une transformation de modèles, il faut posséder un modèle de départ et des règles de transformation, qui sont obtenus grâce à un mapping des méta-modèles source et cible. Le mapping peut donc être vu comme la première étape de la transformation.

Notre objectif de thèse étant de réaliser un mapping entre deux modèles clairement identifiés, ce chapitre était nécessaire afin de poser la « brique théorique » comme fondement de nos travaux. Les deux prochains chapitres auront pour objet de présenter successivement le modèle cible et le modèle source de notre mapping.

Chapitre II. LES REFERENTIELS DE BONNES PRATIQUES INFORMATIQUES

Les liens entre les processus métiers d'une organisation et son système d'information sont aujourd'hui établis. La modélisation des premiers est souvent considérée comme un préalable indispensable à la conception du second [Berthier *et al.*, 2005]. Les exigences¹ métiers seront la base des spécifications du système d'information, qui supportera ensuite l'exécution des processus, tant au niveau fonctionnel, que technique (Figure II-1).



Source : Cigref

Figure II-1 : les différentes « couches » du système d'information

En tant que « nerf central », un système d'information est amené à constamment évoluer, au gré des changements stratégiques, organisationnels, opérationnels, *etc.*, afin d'être toujours apte à lier système de pilotage et système opérant.

¹ Une définition globale du terme « exigence » sera fournie dans le Chapitre III. Nous nous contenterons de préciser pour l'instant qu'une exigence est « ce qui est exigé ».

Ces changements se traduisent par la mise en place, le lancement et la réalisation de multiples projets de systèmes d'information. Ces projets ne sont cependant pas comparables aux activités de production plus « classiques » (BTP...). Les spécificités du monde des systèmes d'information expliquent cette différence :

- un système d'information est composé, au moins en partie, de produits immatériels. Il est donc plus complexe de suivre les réalisations techniques, les avancements et d'assurer la conformité au besoin initial,
- le « turn-over » de la main-d'œuvre est élevé (spécialement dans les sociétés de services), ce qui ne facilite pas les projets « longs »,
- les rôles et responsabilités de la maîtrise d'ouvrage sont souvent sous-estimés, et cette dernière est parfois peu qualifiée. Les activités, comme la conduite du changement, sont donc rarement considérées comme « critiques », alors qu'elles jouent un rôle important dans la réussite des projets,
- les technologies sont multiples et de durée de vie très limitée. Le retour d'expérience sur une technologie est donc relativement faible. Il en résulte une réelle difficulté au niveau de l'estimation des charges par exemple car l'industrie informatique est en perpétuelle « jeunesse »,
- il y a une corrélation non-linéaire entre la variation du nombre de ressources affectées à une tâche et la durée de cette dernière,
- un logiciel est un produit complexe, tant au niveau de ses fonctionnalités propres qu'au niveau de ses interfaces. Un projet est donc rarement l'affaire d'une seule organisation, totalement indépendante. L'externalisation et la sous-traitance sont monnaie courante, mais constituent également un facteur de risque pour les projets (complexité liée à la communication entre parties prenantes, au suivi de projets, aux intérêts commerciaux...),
- l'exploitation d'un système d'information est un domaine bien particulier : les contraintes de sécurité, la gestion des environnements (développement, recette, production...), la supervision des services, la conception et la gestion des architectures des infrastructures... sont des domaines d'ingénierie **spécifiques** qui demandent une large palette de compétences techniques.

Ces facteurs justifient, en partie, le fait que le taux de réussite¹ des projets menés par l'industrie informatique soit relativement faible (environ 34 % en 2004 [Bénard *et al.*, 2005]). Bien que ce taux évolue favorablement (il était de 16 % en 1994 ! [The Standish Group, 1994]), il reste très bas. Malgré la progression des méthodes, le développement de nouveaux outils, des investissements divers et variés sur la qualité... seul un projet sur trois est une réussite.

Certains travaux démontrent que le problème **fondamental** de l'industrie informatique réside dans son incapacité à gérer efficacement le processus logiciel [Niazi *et al.*, 2005]. Dans ce contexte, différentes méthodes d'ingénierie ont été développées afin de rendre plus efficace la gestion du processus logiciel et donc, la performance même de ce dernier. Parmi elles, une famille est particulièrement répandue : les approches SPI (Software Process Improvement, littéralement « amélioration du processus logiciel »). Cette famille inclut les approches qualité liées aux notions de maturité et de capacité. Il s'agit par exemple des fameux « modèles de maturité » et plus généralement des « référentiels de bonnes pratiques informatiques ».

L'introduction a été l'occasion de montrer la concentration et la diversité du monde des référentiels de bonnes pratiques. Nous avons également montré que les référentiels propres au contexte qualité de notre cas applicatif sont clairement identifiés. Rappelons qu'il s'agit du CMMI (pour la conception et le développement des applications informatiques), de l'ISO 9000 : 2000, en ce qui concerne le système de management de la qualité, de l'ISO 20000 (et donc de « Information Technology Infrastructure Library » : ITIL, pour l'exploitation technique de ces applications), et enfin de l'ISO 27000 (pour le système de management de la sécurité). La présentation de ces référentiels est un des objets de ce chapitre (parties 2 à 4). Cependant, il est nécessaire de commencer par définir quelques notions clés : qualité, processus, maturité, capacité... (partie 1). Enfin, la dernière partie (partie 5) nous permettra de proposer un « modèle conceptuel général des référentiels de bonnes pratiques informatiques », en insistant particulièrement sur la notion de « bonne pratique informatique ».

¹ Les projets informatiques considérés par le Standish Group sont des projets de développement. Dans la suite de ce rapport, le terme « projet » désigne des projets de Système d'Information, c'est-à-dire des projets de développement, de déploiement, de paramétrage...

Un projet est considéré comme une réussite s'il s'est fini dans les temps, en respectant le budget prévu et s'il a permis d'aboutir à tous les composants et fonctions prévus initialement.

1. Notions clés

1.1. Qualité et roue de Deming

Dans l'industrie contemporaine, la qualité se doit d'être une préoccupation majeure, commune à tous les intervenants d'un projet. Et c'est effectivement de plus en plus le cas. Une étude¹ réalisée par l'organisme international de normalisation en témoigne : en quatre ans (entre 2002 et 2006), le nombre d'organisations certifiées ISO 9001² a été multiplié par plus de cinq (de 167 123 en 2002, il est passé à 897 886 en 2006). Sachant que cette norme a vu le jour en 1987, cela permet d'illustrer l'effort fait par les organisations en matière de qualité.

La norme ISO 8402 (« Management de la qualité et assurance de la qualité – Vocabulaire ») propose une définition générale intéressante pour le terme « qualité », car elle est applicable à tous les secteurs d'activités et pour tous les types d'organisation. La qualité est ainsi définie comme [ISO, 1994] :

« l'ensemble des caractéristiques d'une entité qui lui confèrent l'aptitude à satisfaire les besoins exprimés et implicites ».

La qualité vise donc à satisfaire pleinement les besoins d'un client ou plus généralement d'un donneur d'ordres. Précisons que ce dernier est placé au centre de toute démarche qualité, puisque ce sont ses besoins qui justifient une telle démarche [Vauquier, 1997].

Dans un objectif de recherche de « qualité totale », l'amélioration au sein d'une organisation se doit d'être **permanente**, **omniprésente** et **structurée** afin d'améliorer de façon continue la rentabilité et l'efficacité auprès des clients [Massot, 1999]. En conséquence, dans une organisation, le besoin en amélioration perd son caractère réactif pour devenir proactif. Il ne s'agit donc plus de réagir à une situation devenue intolérable, mais bel et bien d'éviter la dérive de cette situation et de pouvoir profiter en permanence d'un niveau d'efficacité optimal.

Ainsi, lorsqu'une organisation décide de s'engager dans une démarche qualité, quelles qu'en soient les raisons et les motivations, elle s'engage avant tout dans une **démarche d'amélioration continue** [Froman *et al.*, 2003]. Cette démarche est symbolisée par la roue de Deming³. Ce qualiticien a modélisé l'amélioration de la qualité dans une organisation à travers un cycle

¹ URL : <http://www.iso.org/iso/fr/survey2006.pdf>.

² La norme ISO 9001 sera largement présentée par la suite. A ce stade du rapport, nous précisons simplement que la norme ISO 9001 est la partie « normative » de l'ISO 9000 : 2000, qui est généralement reconnue comme étant la référence en matière de certification de système de management de la qualité.

³ W. Edward Deming (1900-1993), statisticien américain et fondateur du mouvement de la qualité, est à l'origine de l'introduction de la philosophie de la qualité dans l'industrie japonaise d'après-guerre. Le constat assez simple sur lequel s'est appuyé Deming est ainsi présenté par Carol Kennedy : « les processus sont sensibles à des variations induisant des pertes de qualité, si l'on parvient à gérer ces degrés de variation, il devient possible d'en réduire les effets et d'augmenter globalement la qualité ». <http://www.cndp.fr/revueEcoManagement/noteslecture/2005-03.htm>.

consistant à enchaîner sans cesse la planification, la réalisation, la vérification et la correction ou l'ajustement des actions d'amélioration. Le cycle de Deming est donc composé de quatre étapes, d'où l'appellation répandue de cycle PDCA (P pour Plan (Planifier), D pour Do (Développer), C pour Check (Contrôler) et enfin A pour Act (Améliorer)) [Chardonnet *et al.*, 2002] (cf. Figure II-2).

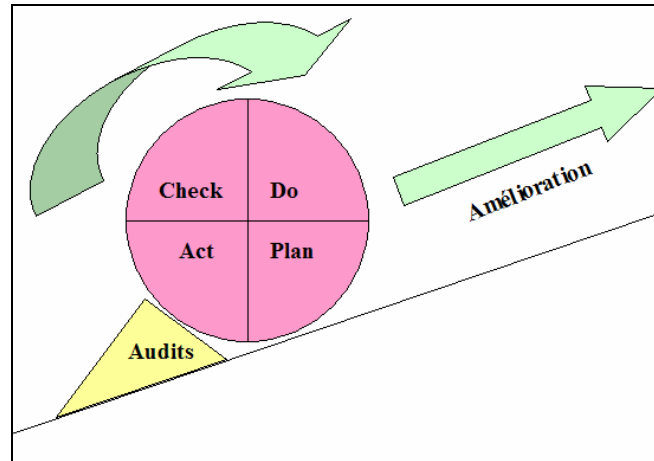


Figure II-2 : roue de Deming

Ce cycle modélise tout simplement le fait que chaque activité de l'entreprise peut être améliorée en quatre étapes récurrentes. Tout l'art d'une entreprise performante consistera à appliquer ce cycle aux niveaux pertinents de son organisation afin d'assurer que les boucles de rétroaction sont efficaces. Ceci afin de détecter à temps les améliorations possibles et mettre en œuvre les actions utiles au juste échelon de l'organisation.

Notons que la roue est symbolisée sur un plan incliné, décrivant la nécessité de progresser en permanence. La cale (les audits) représente l'obligation de vérifier la stabilité¹ du système à chaque cycle avant de continuer à progresser [Massot, 1999].

La mise en place d'une démarche qualité structurée, réfléchie et cohérente n'est pas une tâche simple. Comme nous l'avons vu précédemment, afin de guider les organisations, il existe plusieurs familles de méthodes. Nous nous intéresserons par la suite aux méthodes qui consistent à améliorer les processus. Commençons donc tout d'abord par définir ce terme couramment utilisé dans l'ingénierie des systèmes d'information.

1.2. Processus et processus logiciel

La littérature propose une multitude de définitions pour la notion de processus. Citons par exemple celle de la norme ISO 9001 [ISO, 2000] : « un processus est un ensemble d'activités, corrélées ou interactives, qui transforme des éléments d'entrée en éléments de sortie ».

¹ Vérifier la stabilité du système revient à s'assurer que le nouveau système mis en place apporte bien un gain **durable** pour l'organisation.

D'autres définitions insistent sur la finalité du processus [Vernadat, 1999] : « un processus est un ensemble partiellement ordonné d'étapes exécutées en vue de réaliser au moins un objectif ». D'autres encore mettent en valeur les rôles mis en jeu dans un processus [Morley, 2002] : « un processus représente l'organisation d'un ensemble finalisé d'activités effectuées par des acteurs et mettant en jeu des entités ».

Certains auteurs fournissent également des définitions qui permettent de regrouper toutes les informations citées précédemment. C'est le cas dans [Théroutte, 2002] : « un processus est défini comme un enchaînement partiellement ordonné d'exécution d'activités qui, à l'aide de moyens techniques et humains, transforme des éléments d'entrée en éléments de sortie en vue de réaliser un objectif dans le cadre d'une stratégie donnée ».

Dans le premier chapitre, nous avons évoqué le SPEM (Software Process Engineering Metamodel). Nous allons voir que les définitions précédentes sont en accord avec ce méta-modèle.

Un paquetage du SPEM, nommé « Process_Structure », propose une modélisation conceptuelle de la notion de processus. La Figure II-3 est issue de [Combemale *et al.*, 2006], qui est un extrait simplifié du méta-modèle SPEM originel.

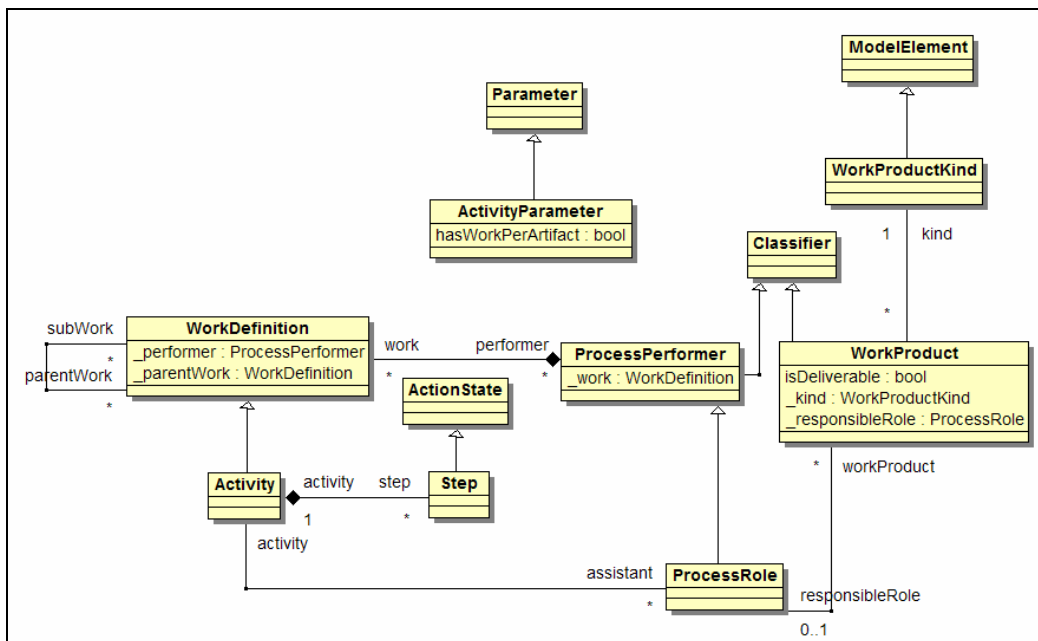


Figure II-3 : vue partielle du méta-modèle SPEM : le processus [Combemale *et al.*, 2006]

Ainsi, le SPEM utilise la notion d'opération (WorkDefinition). Une opération peut être décomposée en plusieurs sous-opérations (liens subWork et parentWork). Chaque opération est sous la responsabilité d'un rôle (ProcessPerformer). Un ensemble d'autres rôles (ProcessRole) peuvent également assister le rôle principal dans la réalisation d'activités (Activity). La notion d'activité est issue d'une spécialisation de la notion d'opération. L'OMG précise qu'une activité

est la réalisation concrète (réelle) d'une opération. L'opération est donc « théorique », au contraire de l'activité. Notons la présence de produits (WorkProduct) qui peuvent être utilisés, consommés ou réalisés par des rôles (ProcessRole) [Izza, 2006].

Il est possible de donner une représentation graphique très simplifiée à la notion de « processus » (Figure II-4), en utilisant le formalisme IDEF0 : « Integration Definition Function Modeling », plus communément désigné sous l'acronyme « SADT » (« Structured Analysis and Design Technic »). Nous utilisons le formalisme des actigrammes : des boîtes représentent des activités et des flèches représentent les relations entre activités. Les relations peuvent être de quatre types : entrée, sortie, contrôle ou ressources (ou moyens / compétences) [Lissandre, 1990].

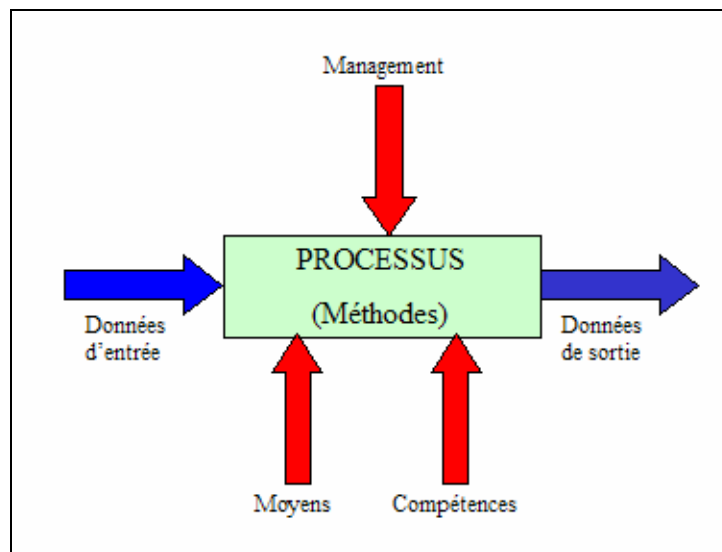


Figure II-4 : représentation schématique d'un processus

Un processus est réalisé grâce à des moyens (ressources humaines, matérielles...) et à des compétences. Il est réalisé en accord avec des procédures, des méthodes, des modes opératoires, sous le contrôle du management.

La norme ISO 9001 : 2000 différencie trois catégories de processus :

- les processus de management (ou de pilotage). Ils sont exprimés par la direction et déterminent la politique et les objectifs à atteindre. Ces processus pilotent et surveillent les autres processus. Il s'agit par exemple du processus de revue de direction, du processus de pilotage de l'amélioration continue...
- les processus de réalisation (ou métiers). Ils concernent directement la réalisation d'un produit ou d'un service. A ce titre, ils intègrent toutes les activités qui permettent leur réalisation, du recueil du besoin jusqu'à sa satisfaction. Il s'agit des processus de conception, de production, de vente...

- les processus supports. Ils sont indispensables à la mise en œuvre des processus de réalisation, car ils fournissent les moyens et les compétences nécessaires à leur accomplissement : citons les processus de gestion des ressources humaines (pour une entreprise dont ce n'est pas le cœur de métier, autrement il s'agirait d'un processus de réalisation !), de gestion des finances (*idem*)...

Nous nous intéressons particulièrement aux organisations qui mettent en œuvre ces différents types de processus sur un domaine (un « métier ») particulier : le « logiciel ». Il est donc d'usage de parler de « processus logiciel ». Nous adoptons la définition suivante ([Wreth, 1993], traduite en français dans [CRAM, 1993]¹) :

« un processus logiciel est un ensemble d'activités (*i.e.* : processus) permettant le développement et la maintenance de logiciels et de produits associés (plans de projet, documents de conception, programmes...) ».

Le processus logiciel est en fait le « macro-processus » qui décrit toutes les activités d'une organisation, qui œuvre dans le domaine de l'informatique. Il est composé de multiples « sous-processus » qui appartiennent chacun à un des trois groupes décrits par la norme ISO 9000 : 2000 (management, réalisation, support).

Nous avons vu que les démarches d'amélioration se basent souvent sur les processus. Cela sous-entend que l'amélioration des processus (et donc les approches SPI, lorsqu'il s'agit d'améliorer un processus logiciel) permet d'accroître la satisfaction des donneurs d'ordres, en améliorant la qualité, le respect des coûts, des délais...

Une organisation, peut donc, pour améliorer ses processus, appliquer une démarche type PDCA. Le point de départ (« Plan » du PDCA) consiste à caractériser la situation actuelle de l'organisation vis-à-vis de certains critères et à définir la cible désirée vis-à-vis de ces mêmes critères (il faut bien savoir où l'on se trouve pour savoir où l'on veut aller !). Les autres activités préconisées par le PDCA vont consister à parcourir le chemin séparant la position actuelle de la position cible, et à s'assurer que la situation est pérennisée. Le besoin de connaître et donc d'évaluer la performance des processus apparaît ici. Il convient alors de parler de capacité des processus et de maturité de l'organisation.

1.3. Capacité, maturité et modèle de maturité

Appliquée à l'être humain, la capacité est la « qualité de celui qui est en état de comprendre, de réaliser une activité » [Le Petit Robert, 1993]. La capacité d'un individu désigne donc le fait qu'il est capable d'obtenir un résultat (intellectuel, matériel...) vis-à-vis d'un problème donné. [Winter, 2000] lie la capacité avec les objectifs que l'intention initiale, qui avait motivé l'exécution d'une

¹ Le CRAM est le centre de recherche pour lequel travaillait la référence francophone (R. Basque) concernant un modèle de maturité précis : le CMM. Il a notamment traduit ce modèle en français.

activité, avait définis : « le concept de capacité remplit l'espace qui existe entre l'intention et le résultat, en prenant pour acquis que le résultat est conforme à l'intention initiale » (traduit par [Renard *et al.*, 2003]). La capacité d'un individu est donc l'adéquation entre le résultat attendu et le résultat obtenu, lors de l'exécution d'une activité.

En extrapolant à la notion de processus, la capacité désigne donc l'adéquation entre les résultats obtenus par l'exécution de celui-ci (c'est-à-dire en « sortie » de processus - cf. Figure II-4) et ceux qui étaient prévus (planifiés). La définition proposée dans [Paulk *et al.*, 1993] et traduite en français dans [CRAM, 1993] va dans ce sens :

« par capacité d'un processus, on entend la gamme des résultats attendus pouvant être obtenus en suivant ce processus. »

La capacité d'un processus d'une organisation fournit donc un moyen de prédire les résultats les plus vraisemblables pouvant être attendus pour le prochain projet qui sera entrepris par l'organisation et qui demande la mise en œuvre de ce processus précis. Prenons l'exemple d'une organisation dont le cœur de métier est de produire des logiciels. Le processus de tests des applications est logiquement un de ses processus opérationnels. La capacité de ce processus dépendra, par exemple, du temps qu'il lui faut pour tester une application, de la qualité et de l'exhaustivité des tests passés... Si l'entreprise est capable de reproduire son processus, invariablement, il est légitime de s'attendre à obtenir la même gamme de résultats, en sortie du processus.

Notons également que les résultats des tests (nombre moyen de défauts par ligne de code, nombre d'évolutions fonctionnelles envisagées...) seront un indicateur de la capacité du processus de réalisation technique.

Il est possible de définir des « niveaux de capacité », c'est-à-dire de proposer plusieurs degrés de capacité pour un processus. Le niveau de capacité du processus dépendra alors de la qualité de ses sorties, de leur précision...

Le processus « réaliser un doctorat » est composé, entre autre, d'un processus « réaliser des publications scientifiques ». Participer à quatre congrès en trois ans correspondra à une capacité du processus de niveau 2, alors que participer à six congrès et avoir publié dans un journal scientifique réputé correspondra au niveau supérieur.

Pour le SEI (Software Engineering Institute¹) [SEI, 2002a], [SEI, 2002b], la capacité d'un processus est à situer sur une échelle à six niveaux. Un processus peut donc être incomplet (niveau 0), exécuté (niveau 1), géré (niveau 2), défini ou documenté (niveau 3), géré quantitativement (niveau 4) et optimisé (niveau 5).

¹ Le SEI est un département du centre de recherche de l'université de Carnegie-Mellon (Pittsburgh - Pennsylvanie – Etats-Unis),

La maturité désigne, quant à elle, « l'étape dans laquelle se trouve un organisme qui a atteint son plein-développement » [Le Petit Larousse illustré, 2001]. Appliquée à l'être humain, la maturité est la période de la vie durant laquelle il a atteint son plein-développement biologique, intellectuel, sociologique, psychologique... La maturité d'un être humain est donc l'obtention d'un ensemble d'aptitudes.

De la même manière, pour pouvoir être qualifiée de « mature », une organisation devra, elle aussi, posséder une certaine liste d'aptitudes, c'est-à-dire que les processus qu'elle utilise doivent avoir atteint, chacun, un certain niveau de capacité.

Une échelle est également établie pour situer la maturité d'une organisation. Cette échelle, pour le SEI, comporte cinq niveaux. Ainsi, selon [SEI, 2002a], [SEI, 2002b] « la maturité d'une organisation correspond au degré auquel une organisation a **explicitement** et **invariablement** déployé des processus gérés (niveau 2), définis ou documentés (niveau 3), gérés quantitativement (niveau 4), optimisés (niveau 5) ». Le niveau 1, qui n'est pas cité dans la phrase précédente, correspond au niveau initial (ad hoc).

Reprenons l'exemple du processus « réaliser un doctorat ». Le Tableau II.1 propose pour quelques processus une caractérisation de deux niveaux de capacité et intègre la notion de niveau de maturité.

	Niveau de capacité 2	Niveau de capacité 3
Processus 1 : « Réaliser des publications scientifiques » : - participer à des congrès - publier dans un journal scientifique	4 0	6 1
Processus 2 : « Gérer un projet » - suivre une formation de gestion de projet - réaliser un planning sur la base d'un WBS validé - suivre le planning - planifier des points d'avancement	X Ø Mensuellement Semestriels	X X Hebdomadairement Mensuels
Processus 3 : « Communiquer et former » - dispenser des cours - suivre une formation à l'animation de réunion	30 h.h Ø	60 h.h X
...		
Niveau de maturité du processus général « Réaliser un doctorat »	2	3

Tableau II.1 : exemple de capacité

Le processus général « réaliser un doctorat » sera de niveau de maturité 3 si tous les processus qui le composent ont atteint le niveau de capacité 3 !

Nous proposons un autre exemple simple, pour illustrer le lien entre maturité et capacité. Ainsi, sur la Figure II-5, chaque stade (*i.e.* : niveau) d'évolution de l'espèce humaine représente un niveau de maturité du processus général de l'Evolution.

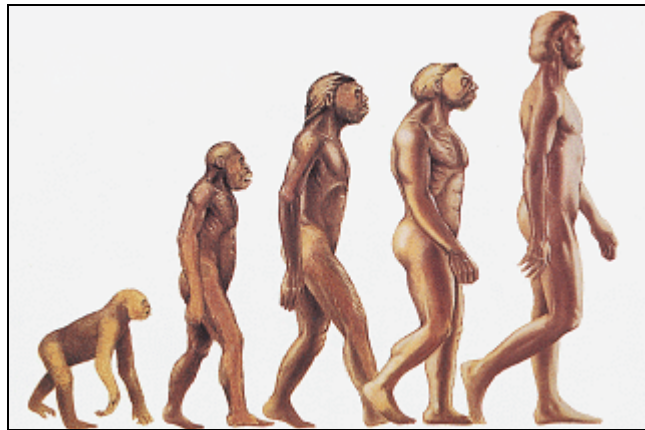


Figure II-5 : évolution de l'espèce humaine au cours des âges

Chaque niveau d'évolution (Homo Erectus, Homo Sapiens...) est défini par des caractéristiques précises (*i.e.* : capacités) : volume de la boîte crânienne, bipédie, utilisation d'outils... Ainsi, l'analyse d'un individu selon ces critères permet d'établir son niveau d'évolution, c'est-à-dire son niveau de maturité !

Il faut bien comprendre que dans le cadre d'une organisation, la notion de maturité ne s'applique pas à chacun de ses processus individuellement, mais à leur ensemble. Ainsi, pour une organisation du secteur informatique, il convient de parler de la maturité globale de son processus logiciel, et non pas de la maturité individuelle de chacun des processus qui composent le processus logiciel.

Précisons également que le passage d'un niveau de maturité au niveau suivant implique généralement une augmentation du « périmètre des améliorations ». Sur l'exemple de la Figure II-6, le niveau de maturité « X » s'atteint en améliorant les processus 1 et 2, jusqu'au niveau de capacité « Y ». Le niveau de maturité « X+1 » va consister à refaire une boucle d'amélioration sur ces processus, mais également sur d'autres (processus 3 et 4), afin de tous les emmener au niveau de capacité « Y+1 ».

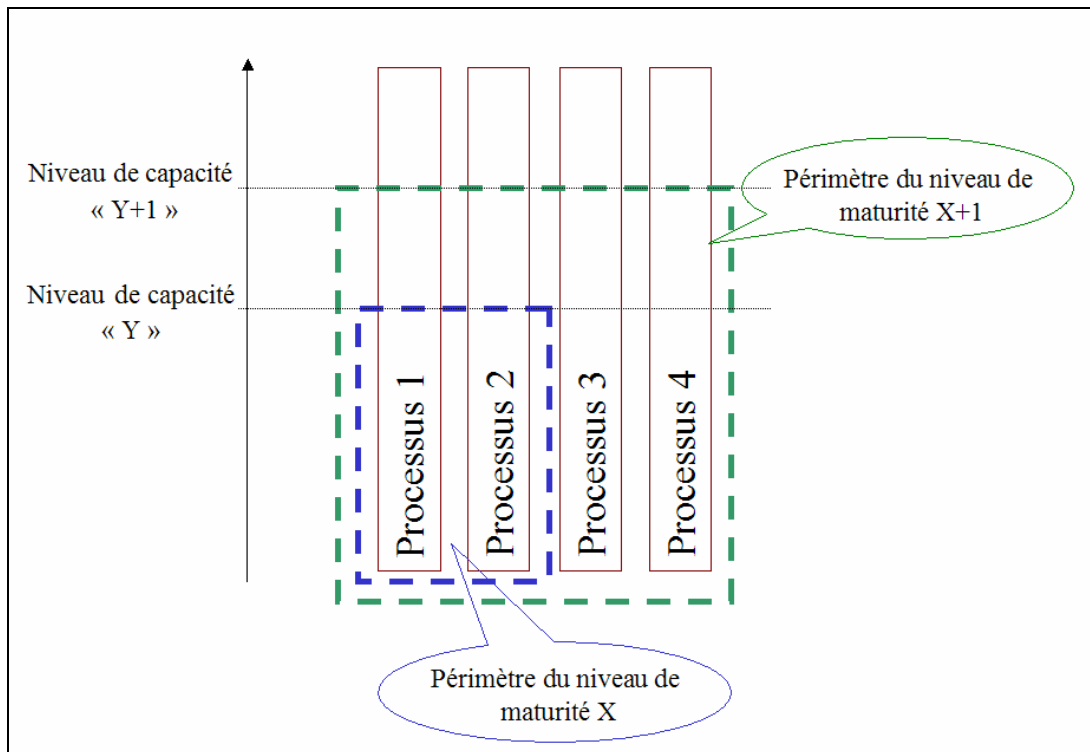


Figure II-6 : lien entre maturité et capacité

Les capacités des différents processus utilisés par une organisation mature (niveau 2, 3, 4 ou 5) sont donc **connues**. Une organisation mature est capable de fournir des prestations et produits avec un niveau de qualité stable et connu (invariablement). A partir de là, une organisation, qui atteint un niveau de maturité que ses concurrents n'ont pas atteint, dispose logiquement d'un atout concurrentiel fort.

Le fait de découper l'évolution de la maturité et les capacités des processus en différents niveaux n'est rien d'autre que le résultat d'un processus d'abstraction de la réalité. Il s'agit donc, d'après le premier chapitre de ce mémoire, d'un « modèle d'**évolution** de la maturité et des capacités ». Usuellement, ces modèles sont appelés « modèles de maturité ». Notons qu'il s'agit donc d'un abus de langage, puisque c'est bien l'**évolution** de la maturité et des capacités qui est ainsi modélisée.

Un modèle de maturité est donc une représentation simplifiée de l'**évolution** que peut suivre la maturité d'une organisation, c'est-à-dire dans quelle mesure elle met en œuvre des processus gérés, définis ou documentés, gérés quantitativement et optimisés.

1.4. Origine des modèles de maturité : la QMMG

Il existe des dizaines de modèles de maturité différents, mais tous puisent leurs origines dans une grille, proposée par Crosby¹ en 1979 [Crosby, 1979], [Farser *et al.*, 2002]. Cette grille a pour nom QMMG : Quality Management Maturity Grid (littéralement « grille de maturité du management de la qualité »). Précisons qu'en 1979, dans le monde de la qualité, le terme maturité n'était pas associé à la notion de processus. La QMMG est plus un modèle d'évolution de la prise de conscience de l'importance de la qualité dans une organisation qu'un modèle de maturité, au sens où nous l'entendons aujourd'hui. Par contre, Crosby, à travers sa grille, a posé les « briques élémentaires » qui sont toujours utilisées aujourd'hui.

Ainsi, la grille définit cinq niveaux de prise de conscience des nécessités de la qualité. Ces cinq niveaux sont :

- **P'incertitude** : l'entreprise n'intègre pas les outils de la qualité comme outils de management,
- **le réveil** : la qualité est reconnue comme importante, mais aucune action concrète n'est décidée,
- **la vision éclairée** : des démarches d'amélioration de la qualité sont mises en place,
- **la sagesse** : la prévention fonctionne, les problèmes sont identifiés à la source, les actions correctives sont mises en place,
- **la certitude** : la qualité devient un outil de management, la direction fait face et bâtit un plan formel d'action.

Pour chacun de ces cinq niveaux, Crosby examine le statut de la qualité et sa place dans l'organisation des entreprises, selon six axes (catégories) :

- l'attitude de la direction par rapport à la qualité : comment la direction **utilise-t-elle** les outils de la qualité dans le management ?
- le niveau d'organisation de la qualité : quel est le **positionnement hiérarchique** du responsable / département qualité dans l'organisation ?
- la gestion des problèmes : jusqu'à quel point les problèmes sont-ils **anticipés** ?

¹ Philip B. Crosby (né en 1926) est une référence en matière de qualité. Il est célèbre pour sa formule « la qualité, c'est gratuit » dont il a fait le titre de son ouvrage le plus connu « *Quality is free* ».

- le coût de la non-qualité en pourcentage des ventes : quel est le **niveau de maîtrise** de la qualité sur la production ?
- les actions d'amélioration de la qualité : jusqu'à quel point l'amélioration est-elle **soutenue** ?
- un bilan de l'attitude de l'entreprise.

Chaque colonne de la QMMG représente un niveau et chacune des six catégories occupe une ligne. Le Tableau II-2 est un extrait de la grille QMMG. La grille complète est disponible en annexe.

	Incertitude	Réveil	Vision éclairée	Sagesse	Certitude
...	...				
La gestion des problèmes	Problèmes traités quand découverts et souvent partiellement résolus	Des équipes traitent les problèmes avec une vision à court terme	Problèmes résolus dans l'ordre et actions correctives régulières	Problèmes identifiés à leur source	A l'exception de rares cas, prévention des problèmes.
...	...				

Tableau II-2 : extrait de la QMMG de Crosby [Fraser *et al.*, 2002]

L'utilisation de la grille par une organisation est prévue de la manière suivante :

- la première étape est de caractériser la situation initiale de l'organisation. Elle doit définir, pour chaque catégorie (par exemple, la gestion des problèmes) à quel niveau elle se trouve (« incertitude », « réveil »...),
- la seconde étape consiste à définir la cible souhaitée. Le cas idéal est logiquement d'atteindre le niveau « certitude » pour chaque catégorie,
- Enfin, l'organisation doit progresser, c'est-à-dire passer de la situation initiale à la situation cible souhaitée. Corsby propose un programme adapté en quatorze points pour réaliser cette progression. Le programme recouvre des thèmes aussi variés que la mise en place d'une journée « zéro défaut », la conception d'indicateurs de mesure et l'amélioration permanente (le détail de ces quatorze points est disponible en annexe).

En résumé, Crosby propose :

- une grille modélisant la maturité des entreprises vis-à-vis de la prise de conscience de la nécessité de qualité, qui est un outil d'auto évaluation,
- un programme qui propose aux entreprises une liste de points à améliorer afin de progresser vers les niveaux supérieurs de la grille.

La plupart des modèles de maturité ont une structure comparable, c'est-à-dire :

- d'une part, une modélisation de la maturité des organisations, en fonction de certains critères,
- et d'autre part, une manière d'évoluer progressivement vers les niveaux supérieurs.

Les parties suivantes nous permettent de présenter les modèles qui ont un intérêt direct au regard de la problématique industrielle en lien avec nos travaux. Une étude comparative d'autres modèles de maturité a été réalisée dans le cadre de la thèse [Deguil *et al.*, 2006].

2. Capability Maturity Model Integration (CMMI)

2.1. L'origine du CMMI : le Capability Maturity Model (CMM)

Le Software Engineering Institute a tout d'abord établi, sous l'impulsion du département de la défense américaine (DoD), dans les années quatre-vingt, un « questionnaire de maturité ». Il s'agissait d'un ensemble de questions fermées, portant aussi bien sur le processus de développement informatique de l'entreprise interrogée, sur son organisation ou encore sur son management. Ainsi, les questions étaient diverses et variées, par exemple :

- est-ce que les activités de gestion de configuration font l'objet d'un plan ?
- Est-ce que les projets suivent une procédure écrite pour réaliser les activités d'ingénierie logicielle (une procédure décrivant, par exemple, les outils et méthodes pour produire et maintenir un logiciel) ?
- Est-ce que le management et les équipes techniques suivent des formations concernant l'amélioration des processus ?
- ...

Grâce à la base de connaissance établie par les réponses à ce questionnaire, le SEI a ensuite imaginé, en 1990 le CMM : Capability Maturity Model (« modèle d'évolution des capacités logiciel », traduction : [CRAM, 1993]).

Le CMM est un guide pour l'amélioration des pratiques en matière de développement et de maintenance du logiciel. « Il est composé de pratiques clés qui expriment la meilleure façon de travailler pour produire des logiciels **de qualité**, avec une productivité accrue et dans le **respect du budget** et des **délais** » [Basque, 1993]. Ces pratiques clés ont été identifiées, grâce au questionnaire cité précédemment et donc à la suite d'une vaste consultation de la communauté informatique, sur la base des succès et des échecs vécus, sur des milliers de projets passés ou en cours.

Un rapport du SEI présente le CMM [Wreth, 1993] de la manière suivante :

« le CMM fournit une **structure conceptuelle** pour l'amélioration de la gestion et du développement de produits logiciels dans un cadre formel et discipliné. Il ne garantit pas que le produit logiciel sera obtenu avec succès ou bien que tous les problèmes de l'ingénierie logiciel seront résolus. Il identifie des **pratiques**¹ à respecter pour l'obtention d'un processus logiciel mature, mais il ne prétend être ni exhaustif, ni autoritaire ».

Des détails sur la structure conceptuelle du CMM ainsi que quelques exemples sont fournis en annexe.

¹ Nous reviendrons sur la définition de ce terme par la suite.

Le périmètre d'application du CMM se limite à l'industrie du « développement logiciel ». Cependant, très vite, l'engouement autour de ce modèle allait dépasser cette seule industrie, donnant naissance à de multiples « clones » : SE-CMM (pour System Engineering) [SEI, 1995], SA-CMM (pour Software Acquisition) [SEI, 2002c], IPD-CMM (pour Integrated Product Development) [SEI, 1997] et même People-CMM (pour la gestion des ressources humaines) [SEI, 2001]. Tant et si bien qu'il fallut rebaptiser le CMM « initial » en SW-CMM (SW pour Software) afin d'éviter les confusions.

Cette prolifération de modèles (complémentaires ou concurrents) engendra une certaine confusion allant à l'encontre du but initialement recherché : la rationalisation des processus. Aussi fut-il décidé de rassembler (d'intégrer) tant bien que mal les divers modèles au sein d'un modèle unique baptisé Capability Maturity Model Integration (CMMI). Dans la version actuelle (CMMI version 1.2) datant de 2006, seule la gestion des ressources humaines n'a pas été intégrée [Goffinet, 2004].

2.2. Le CMMI

Le modèle CMMI est un modèle assez complexe, pour plusieurs raisons :

- il est censé couvrir plusieurs domaines : ingénierie système (SE), ingénierie logicielle (SW), intégration produit et développement de processus (IPPD) et enfin gestion des fournisseurs (SS). Il possède donc un caractère « générique » qui rend son appréhension plus difficile,
- il laisse un haut degré de liberté à l'organisation qui le met en place : le SEI détaille « quoi obtenir ? » mais pas « comment faire ? ». Il précise certaines pistes par l'intermédiaire d'exemples de produits du travail et de sous-pratiques (Figure II-7),
- sa structure conceptuelle est vaste : il met en jeu de nombreux types de composants différents.

CMMI-SE/SW/PPD/SS, v1.1 Staged Representation
Examples of types of work products for which size estimates are made include the following: [PA163,IG101,SP102,N103]
<ul style="list-style-type: none">• Deliverable and nondeliverable work products• Documents• Operational and support software
Examples of size measures include the following: [PA163,IG101,SP102,N104]
<ul style="list-style-type: none">• Number of functions• Function points• Source lines of code• Number of classes and objects• Number of requirements• Number of interfaces• Number of pages• Number of inputs and outputs• Number of technical risk items• Volume of data

Figure II-7 : extrait du CMMI

Afin de ne pas rendre la lecture et la compréhension de cette partie trop fastidieuse, nous nous concentrons sur la présentation des composants clés du modèle. La bibliographie sur le sujet est à la mesure du succès que connaît le CMMI. Il est donc aisé, pour celui qui le désire, de trouver de plus amples informations. Les ouvrages de référence sont [SEI, 2002a] et [SEI, 2002b].

2.2.1. *Eléments de base*

Nous présentons ici les éléments les plus importants de la structure conceptuelle dans le cadre de nos travaux : bonne pratique, objectif et secteur clé.

2.2.1.1. Bonne pratique

Le dictionnaire [Le petit Larousse illustré, 2001] définit une pratique comme étant : « une application, une mise en œuvre des règles [...] d'une science, d'une technique ». Il faut donc différencier la règle (« prescription qui s'impose à quelqu'un dans un cas donné »), qui encadre la réalisation d'une activité, de la pratique, qui est l'application concrète de cette règle. Une pratique est donc une manière de faire, de réaliser une activité, en respectant les règles propres au domaine de cette activité.

L'activité « rédiger un mémoire de doctorat » est encadrée par certaines règles :

- des règles bibliographiques, qui imposent des normes de citations et de références,
- des règles linguistiques et syntaxiques, qui cadrent la rédaction,
- des règles éthiques, qui obligent à citer les références,
- ...

Les règles n'imposent généralement pas une seule « manière de faire » : les références bibliographiques peuvent être repérées entre crochets, entre parenthèses... Il est possible de faire

apparaître les noms des auteurs cités, ou seulement les premières lettres... Le choix d'écrire une référence [XXXX, 19XX] est donc une mise en œuvre des règles. Il s'agit d'une pratique.

Dans un milieu professionnel donné, les acteurs sont constamment confrontés, pour partie, aux mêmes problématiques et donc amenés à résoudre les mêmes types de problèmes. Il paraît donc pertinent d'essayer d'identifier, en commun, la « meilleure manière » de réaliser une activité ou un ensemble d'activités. C'est, par exemple, le but d'une « communauté de pratiques » [Wenger, 2005]. On ne parle alors plus de pratique, mais de bonne pratique.

La définition proposée par l'encyclopédie Wikipedia va dans ce sens [Wikipedia, 2008c] : « le terme bonne pratique désigne, dans un milieu professionnel donné, un ensemble de comportements qui font consensus et qui sont considérés comme indispensables ».

Une bonne pratique étant issue d'une analyse consensuelle et d'un retour d'expériences, il s'en déduit une mécanique de « prescription *de la règle* » à appliquer aux activités (*la règle de l'art*).

Les référentiels de bonnes pratiques proposent donc un ensemble « d'activités attendues » qui ont déjà démontré leur efficacité et leur pertinence *sur le terrain*. Dans notre vision, la bonne pratique sera, en quelque sorte, le patron¹ (au sens de « cadre ») dans lequel une activité, pour être réalisée **correctement**, doit s'inscrire. Nous illustrons ceci avec la Figure II-8. Le formalisme utilisé est IDEF0.

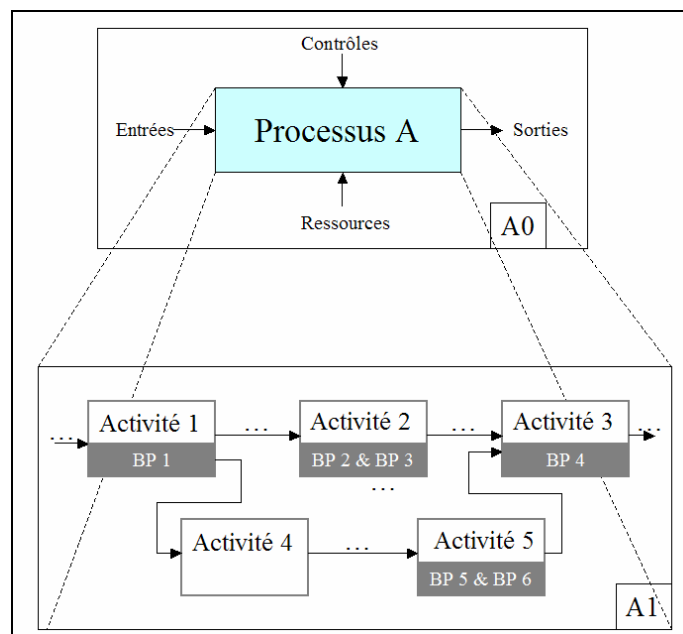


Figure II-8 : bonnes pratiques

¹ Nous nous sommes inspirés de la notion de « design pattern » (patron de conception), utilisée en génie logiciel. « Les patrons de conception décrivent des solutions standard pour répondre à des problèmes d'architecture et de conception des logiciels. À la différence d'un algorithme qui s'attache à décrire d'une manière formelle comment résoudre un problème particulier, les patrons de conception décrivent des procédés de conception généraux. On peut considérer un patron de conception comme une formalisation de bonnes pratiques, ce qui signifie qu'on privilégie les solutions éprouvées » [Wikipedia, 2008d].

Nous considérons les bonnes pratiques (en gris sur la Figure II-8) comme étant des préconisations qui décrivent et permettent de « guider » l'exécution d'activités. Cette vision se situe dans le prolongement de celle du SEI, qui précise dans [SEI, 2002a] : « un processus [...] est constitué d'activités, qui peuvent être reconnues comme étant des implémentations des pratiques du modèle CMMI. Ces activités peuvent être en lien avec une ou plusieurs pratiques [...] ».

C'est d'ailleurs ainsi que [Miler *et al.*, 2004] positionne la notion de pratique par rapport à celle d'activité. La pratique se situe dans le monde théorique « désiré », alors que l'activité se situe dans le monde réel (relation « realization » sur la Figure II-9). Cette figure est proposée dans [Miler *et al.*, 2004] : il s'agit d'une extension d'un des modèles du SPEM. Il y a une équivalence entre la notion « Artifact » sur la Figure II-9 et celle de « WorkProduct » sur la Figure II-3.

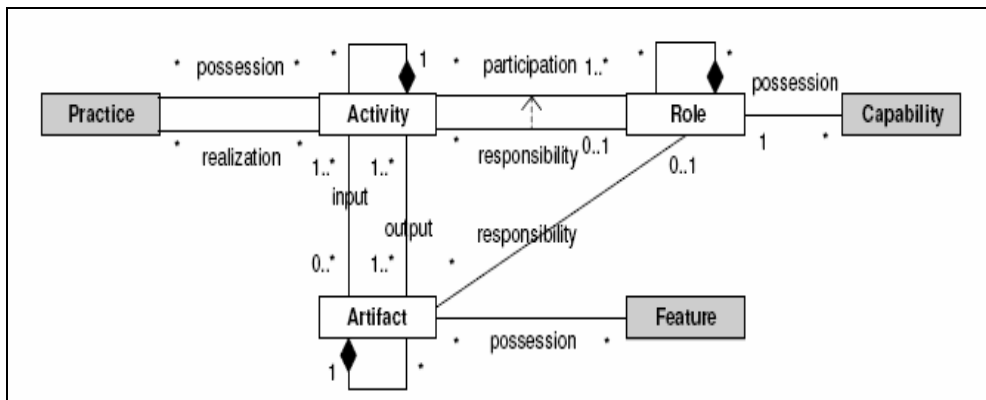


Figure II-9 : pratique et activité [Miler *et al.*, 2004]

Finalement, l'OMG, à travers le SPEM, propose une définition générale pour la notion de bonne pratique, qui nous semble pertinente : « une bonne pratique représente une manière qui a fait ses preuves ou une stratégie pour réaliser une tâche dans le but d'atteindre un *objectif* qui aurait un impact positif sur les produits du travail ou sur la qualité des processus mis en jeu ». Le respect de certaines réglementations, la mise en place de certaines méthodes d'amélioration ou la certification de certaines normes peut imposer le déploiement et le respect de bonnes pratiques métiers. Dans ce cadre, les bonnes pratiques ne sont plus de simples préconisations, mais également des exigences. Dans le contexte d'un modèle de maturité ou de capacité, le déploiement des bonnes pratiques sera l'outil de mesure de la capacité, et donc de la maturité. Cette vérification se fait grâce au recueil et à l'analyse de preuves de déploiement. Pour le CMMI, la méthode d'évaluation est nommée SCAMPI « Standard CMMI Appraisal Method for Process Improvement ». Le lecteur intéressé pourra se reporter à [SEI, 2006].

Le modèle CMMI différencie pratique spécifique (SP – Specific Practice) et pratique générique (GP – Generic Practice). Une pratique spécifique concerne une activité technique précise, alors qu'une pratique générique permettra de s'assurer que le processus associé est efficace et reproductible. Les pratiques génériques sont donc « transversales », car elles peuvent s'appliquer à n'importe quel processus.

« Etablir un système de gestion de configuration » est le libellé d'une pratique spécifique qui appartient au CMMI¹. Cette pratique vise à établir et maintenir des systèmes de gestion de configuration et de gestion des changements, afin de maîtriser les « produits du travail ». Le SEI précise également que la réalisation de (ou des) l'activité(s) associée(s) à cette pratique permet d'obtenir, par exemple : des « produits du travail » maîtrisés, des procédures propres au système de gestion des changements (procédure de contrôle d'accès...), une base de données des demandes de changement...

« Planifier le processus », ou encore « Former les ressources »² sont des exemples de libellés de pratiques génériques. Elles s'appliquent à chaque « processus attendu » (*i.e.* : secteur clé, cf. 2.2.1.2), et donc aux activités qui les composent. La première va consister à réaliser un plan qui doit prévoir, décrire et encadrer l'exécution de chaque processus, au niveau projet. La seconde consiste simplement à former les ressources, afin de s'assurer qu'elles possèdent les compétences nécessaires pour exécuter les activités prévues. Appliquées au processus de gestion de configuration, ces deux pratiques génériques vont donc permettre d'obtenir le plan de gestion de configuration du projet et former les ressources qui seront amenées à utiliser les systèmes de gestion de configuration et de gestion des changements.

2.2.1.2. Objectif et secteur clé

Au sens du CMMI, un sous-ensemble de pratiques contribue à l'atteinte d'un objectif. Le découpage entre pratiques spécifiques et pratiques génériques se retrouve donc également au niveau des objectifs. En conséquence, il existe des objectifs spécifiques (SG – Specific Goal - relatifs à des activités métiers) et des objectifs génériques (GG – Generic Goal).

Ainsi, la pratique spécifique citée dans la partie précédente est associée à un objectif spécifique dont le libellé est : « Etablir les configurations de référence ». Il consiste à :

- identifier les items de configuration (la première pratique spécifique associée à cet objectif),
- établir un système de gestion de configuration (la seconde pratique spécifique associée à cet objectif – il s'agit de l'exemple cité dans la partie précédente),
- créer ou mettre à disposition les configurations de référence (la troisième pratique spécifique associée à cet objectif).

De manière similaire, les deux pratiques génériques sont rattachées à un objectif générique, dont le libellé est « Implémenter un processus géré ».

L'enchaînement de certaines pratiques permet de définir un « chemin », une trajectoire, vers un objectif particulier. Cela sous-entend que le chemin proposé par les bonnes pratiques n'est pas le seul envisageable. Il s'agit seulement de celui « qui a déjà fait ses preuves ». Ceci est pris en compte par le CMMI car, lors d'une évaluation du niveau de maturité d'une organisation ou de la

¹ Il s'agit de la SP 1.2 du secteur clé « gestion de la configuration ».

² Il s'agit des GP 2.2 et 2.5.

capacité de ses processus, c'est l'atteinte des objectifs qui est vérifiée et non pas l'implémentation des bonnes pratiques. Pragmatiquement, le CMMI définit trois classes de composants :

- les composants requis (« required ») : il s'agit des objectifs spécifiques et génériques. L'atteinte des objectifs est vérifiée dans les évaluations,
- les composants attendus (« expected ») : il s'agit des pratiques génériques et spécifiques. Une évaluation CMMI vérifiera que les pratiques sont mises en place. Si ce n'est pas le cas, **une méthode alternative** est autorisée,
- les composants donnés à titre informatif (« informative ») : il s'agit des sous-pratiques, des descriptions des produits du travail... Leur présence dans le modèle CMMI a pour seul but d'aider les organisations à comprendre ce qui doit être mis en place.

Un secteur clé (PA – Process Area¹) est, quant à lui, un domaine particulier sur lequel une organisation doit se concentrer dans le but d'améliorer son processus logiciel. Dans le CMMI, les secteurs clés sont des regroupements d'objectifs (et donc, indirectement, de pratiques !). Ils sont caractérisés par une typologie d'activités métier.

Le terme « clé » renvoie à des ouvertures contrôlées. En cela, il faut comprendre que certains secteurs ne sont pas indispensables pour atteindre un niveau de maturité donné. Le CMMI ne décrit pas en détail tous les secteurs du processus logiciel, mais seulement ceux qui ont été jugés déterminants.

Par exemple, trois objectifs spécifiques sont associés au secteur clé « Gestion de configuration (CM) » :

- « établir les configurations de référence »,
- « suivre et contrôler les changements » (Cf. exemple de la partie précédente),
- « assurer l'intégrité des données ».

Un secteur clé n'est pas un sous-processus du processus logiciel. Le CMMI ne propose pas des processus « prêts à l'emploi ». L'implémentation d'un secteur clé du CMMI ne correspondra pas forcément à un seul et même processus dans l'organisation. Une étape d'appropriation et de personnalisation est nécessaire. Une organisation souhaitant mettre en œuvre le CMMI devra concevoir elle-même ses processus et agencer ses activités à sa manière, en fonction de ses contraintes propres (organisation, méthodologie de projet...).

Par exemple, bien qu'il existe un secteur clé « Gestion de configuration » dans le CMMI, il n'existera pas nécessairement un processus « Gestion de configuration » dans la cartographie des processus de

¹ La bibliographie utilise également la traduction « domaine de processus ».

L'organisation qui met en œuvre le CMMI. Les activités rattachées aux pratiques qui correspondent à ce secteur clé peuvent tout à fait être disséminées dans d'autres processus.

En résumé, en assimilant un secteur clé à un « processus attendu distribué », il existe une analogie entre pratique et activité, d'une part, et entre objectif du processus et objectif du secteur clé d'autre part (cette notion a été introduite par les définitions apportées dans la partie 1.2).

2.2.1.3. Résumé de la structure

La Figure II-10 montre les principaux composants du modèle CMMI.

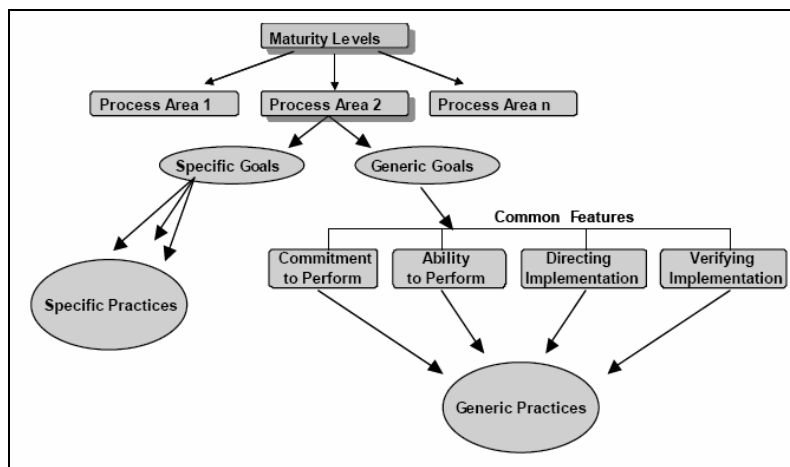


Figure II-10 : les principaux composants du modèle CMMI

Ainsi, un secteur clé est composé d'objectifs spécifiques et d'objectifs génériques. Un objectif est atteint via le déploiement de pratiques (spécifiques et génériques).

Afin de s'adapter à la politique qualité d'amélioration continue de l'organisation qui souhaite mettre en œuvre le CMMI, le SEI propose deux manières d'implémenter son modèle de maturité. Le SEI désigne la première comme étant l'architecture « discrète ou étagée » (staged) [SEI, 2002a] et la seconde comme étant l'architecture « continue » (continuous) [SEI, 2002b].

2.2.2. Les deux architectures : présentation et comparaison

L'architecture étagée utilise les niveaux de maturité. Ils sont au nombre de cinq (comme nous l'avons vu au 1.3) :

- *Niveau initial (niveau 1)* : les processus utilisés par l'organisation sont généralement « chaotiques ». Très peu de processus sont définis et la réussite dépend de l'effort individuel. Les organisations au niveau initial fournissent souvent des produits et services acceptables, mais de nombreux dépassements de budget ou de planning sont à déplorer.

- *Niveau « reproductible » (niveau 2) : une gestion de projet élémentaire est définie pour assurer le suivi des coûts, des délais et des fonctionnalités du produit. Les processus mis en place permettent de reproduire la réussite de projets d'un même domaine d'application.*
- *Niveau « défini » (niveau 3) : les activités de gestion et d'ingénierie sont documentées, normalisées et intégrées dans le processus standard de l'organisation.*
- *Niveau « maîtrisé quantitativement » (niveau 4) : des mesures détaillées sont prises en ce qui concerne le déroulement des processus et la qualité des produits du travail. Les processus et le niveau de qualité des produits sont connus et contrôlés quantitativement.*
- *Niveau « d'optimisation » (niveau 5) : une amélioration continue du processus est mise en œuvre par une rétroaction quantitative émanant du processus lui-même et par l'application d'idées et de technologies novatrices.*

Chaque niveau de maturité est composé d'un certain nombre de secteurs clés. Une organisation passe du niveau 1 (niveau initial) au niveau 2 en implémentant toutes les pratiques spécifiques relatives aux secteurs clés qui appartiennent au niveau de maturité 2, ainsi que toutes les pratiques génériques qui appartiennent à l'objectif générique 2.

Le passage au niveau de maturité 3 implique que l'organisation implémente toutes les pratiques spécifiques des secteurs clés relatifs au niveau de maturité 3. Il implique également de déployer les pratiques génériques relatives aux objectifs génériques 2 et 3 sur l'ensemble des secteurs clés déployés jusqu'à maintenant. Tel un jeu de poupées russes, chaque niveau est intégré dans le niveau qui lui est supérieur (cf. Figure II-11).

L'architecture continue n'utilise pas la notion de niveau de maturité. Au sein d'une organisation, chaque secteur clé possède un niveau de capacité. Tous les secteurs clés n'atteignent pas forcément le même niveau, ce qui permet de déceler les points forts et les points faibles de l'organisation. La liberté de manœuvre y est donc plus grande, car l'organisation a le choix dans l'ordre de mise en œuvre des secteurs clés [Reifer, 2000]. Notons que certains travaux proposent des méthodes de priorisation de déploiement des secteurs clés [Huang *et al.*, 2006].

Dans l'architecture continue, il existe un objectif générique (GG 1) qui impose à l'organisation de déployer toutes les pratiques spécifiques. Pour un secteur clé donné, l'atteinte du niveau de capacité 1 implique que l'organisation ait déployé toutes les pratiques spécifiques relatives au secteur clé en question, donc que l'objectif générique 1 soit atteint. L'atteinte du niveau de capacité 2 implique que le secteur clé soit déjà au niveau 1 et que l'organisation ait déployé toutes les pratiques génériques relatives à l'objectif générique 2. La règle est donc la suivante : « un

secteur clé est au niveau de capacité « Y », si l'organisation a atteint les objectifs génériques « Y », « Y-1 »... ».

Dans cette architecture, le SEI propose une classification « informative » des secteurs clés, qui se rapproche de la classification des processus proposée par l'ISO 9000 : 2000 (cf. 1.2). Le SEI distingue quatre catégories :

- management de projet,
- management de processus,
- ingénierie,
- support.

Le Tableau II-3 montre la répartition des vingt cinq secteurs clés, selon les deux architectures du CMMI.

	Management Projet	Management processus	Ingénierie	Support
Niveau de maturité 2	<ul style="list-style-type: none"> • Planification (PP) • Suivi et contrôle (PMC) • Contrats fournisseurs (SAM) 		<ul style="list-style-type: none"> • Gestion des exigences (REQM) 	<ul style="list-style-type: none"> • Assurance qualité produits et processus (PPQA) • Gestion de configuration (CM) • Mesures et analyse (MA)
Niveau de maturité 3	<ul style="list-style-type: none"> • Gestion de projet intégrée (IPM) • Gestion des risques (RSKM) • Gestion intégrée des fournisseurs (ISM) • Equipe intégrée (IT) 	<ul style="list-style-type: none"> • Focalisation sur le processus de l'organisation (OPF) • Définition du processus de l'organisation (OPD) • Formation de l'organisation (OT) 	<ul style="list-style-type: none"> • Développement des exigences (RD) • Solution technique (TS) • Intégration produit (PI) • Vérification (VER) • Validation (VAL) 	<ul style="list-style-type: none"> • Environnement organisationnel pour l'intégration (OEI) • Analyse décision et résolution (DAR)
Niveau de maturité 4	<ul style="list-style-type: none"> • Gestion de projet quantitative (QPM) 	<ul style="list-style-type: none"> • Performance du processus de l'organisation (OPP) 		
Niveau de maturité 5		<ul style="list-style-type: none"> • Innovation de l'organisation et déploiement (OID) 		<ul style="list-style-type: none"> • Analyse des causes et résolution (CAR)

Tableau II-3 : organisation des secteurs clés par niveau de maturité et catégories

Comme nous l'avons vu, les deux architectures sont radicalement différentes. Alors que l'architecture continue s'intéresse à la capacité d'un secteur clé, l'architecture discrète s'intéresse à la maturité globale de toute une organisation. Cependant, concrètement, ce sont les mêmes pratiques et les mêmes objectifs qui sont mis en jeu.

Il est possible de définir des profils cibles, qui précisent quels secteurs clés doivent être mis en œuvre, et à quel niveau de capacité, pour correspondre au niveau de maturité utilisé dans la représentation discrète du CMMI [Chandelon, 2004]. Ces équivalences sont illustrées par la Figure II-11.

Secteur clé	Abr.	Niveaux de Maturité	Niveaux de capacité				
			NC 1	NC 2	NC 3	NC 4	NC 5
Requirements Management	REQM	2	Profil Cible 2				
Measurement and Analysis	MA	2					
Project Monitoring and Control	PMC	2					
Project Planning	PP	2					
Process and Product Quality Assurance	PPQA	2					
Supplier Agreement Management	SAM	2					
Configuration Management	CM	2					
Decision Analysis and Resolution	DAR	3	Profil Cible 3				
Product Integration	PI	3					
Requirements Development	RD	3					
Technical Solution	TS	3					
Validation	VAL	3					
Verification	VER	3					
Organizational Process Definition	OPD	3					
Organizational Process Focus	OPF	3					
Integrated Project and Product Management	IPPD	3					
Risk Management	RSKM	3					
Integrated Supplier Management	ISM	3					
Organizational Training	OT	3					
Integrated Teaming	IT	3					
Organizational Environment for Integration	OEI	3					
Organizational Process Performance	OPP	4	Profil Cible 4				
Quantitative Project Management	QPM	4					
Organizational Innovation and Deployment	OID	5	Profil Cible 5				
Causal Analysis and Resolution	CAR	5					

Figure II-11 : correspondances entre niveaux de maturité et niveaux de capacité du CMMI [Chandelon, 2004]

Sur la Figure II-11, le « profil cible n » correspond au niveau de maturité n. Ainsi, une organisation qui a choisi de mettre en œuvre l'architecture continue du CMMI, et dont les vingt et un premiers secteurs clés de la Figure II-11 sont au niveau de capacité 3, a également atteint le niveau de maturité 3. On retrouve bien l'idée que l'atteinte d'un niveau de maturité équivaut à l'atteinte d'un ensemble de niveaux capacités.

L'équivalence entre les deux architectures pour les niveaux de maturité 4 et 5 est plus subtile. A ces niveaux, il est demandé à l'organisation de choisir quelques secteurs clés « critiques » qu'elle souhaite améliorer, via la mise en œuvre des pratiques clés appartenant aux secteurs clés OPP (Performance du processus de l'organisation) et QPM (Gestion de projet quantitative) pour le niveau de maturité 4, et OID (Innovation de l'organisation et déploiement) et CAR (Innovation de l'organisation et déploiement) pour le niveau 5. Ce choix étant propre à chaque organisation, il n'est donc pas possible de définir des profils génériques pour les deux derniers niveaux de maturité du CMMI.

Nous avons donc vu que, malgré des différences certaines, tant au niveau des buts recherchés qu'au niveau des façons de progresser à travers les niveaux de maturité et de capacité, les deux architectures utilisent, en partie, les mêmes composants (pratiques, objectifs, secteurs clés ...). Ceci est mis en évidence, à travers deux diagrammes de classes en UML, dans la partie suivante.

2.2.3. *Modèle conceptuel en UML du CMMI*

Différents travaux de recherche proposent une modélisation conceptuelle du modèle CMMI. Citons par exemple les travaux du CETIC (Centre d'Excellence en Technologies de l'Information et de la Communication) [Alexandre *et al.*, 2003], qui proposent deux modèles conceptuels : un pour l'architecture discrète et un pour l'architecture continue (**Error! Reference source not found.**). Nous replaçons ces modèles dans l'architecture à quatre niveaux de l'OMG, en fin de chapitre, sur la Figure II-19.

Les parties de ces diagrammes qui nous intéressent sont celles qui mettent en relation les notions de pratiques (spécifiques et génériques), d'objectifs (spécifiques et génériques) et de secteurs clés. Nous proposons la Figure II-12. Sur cette figure, nous privilégions des relations d'agrégation, au lieu de simples associations. De plus, il n'apparaît que les composants du CMMI qui présentent un intérêt direct pour la suite de nos travaux.

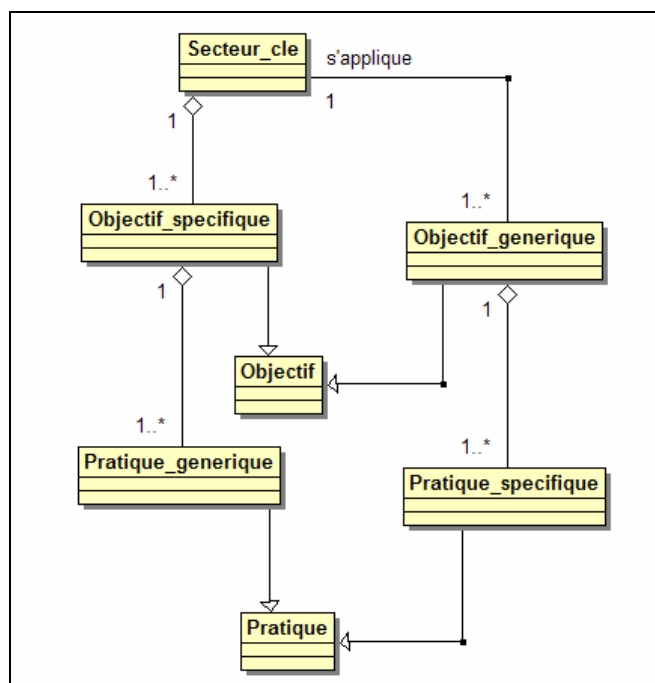


Figure II-12 : noyau structurel du CMMI en UML

La Figure II-12 montre donc que les objets « Pratique_générique » et « Pratique_spécifique » héritent de l'objet « Pratique ». De la même manière, « Objectif_générique » et « Objectif_spécifique » héritent de l'objet « Objectif ». Un secteur clé est composé d'objectifs spécifiques, eux-mêmes composés de pratiques génériques. Un objectif générique est composé de pratiques génériques et s'applique à un secteur clé.

Une étude complémentaire d'autres référentiels en rapport avec la problématique industrielle et donc avec le contexte qualité dans lequel nous évoluons va nous permettre de caractériser plus précisément quelques notions, et particulièrement celle de « pratique ». Il s'agit donc pour nous d'étudier l'ISO 9000 : 2000, l'ISO 20000, ainsi que le référentiel ITIL.

3. La série des normes ISO 9000

3.1. Historique

En 1987, l'ISO publia la série des normes ISO 9000 qui proposent des référentiels d'exigences en matière de gestion et d'assurance de la qualité. Au fil des années, d'autres normes sont venues s'ajouter à la série afin d'en clarifier la compréhension pour les utilisateurs. En 1994, tenant compte de toutes les améliorations, la deuxième édition des normes ISO 9000 paraît et connaît un franc succès. Enfin, une nouvelle version est parue en 2000, l'ISO 9000 : 2000. Cette version introduit la notion de système de management de la qualité et s'articule autour de quatre points majeurs [Mathieu, 2000] et [AFNOR, 2000] :

- **orientation client** : de nouvelles dispositions sont prises telles que :
 - l'identification initiale des attentes, des besoins et des exigences du client,
 - la sensibilisation des acteurs en interne sur ces besoins,
 - un dispositif de communication en externe,
 - un système d'information sur la satisfaction client pour s'assurer que l'entreprise a bien répondu aux besoins exprimés initialement.

- **Approche processus** : elle engage l'entreprise à identifier et à piloter les différentes chaînes d'activités qui contribuent à l'obtention de la satisfaction client et à la création de valeur ajoutée. L'intérêt de cette approche est de permettre aux entreprises d'être davantage agiles et réactives face à un marché turbulent et à des clients de plus en plus exigeants.

- **Composante ressources humaines** : elle est renforcée pour mettre en valeur le rôle essentiel des hommes dans le succès de l'entreprise. Parmi les nouveautés apportées, il faut relever :
 - l'identification des besoins en matière de compétence pour les personnes dont l'activité affecte la conformité du produit,
 - l'évaluation de l'efficacité des formations,
 - une sensibilisation sur la manière dont le personnel contribue à l'atteinte des objectifs qualité.

- **Amélioration continue** : elle devient une exigence en soi. Elle vise à favoriser l'accroissement durable des performances de l'entreprise et l'intégration d'une vraie culture PDCA.

3.2. Composition de l'ISO 9000 : 1994

L'ISO 9000 : 1994 est bâtie autour d'une vingtaine de textes normatifs. Une cartographie de ces normes (Figure II-13) permet de les classer en quatre catégories :

- les « cartes routières » avec en particulier les normes ISO 8402 sur la terminologie et ISO 9000-1 sur les concepts liés à la qualité,
- les recommandations en matière de management de la qualité avec les normes de la série ISO 9004,
- les modèles d'assurance de la qualité. ISO 9001, ISO 9002 et ISO 9003 servent de base à la certification,
- les normes outils de la série ISO 1000x.

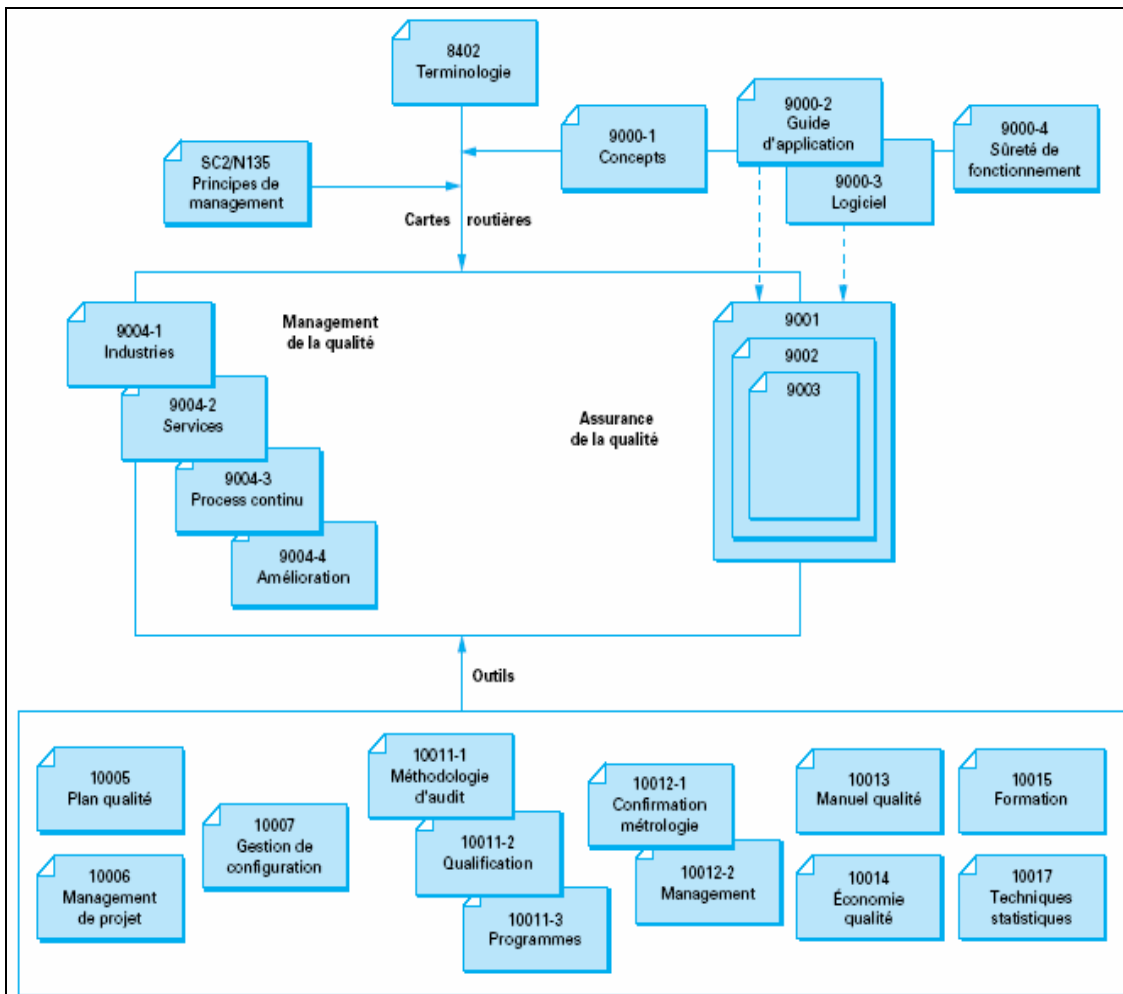


Figure II-13 : cartographie de l'ISO 9000 : 1994 [Mathieu, 2000]

3.3. La composition de l'ISO 9000 : 2000 et le modèle de processus

L'organisation de l'ISO 9000 : 2000 a été grandement simplifiée par rapport à la version de 1994. Il n'y a plus que trois normes principales :

- ISO 9000 : « Systèmes de management de la qualité – Principes essentiels et vocabulaire »,
- ISO 9004 : « Systèmes de management de la qualité – Lignes directrices pour l'amélioration des performances »,
- ISO 9001 : « Systèmes de management de la qualité – Exigences ».

La norme ISO 9000 : 2000 est conçue comme une clé d'entrée permettant de comprendre la finalité des systèmes de management de la qualité et les termes associés. Deux parties principales sont prévues à cet effet : principes essentiels et vocabulaire.

La première vise à expliquer, de façon simple et concise, les principaux concepts introduits dans les normes de systèmes de management de la qualité. La seconde spécifie les différents termes liés au management de la qualité et leurs définitions.

La norme ISO 9000 : 2000 est complétée par une annexe qui présente de manière synthétique les différents concepts utilisés dans la norme. Ce sont des outils graphiques qui permettent de visualiser et de comprendre comment les termes sont reliés entre eux (exemple : Figure II-14).

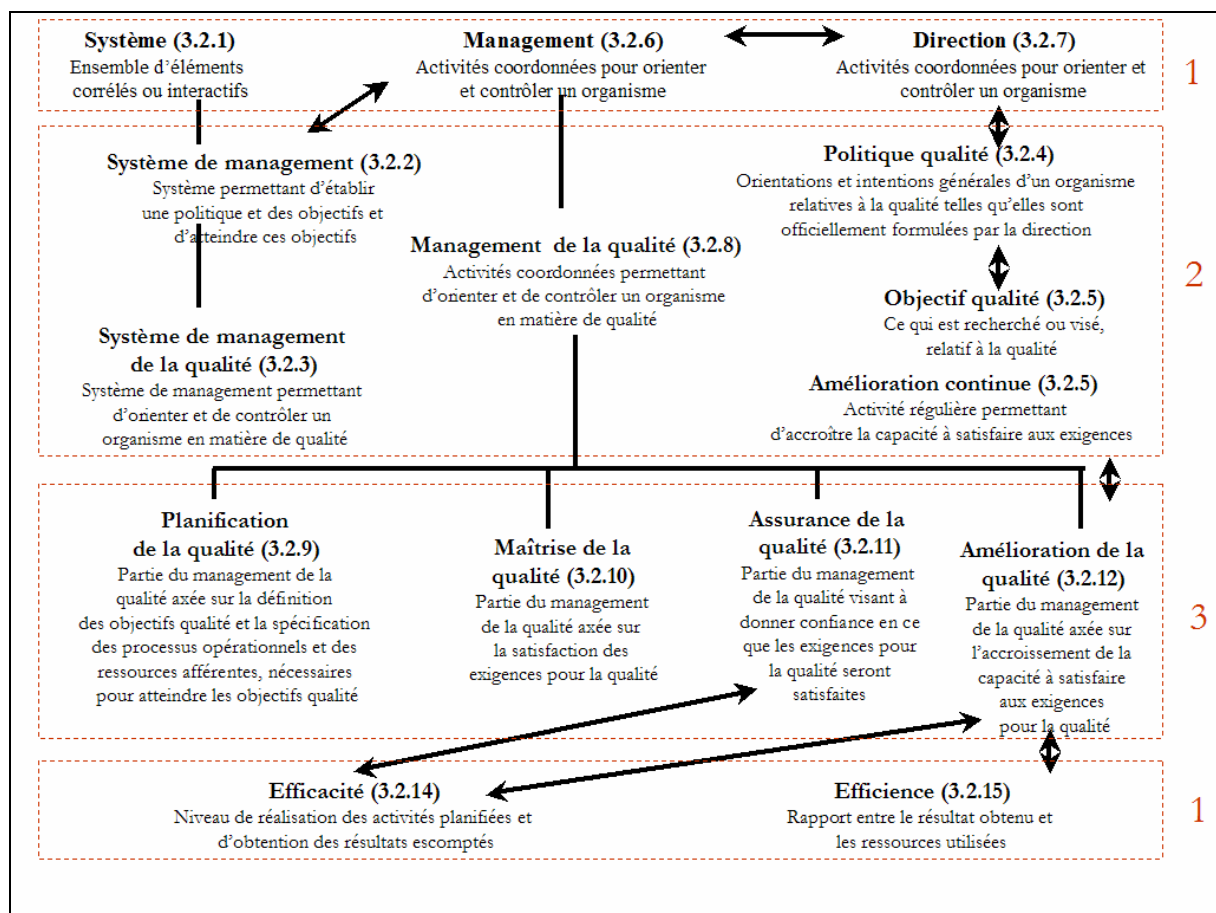


Figure II-14 : exemple d'annexe de la norme ISO 9000 [AFNOR, 2000]

Ce diagramme de concepts est particulièrement intéressant, car il définit les termes clés relatifs à la notion de qualité et utilisés dans la série des normes ISO 9000 : 2000. Ces définitions sont importantes pour la suite, car les textes réglementaires pharmaceutiques les utilisent et nous retrouverons donc souvent les termes présentés.

Les éléments du cadre n° 1 sont les « briques élémentaires » sur lesquelles s'appuient les définitions des cadres suivants. Ce sont des définitions génériques, et donc non spécifiques au domaine de la qualité. Les éléments du cadre n° 2 sont des définitions propres au domaine de la qualité, obtenues à partir des définitions élémentaires du cadre n° 1. Le cadre n° 3 est un « zoom » sur les quatre composantes du « management de la qualité » : planification, maîtrise, assurance et amélioration. Le cadre n° 4 présente deux notions critiques dans la mesure de la performance : l'efficacité (rapport entre le résultat obtenu et le résultat voulu) et l'efficience (rapport entre le résultat obtenu et les ressources utilisées).

La norme ISO 9004 est conçue sous forme de recommandations décrivant de façon complète un système de management de la qualité et les processus le composant. Comme son titre l'indique, elle vise l'amélioration des processus d'une entreprise lui permettant d'accroître sa performance.

La norme se présente sous forme de conseils et de bonnes pratiques destinés uniquement à un usage interne et ne peut donc être utilisée pour la certification.

Cette norme est globale dans le sens où elle couvre l'ensemble des processus contribuant à la satisfaction des clients et des autres parties intéressées :

- le personnel,
- les actionnaires,
- les fournisseurs,
- et la société (pouvoirs publics, collectivités, associations, citoyens...).

Présentée sous forme de spécifications (« ce qu'il faut faire »), **la norme ISO 9001** s'attache essentiellement au management des processus nécessaires pour satisfaire les besoins et exigences du client en assurant la conformité durable du produit.

Les exigences de la norme ISO 9001 ne sont donc pas uniquement centrées sur l'assurance de la qualité visant la conformité d'un produit. Elle inclut également la nécessité pour un organisme de démontrer son aptitude à satisfaire le client.

Outre la satisfaction du client, la norme ISO 9001 présente également, de façon explicite, deux autres objectifs :

- la prise en compte des aspects réglementaires liés aux produits,
- l'amélioration continue de l'efficacité du système de management de la qualité.

La Figure II-15 illustre les relations entre les trois normes que nous venons de présenter. Elle situe également la norme ISO 19011, qui donne les lignes directrices pour l'audit des systèmes de management de la qualité.

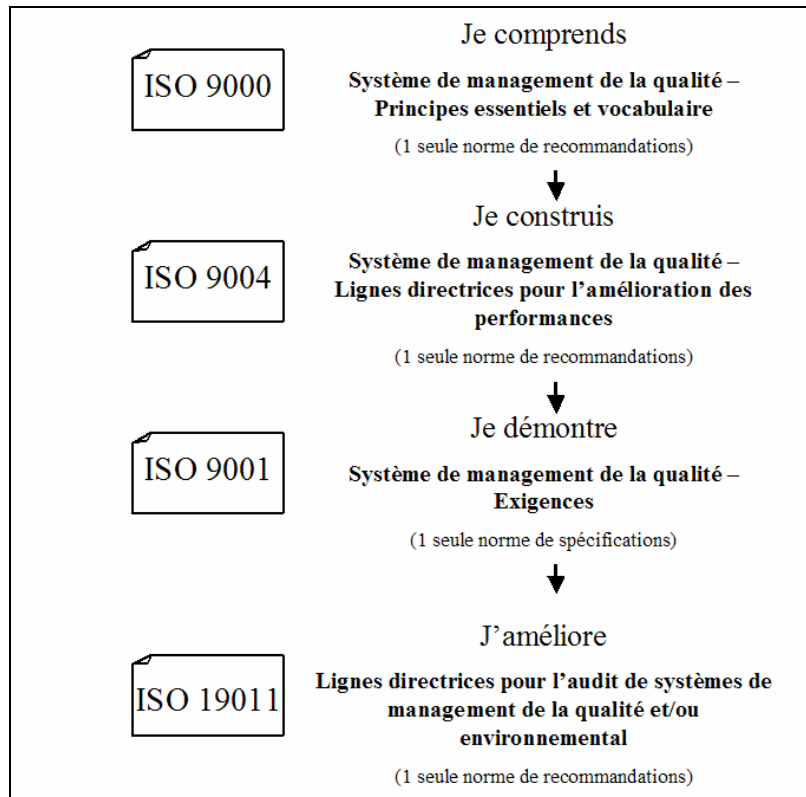


Figure II-15 : l'enchaînement de la série ISO 9000 [Mathieu, 2002]

Ces trois normes s'articulent selon la logique du principe du PDCA (il faut rajouter la norme ISO 19011) :

- « plan » : « Je comprends », il s'agit de la norme ISO 9000, qui permet de prendre en main la norme,
- « do » : « Je construis », il s'agit de la norme ISO 9004, qui permet de démarrer le projet d'amélioration des processus,
- « check » : « Je démontre », il s'agit de la norme ISO 9001, qui propose une liste d'exigences à respecter,
- « act » : « J'améliore », il s'agit de la norme ISO 19011, qui ne fait pas l'objet de plus de détail au sein de ce rapport. Le lecteur intéressé pourra se reporter à [ISO, 2002].

Le couple ISO 9001/9004 est logiquement construit pour permettre l'intégration de différentes exigences dans un système de management de la qualité adapté à chaque entreprise ou organisme.

Nous montrons la structure de la norme ISO 9000 : 2000 sur la Figure II-16.

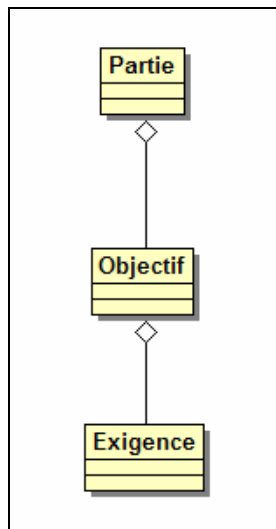


Figure II-16 : modèle conceptuel de l'ISO 9000 : 2000

Ainsi, chaque partie de la norme est composée d'objectifs (relation d'agrégation). Ceux-ci sont décomposés en exigences.

Cette norme est compatible avec tous les secteurs d'activité. Il est donc possible de la « coupler » avec un référentiel de bonnes pratiques métiers. La norme ISO 20000 est construite de cette façon. Elle est dédiée au management de services informatiques. Nous la présenterons dans la partie suivante.

4. Le couple ISO 20000 / ITIL

4.1. L'ISO 20000

L'ISO 20000 intègre donc la vision dynamique d'amélioration proposée par l'ISO 9000 : 2000 et l'applique aux processus d'exploitation de systèmes informatiques.

La norme ISO 20000 est découpée en deux livres [ISO, 2005a] et [ISO, 2005b] :

- le premier définit un ensemble d'exigences minimales et encourage l'adoption d'une approche orientée processus dans le but d'améliorer la prestation de services gérés pour répondre aux besoins des entreprises et des clients,
- le second propose un « code de pratiques en matière de gestion de services » qui distille les points clés des meilleures pratiques de ce domaine. Ce document a pour objectif d'aider les entreprises à mettre en place des processus en vue d'atteindre les objectifs énoncés dans le livre 1.

Les deux livres sont structurés de la même manière, comme le montre le tableau inclus en annexe.

Notons que la partie 6.6 (gestion de la sécurité de l'information) fait explicitement référence à la norme ISO 17799 (dont le nouveau nom est ISO 27002 : 2005) [ISO, 2005c]. Elle définit douze domaines (Articles) de sécurité de l'information constituant au total quarante et une catégories (objectifs de sécurité) au sein desquelles ont été définies cent trente sept mesures de sécurité (contrôles). Chacune des mesures de sécurité permet de s'assurer que les bonnes pratiques communément admises sont mises en œuvre et respectées pour chacune des catégories dans chaque domaine de la sécurité de l'information.

Détaillons le contenu du chapitre 9.2 : « gestion des changements » à titre d'exemple.

Le livre 1 de la norme ISO 20000 précise un objectif pour chaque partie. Par exemple, pour le chapitre 9.2, l'objectif est : « s'assurer que les changements sont évalués, approuvés, implémentés et revus d'une manière contrôlée ». Les objectifs sont ensuite déclinés en plusieurs exigences. Ainsi, le chapitre 9.2 du livre 1 propose neuf exigences, dont les deux suivantes :

- « il doit y avoir des politiques et procédures pour contrôler l'autorisation et l'implémentation des changements d'urgence. »,
- « les demandes de changement doivent être analysées régulièrement afin de détecter les variations des niveaux de criticité des demandes de changements, les demandes récurrentes, les besoins émergents et les autres informations pertinentes. Les résultats et conclusions de ces analyses doivent être enregistrées. »

Concernant ce chapitre dans le second livre de la norme, les bonnes pratiques sont organisées autour de quatre « sous-chapitres » :

- **planification et implémentation des demandes de changement.** Dix neuf bonnes pratiques sont proposées, par exemple « Les informations liées à la planification doivent être disponibles aux personnes affectées par le changement »,
- **clôture et bilan,** qui propose cinq bonnes pratiques, parmi lesquelles « toutes les non-conformités doivent être enregistrées et donner lieu à des actions de résolution »,
- **changements d'urgence,** qui propose deux bonnes pratiques qui concernent la mise à jour de la documentation et la définition de la criticité des demandes de changements,
- **analyse du reporting et actions,** qui propose une seule bonne pratique.

Les bonnes pratiques du livre 2 ne sont donc pas associées « nativement » à une exigence du livre 1. ISO 20000 associe donc à un objectif une liste d'exigences (livre 1) et une liste de bonnes pratiques (livre 2).

Une modélisation conceptuelle simple de l'ISO 20000 est proposée dans la figure suivante (Figure II-17).

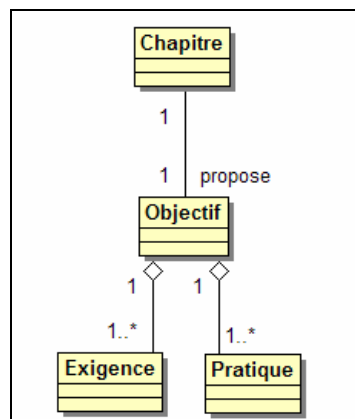


Figure II-17 : modélisation UML de l'ISO 20000

Un chapitre de l'ISO 20000 correspond à un « processus attendu » que l'organisation devra instancier, en se l'appropriant et en l'adaptant par rapport à ses processus réels et à ses contraintes organisationnelles.

Les bonnes pratiques métiers de la norme ISO 20000 sont issues du référentiel ITIL.

4.2. L'ITIL : la facette « métier » de la norme ISO 20000

ITIL est un **référentiel** de bonnes pratiques proposé par l'OGC (Office of Government Commerce) du Royaume-Uni. Il rassemble, dans un ensemble de guides, les meilleures pratiques en matière de management des services informatiques. Il a été développé vers la fin des années quatre-vingt pour mieux gérer l'importance croissante des systèmes informatiques.

ITIL se décompose en modules (Figure II-18), correspondant aux problématiques d'exploitation de systèmes informatisés. Les deux premiers sont considérés comme le cœur de la méthode ITIL [itSMF, 2004] (traduction de [Dumont, 2006]) :

- **Soutien des services informatiques,**
- **Fournitures des services informatiques,**
- Planification de la mise en place de la gestion de services,
- Gestion des infrastructures TIC,
- Gestion des applications,
- Gestion de la sécurité,
- Gestion des actifs logiciels,
- Perspective métier.

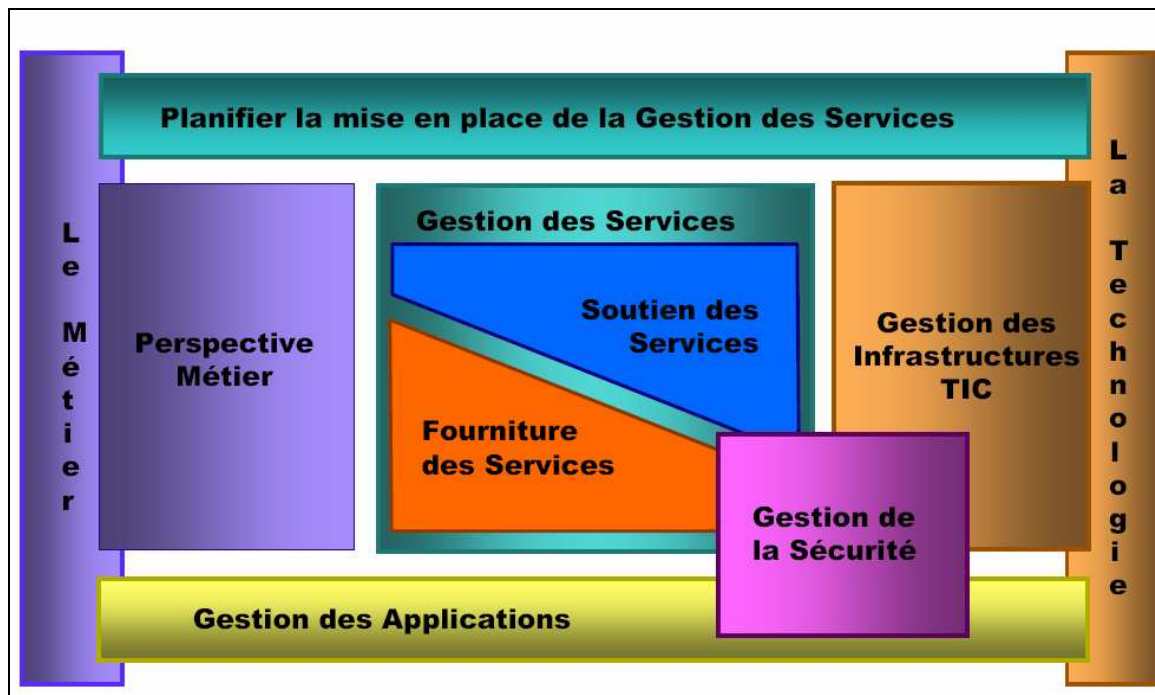


Figure II-18 : les modules ITIL

Le module « Soutien de services informatiques » s'attache au fonctionnement et au support de l'infrastructure technologique. Il est composé des cinq processus suivants :

Processus	Objectif
Gestion des configurations	Gérer l'infrastructure technologique en faisant un état des lieux de l'existant afin de mieux le gérer et le faire évoluer.
Gestion des incidents	Mieux détecter les incidents, améliorer le délai de résolution des incidents selon leur criticité sur le fonctionnement de l'entreprise.
Gestion des problèmes	Mieux gérer les problèmes récurrents et mettre en œuvre des solutions de prévention afin de réduire leur occurrence, voire les supprimer.
Gestion des changements	Mettre en œuvre des démarches de conduite du changement afin d'anticiper les effets de bord.
Gestions des mises en production	S'assurer de l'adéquation du service avec les besoins métiers.

Tableau II-4 : le module « soutien des services informatiques » [Eurotil, 2004]

Le module « Fourniture de services informatiques » est décomposé en cinq processus, qui correspondent globalement à l'interface entre les clients et l'informatique.

Processus	Objectif
Gestion des niveaux de service	Maintenir un certain niveau de qualité de service grâce à des contrats de service renégociés périodiquement.
Gestion des capacités	Vérifier l'adéquation des capacités et performances avec les exigences actuelles et à venir.
Gestion de la continuité de service	Définir et mettre en œuvre des délais contractuels pour la reprise après incident.
Gestion de la disponibilité	Assurer un niveau de disponibilité suffisant à un coût raisonnable.
Gestion financière des services informatiques	Gérer la rentabilité des moyens mis en œuvre pour fournir le service.

Tableau II-5 : le domaine « fourniture de services informatiques » [Eurotil, 2004]

Notons que le périmètre couvert par ITIL est bien plus restreint que celui du CMMI, qui concerne le développement logiciel, les processus de gestion du personnel, l'ingénierie système... ITIL, lui, est spécifiquement dédié à l'exploitation des systèmes informatisés. Actuellement, ITIL et le CMMI sont vus comme étant complémentaires. Le CMMI est plus utilisé dans les phases de conception et de réalisation (acquisition, développement, paramétrage...), alors qu'ITIL est utilisé dans les phases de production et d'utilisation [Nomia, 2005].

Pris tel quel, il ne s'agit donc pas d'un modèle de maturité comme l'est le CMMI. ITIL ne propose pas de « chemin d'amélioration de la maturité ou de la capacité » aux organisations qui souhaitent le mettre en place. ITIL ne propose pas non plus de bonnes pratiques relatives à la gestion des processus métiers : il n'y pas d'équivalent dans ITIL des pratiques génériques du

CMMI. Ceci explique le fait que la norme ISO 20000 ait été conçue à partir du référentiel ITIL et de la norme ISO 9000 : 2000.

Les trois parties précédentes de ce chapitre nous ont donc permis de présenter les buts et structures des référentiels de bonnes pratiques informatiques relatifs à notre périmètre d'application. La partie suivante sera l'occasion de synthétiser les informations que nous venons d'exposer, afin d'aboutir à un modèle conceptuel générique.

5. Synthèse

L'obtention d'un modèle conceptuel générique des référentiels de bonnes pratiques informatiques nécessite le passage par trois « temps forts » :

- le premier va consister à construire les modèles conceptuels des différents référentiels de notre périmètre. C'est ce que nous avons débuté dans les parties précédentes.
- le second aura pour objectif d'identifier les « équivalences » entre composants des différents modèles conceptuels,
- le troisième nous permettra de construire le modèle générique.

L'architecture à quatre niveaux de l'OMG est un outil pertinent pour représenter cette démarche (Figure II-19).

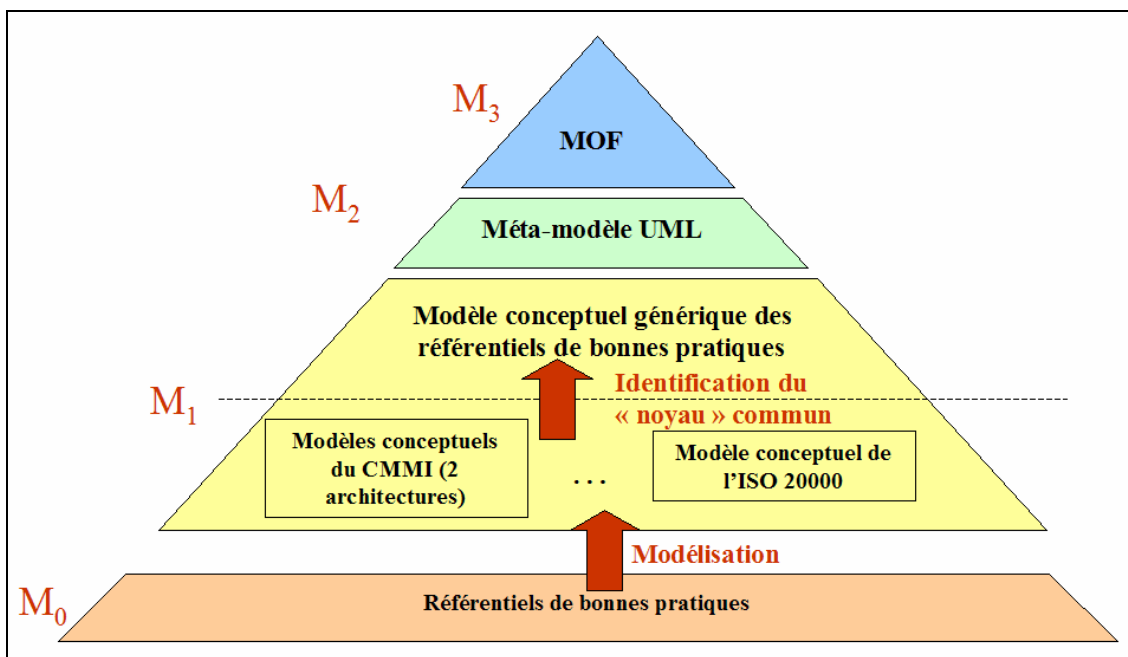


Figure II-19 : positionnement du modèle conceptuel générique des référentiels de bonnes pratiques dans l'architecture de l'OMG

Le niveau M_0 est toujours le « monde réel ». Dans notre problématique, le monde réel est composé des différents référentiels de bonnes pratiques. Le niveau M_1 est découpé en deux sous-niveaux. Le premier est composé des modèles conceptuels des référentiels de bonnes pratiques (c'est-à-dire du monde réel), et il est obtenu à partir d'un processus de modélisation appliqué au CMMI, à l'ISO 9000 : 2000 et à l'ISO 20000. Le second sous-niveau est composé du modèle conceptuel générique recherché, obtenu grâce à l'identification du noyau commun aux différents

modèles conceptuels. Notre choix est d'utiliser le langage UML pour exprimer ce modèle générique, il s'agit donc du niveau M_2 , qui est conforme au MOF (M_3).

Dans le premier chapitre, nous avons précisé que deux définitions étaient envisageables pour le terme méta-modèle. La première, qui définit le méta-modèle comme « le modèle du langage », correspond à la vision que nous proposons en utilisant l'architecture de l'OMG. Au sens de la seconde, c'est-à-dire « modèle de modèle », le modèle conceptuel générique que nous avons obtenu est un méta-modèle.

L'identification des équivalences entre les concepts manipulés par chaque référentiel (second temps fort) est réalisée sur la base d'une analyse rigoureuse des structures et des définitions proposées par chaque référentiel. Le résultat se traduit par la taxonomie présentée dans le Tableau II-6. Il s'agit d'une simple mise en relation syntaxique. Par exemple, nous rapprochons les secteurs clés du CMMI aux parties des normes ISO 20000 et ISO 9000 : 2000, qui correspondent tous, finalement, à un découpage métier des activités.

	Processus attendu	Objectif générique attendu	Objectif spécifique attendu	Bonne pratique générique	Bonne pratique spécifique
CMM	Secteur clé (key process area)	Objectif générique (generic goal)	Objectif spécifique (specific goal)	Pratique générique (generic practice)	Pratique spécifique (specific practice)
CMMI	Secteur clé (process area)	Objectif générique (generic goal)	Objectif spécifique (specific goal)	Pratique générique (generic practice)	Pratique spécifique (specific practice)
ISO 20000	Titre de Partie	Objectif (des parties 3 et 4)	Objectif (des parties 5 à 10)	Pratique (des parties 3 et 4)	Pratique (des parties 5 à 10)
ISO 9000 : 2000	Titre de Partie	Objectif		Exigence	
« Association » dans le « Monde réel »	Processus	Objectif / finalité		Activité métier	

Tableau II-6 : taxonomie des composants des référentiels de bonnes pratiques

Les composants qui nous intéressent sont donc les équivalents « théoriques » des processus, de leurs objectifs (ou finalités) et des activités qui les composent. Rappelons que, selon notre point de vue, les processus et activités mis en œuvre par une organisation sont considérés comme « réels », et les éléments décrits par les référentiels sont du domaine du « théorique » (puisque'ils ne sont pas instanciés dans le monde réel).

Nous ne différencions pas les deux architectures du CMMI, car comme nous l'avons vu, ce sont les mêmes composants qui sont mis en jeu. La structure de la norme ISO 20000 permet de différencier des bonnes pratiques génériques et des bonnes pratiques spécifiques. Les équivalents des bonnes pratiques génériques visent à la reproductibilité des processus (comme dans le CMMI). Il s'agit des pratiques des parties 4 et 5 de la norme. Les équivalents des pratiques spécifiques sont contenus dans les parties cinq à dix. Elles s'intéressent aux activités « métier » de l'exploitation de services informatiques.

L'étude [Liao *et al.*, 2005] démontre que d'autres modèles de maturité mettent en œuvre les mêmes composants que les référentiels que nous venons de présenter. Citons par exemple le modèle SPICE (Software Process Improvement and Capability dEtermination), qui peut être vu comme est un modèle de capacité des processus informatiques (créé par l'ISO en 1995 [ISO, 1995]). Les référentiels de bonnes pratiques, qu'ils soient des modèles de maturité ou des normes, manipulent donc les mêmes concepts, même s'ils ne portent pas le même nom.

Afin de construire le modèle conceptuel du « noyau structurel commun » aux référentiels de bonnes pratiques informatiques, il nous faut maintenant caractériser chaque concept, c'est-à-dire : identifier les attributs que vont posséder les classes du modèle générique en UML.

Par exemple, une bonne pratique spécifique sera caractérisée par :

- un libellé. Il s'agit d'une description « courte » de son objet,
- une description. Il s'agit d'un détail du libellé. Il peut contenir, par exemple, une décomposition en sous-pratiques,
- des produits du travail. Il s'agit d'une liste des produits du travail typiques qu'il est possible d'obtenir en la mettant en œuvre,
- son origine. Il est intéressant de garder l'information du référentiel dont elle est issue,
- le niveau de maturité auquel elle est associée. Dans le cas d'un modèle de maturité, connaître le niveau de maturité auquel est associée l'activité peut être une information importante,
- la ou les phase(s) du cycle de vie applicatif concernée(s). L'introduction du mémoire a proposé un cycle de vie applicatif générique. Certaines activités ne sont relatives qu'à une étape précise du cycle vie et d'autres sont transversales. Nous faisons apparaître cette information ici.
- Le nom du « composant équivalent » dans le référentiel dont elle est issue.

Un référentiel de bonnes pratiques se caractérise, lui, par exemple, par les attributs suivants :

- l'organisme qui en est à l'origine,
- sa date de dernière mise à jour,
- les secteurs d'activité concernés,
- sa version,
- les processus métiers qu'il couvre.

Un référentiel de bonnes pratiques peut ensuite être spécialisé (héritage en langage UML) en :

- un modèle de maturité, qui est défini par une échelle de maturité, par une méthode d'évaluation de la maturité ainsi que par une méthode de progression entre les niveaux de maturité,
- un modèle de capacité, qui est défini par une échelle de capacité, par une méthode d'évaluation de la capacité ainsi que par une méthode de progression entre les niveaux de capacité,
- une norme, qui possède une méthode de certification.

Les autres composants du modèle générique possèdent les mêmes types d'attributs. Finalement, nous proposons le modèle suivant (Figure II-20). A titre illustratif, la Figure II-21 présente quelques instances de classes issues de la Figure II-20

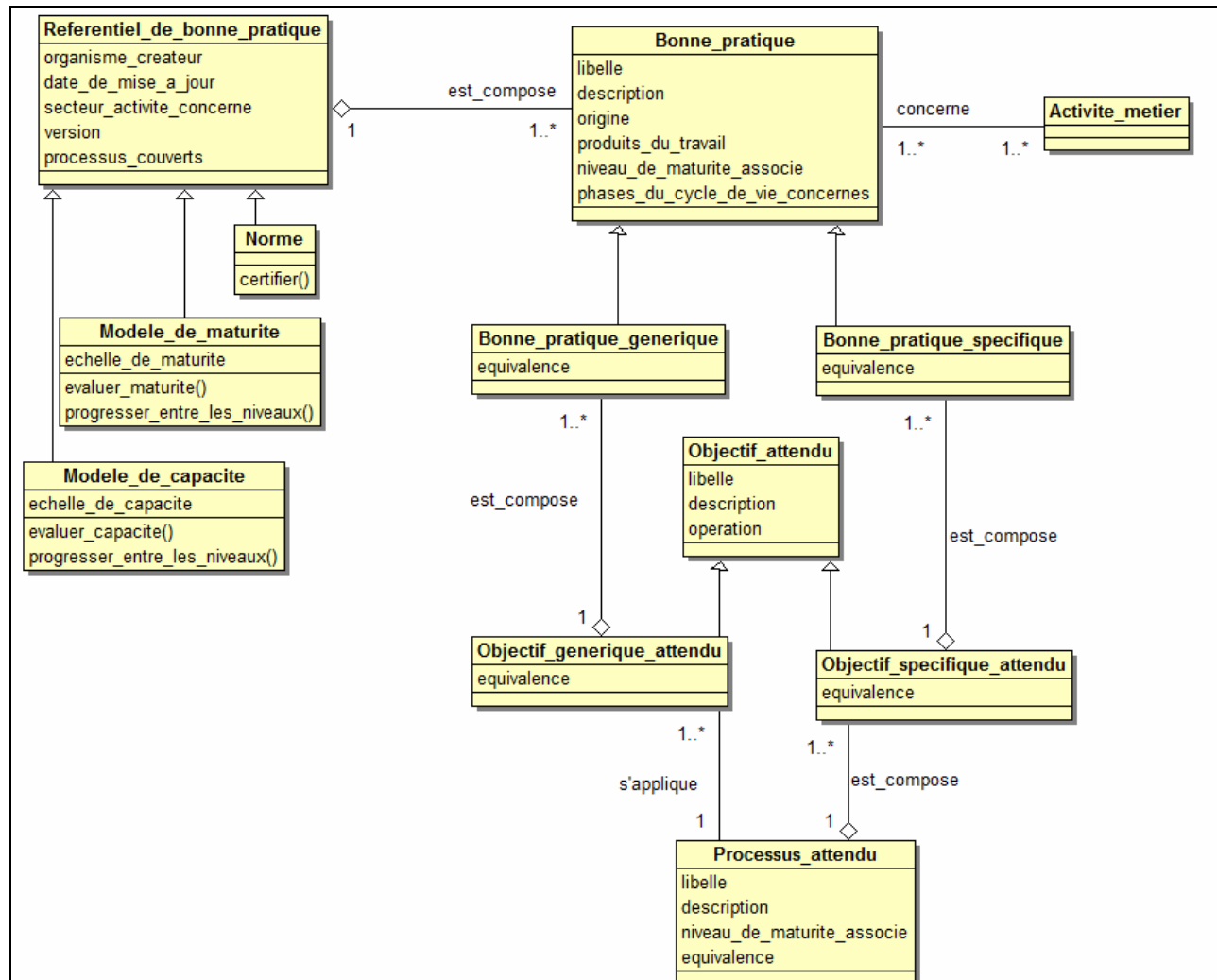


Figure II-20 : modèle conceptuel en UML des référentiels de bonnes pratiques

Classe	Modele_de_maturité
Nom de l'instance	CMMI Architecture Discrète
Date_de_mise_a_jour	2002
Echelle_de_maturite	Echelle à 5 niveaux
Nom	Capability Maturity Model Integration - Staged
Organisme_crateur	Software Engineering Institute
Processus_couverts	Processus logiciel
Secteur_activite_concerne	Conception, réalisation, maintenance des systèmes
Version	1.2

Classe	Processus_attendu
Nom de l'instance	Configuration Management
Description	Le but de ce secteur est d'établir et de maintenir l'intégrité des produits du travail en identifiant les configurations, en les contrôlant...
Equivalence	Secteur clé (Process Area)
Identifiant	CM
Libelle	Gestion de la configuration
Niveau_de_maturite_associe	2

Classe	Objectif_specifique_attendu
Nom de l'instance	Track and control changes
Description	Les changements apportés aux produits du travail sont tracés et contrôlés
Equivalence	Objectif spécifique (Specific Goal)
Identifiant	CM SG 2
Libelle	Suivre et contrôler les changements
Origine	CMMI

Classe	Bonne_pratiques_specifique
Nom de l'instance	Track change requests
Description	1. Initier et enregistrer la demande 2. Analyser l'impact ..
Equivalence	Pratique spécifique (Specific practice)
Identifiant	CM SP 2.1
Libelle	Suivre les demandes de changement
Niveau_de_maturite_associe	2
Origine	CMMI
Phases_du_cycle_de_vie_concernees	Tout le cycle de vie
Produits_du_travail	Demandes de changement

Figure II-21 : exemples d'instances d'objets

La Figure II-21 présente quelques exemples d'objets obtenus après l'instanciation des classes « Modèle_de_maturité » (exemple de l'architecture discrète du CMMI), « Processus attendu » (exemple du secteur clé « gestion de la configuration » du CMMI, « objectif spécifique attendu » (exemple du second objectif spécifique du secteur clé gestion de configuration », et enfin « Bonne_pratique_specifique » (exemple d'une des pratiques relatives à la gestion de configuration proposée par le CMMI).

Nous avons donc obtenu le modèle générique des référentiels de bonnes pratiques informatiques que nous recherchions.

6. Conclusion

En quelques années, la qualité est passée d'une vision descriptive (démarche type PDCA), à une vision normative, puis prescriptive. Pour apporter des réponses adéquates à chaque domaine d'activités, il existe de multiples référentiels de bonnes pratiques. Ils diffèrent de par leur domaine d'application, leur provenance, la granularité de l'information qu'ils proposent, leur structure... Notre problématique industrielle nous pousse à nous intéresser spécifiquement à certains d'entre eux : il s'agit de ceux présentés dans ce chapitre. Outre ces présentations, ce chapitre nous a permis :

- de définir la notion de bonnes pratiques, en nous appuyant sur les différents référentiels étudiés,
- d'identifier les caractéristiques de ces dernières,
- d'obtenir un modèle conceptuel général des référentiels de bonnes pratiques, en nous appuyant sur une taxonomie de leurs composants.

Nous n'affirmons cependant pas que ce modèle est valide pour tous les référentiels existants. Il l'est seulement pour nos référentiels, et c'est le résultat que nous recherchions. Il ne peut donc pas s'agir d'une première esquisse d'une ontologie des référentiels de bonnes pratiques informatiques.

Nous disposons avec lui de la première brique du mapping. Le chapitre suivant va nous permettre d'obtenir la brique suivante : le modèle conceptuel des exigences réglementaires.

Chapitre III. EXIGENCES RÉGLEMENTAIRES PHARMACEUTIQUES APPLICABLES AUX SYSTÈMES INFORMATISÉS

Nous avons vu dans l'introduction qu'un médicament est un produit actif, qui peut comporter des **risques** pour la santé des consommateurs. Son histoire est d'ailleurs jalonnée d'accidents graves. Sur les cinquante dernières années, certaines affaires ont été particulièrement retentissantes [Studer, 2006], [Chast, 1995], [Fouassier, 2006], [Degroote *et al.*, 2003], [Chauveau, 2004] :

- l'affaire de la « **Poudre Baumol** » (1952). Cette poudre était destinée aux soins du nourrisson. Un lot avait reçu de l'anhydride arsénieux au lieu d'un sel de sodium inoffensif, ce qui a entraîné le décès de soixante-treize nouveaux-nés et provoqué près de trois cents accidents. L'absence de contrôle sur les matières premières a été reconnue comme étant la cause de l'accident,
- l'affaire du **Stalinon** (1952). Le Stalinon était un médicament destiné à la furonculose résistante aux antibiotiques. Son utilisation a provoqué cent cinq décès et a laissé cent cinquante rescapés invalides. Le procès, à l'automne 1957, s'achève par la condamnation du pharmacien exploitant le Stalinon et celle de l'entreprise de façonnage qui préparait la spécialité. Mais d'autres éléments apparaissent au travers du procès. Certains comprimés soumis à expertise avant l'accident se sont révélés non conformes à la formule déclarée. Le principe actif du médicament est, en outre, un composé instable dont le caractère dangereux est connu. Autrement dit, le visa a été délivré alors que le médicament n'était pas sûr,
- l'affaire de la **Thalidomide** (1957). Il s'agit d'un médicament vendu comme hypnotique et chez les femmes enceintes comme antiémétique pour combattre les nausées matinales et d'autres symptômes. Il est responsable de dix mille malformations très graves de nouveaux-nés. L'insuffisance des essais précliniques a été identifiée comme étant une des causes de ce désastre,
- l'affaire du **Talc Morhange** (1973). L'enquête révèle qu'une erreur de manipulation a conduit à mélanger trente huit kilos d'hexachlorophène, produit hautement toxique, à six cents kilos de talc. Plus de deux cents enfants sont atteints (mort, coma ou séquelles neurologiques).

La réglementation pharmaceutique a pour but d'éviter que de telles tragédies ne se reproduisent. Chaque affaire correspond à une évolution profonde des textes et/ou des périmètres réglementaires. Ainsi, les drames liés à la poudre Baumol et au Stalinon sont à l'origine de

« l'institution d'un contrôle continu sur les matières premières destinées à la fabrication de produits pharmaceutiques et cosmétiques ». L'affaire de la Thalidomide a eu pour conséquence la création des Centres Nationaux de Pharmacovigilance. Aux USA, les amendements Kefauver-Harris ont imposé aux fabricants de démontrer l'efficacité (et non plus seulement la sécurité d'emploi) des médicaments avant leur mise sur le marché. Depuis l'affaire du talc Morhange, une loi réglemente la fabrication des produits cosmétiques et rend obligatoires l'étiquetage et la traçabilité des lots de produits.

Les molécules mises en jeu lors de production d'un produit actif sont toujours plus complexes, et la caractérisation sans équivoque de leur activité et de leurs effets ne s'en trouve pas facilitée. Il y a des risques réels liés à la conception, aux tests, à la production, à la distribution et à l'utilisation des médicaments. Ces risques doivent être identifiés et maîtrisés par les laboratoires pharmaceutiques. La réglementation pharmaceutique impose à ces derniers le respect de certaines méthodes, l'atteinte d'un niveau de qualité et de maîtrise de leurs processus « à risques », des procédures de contrôle... Quelques instances réglementaires ont pour rôle de superviser les laboratoires pharmaceutiques dans leurs tâches. Les exigences imposées par la réglementation sont décrites dans des textes réglementaires, qui ont pour but, par exemple, de garantir que « les produits sont fabriqués et contrôlés de façon uniforme et selon des normes de qualité adaptées à leur utilisation et spécifiées dans l'Autorisation de Mise sur le Marché » [OMS, 1997].

De plus, c'est une évidence de dire que l'informatique prend de plus en plus d'importance dans toutes les industries, et ce, à une vitesse redoutable ! L'industrie pharmaceutique ne fait pas exception. Les systèmes de gestion de données, les outils de travail collaboratif, les systèmes de gestion documentaires et de publication, les applications Internet et Intranet, les systèmes de calcul, de résolution de problèmes, les outils utilisés en laboratoire, les outils de contrôle de processus de production, *etc.*, sont légion [Basavapathruni, 2007]. Il n'est donc pas étonnant que les textes réglementaires s'intéressent aux cas des « systèmes informatisés » utilisés dans les différentes phases du cycle de vie du médicament.

L'objectif de ce chapitre est d'obtenir un modèle conceptuel des exigences réglementaires pharmaceutiques. Afin d'y arriver, nous proposons quatre parties, qui vont successivement nous permettre de :

- présenter les principales instances réglementaires liées à notre périmètre, défini par notre contexte industriel et les textes réglementaires qui nous intéressent,
- définir la notion de système informatisé dans l'industrie pharmaceutique,
- détailler les principales exigences réglementaires qui s'appliquent aux systèmes informatisés,
- obtenir un référentiel d'exigences, structuré et cohérent, ainsi qu'un modèle conceptuel de ce référentiel.

1. Les autorités réglementaires

Les réglementations sont souvent synonymes de complexité. Ceci est particulièrement vrai dans l'industrie pharmaceutique, dont les productions sont en général destinées à plusieurs pays, chacun ayant ses propres réglementations. Il est donc nécessaire de présenter les réglementations qui s'appliquent dans les zones géographiques incluses dans le périmètre d'étude de la thèse. Dans cette partie, nous nous intéressons aux organismes réglementaires des Etats-Unis, d'Europe et de France.

1.1. L'autorité américaine : la Food and Drug Administration (FDA)

La FDA est l'agence gouvernementale des Etats Unis, responsable de la sécurité sanitaire des produits alimentaires, ainsi que du contrôle et de la réglementation des médicaments. Elle fait partie de la branche exécutive du gouvernement américain. Elle opère depuis 1979 dans le cadre du Département de la santé publique (PHS), lui-même intégré au « Département of Health and Human Services ». Le mandat de l'agence est celui de la **protection du consommateur dans le cadre du commerce inter-étatique** [<http://www.fda.gov>], [Frkic, 2000].

Son statut d'agence réglementaire scientifique la charge plus précisément de la sécurité des produits nationaux et des produits importés dans plusieurs domaines, notamment : les produits alimentaires, cosmétiques, vétérinaires, radiologiques, médicaments et les appareils médicaux. La FDA en certifie la sécurité, l'efficacité et la conformité de leur étiquetage [Williams, 2007] et [Manasse, 2007].

Au niveau réglementaire, les cinq missions principales de la FDA sont [Frkic, 2000] :

- la mise en place des règles afin d'assurer la protection du consommateur (approbation préalable à la commercialisation, réglementation, contrôle...),
- la dissémination de l'information sur la réglementation en vigueur auprès de l'industrie et du public afin d'assurer une meilleure protection du consommateur,
- la surveillance de l'industrie dans son obligation de conformité avec la loi sur la protection sanitaire,
- le soutien scientifique nécessaire afin d'assurer la mise à jour de la réglementation,
- la mise en œuvre de la réglementation relative à la protection sanitaire.

Tout industriel qui souhaite commercialiser un produit alimentaire ou pharmaceutique sur le marché américain doit respecter les réglementations américaines. Les industriels français qui exportent aux Etats-Unis doivent donc s'y conformer. C'est le cas pour le groupe Pierre Fabre.

La réglementation américaine est décrite dans le CFR (Code of Federal Regulations) : le code des réglementations fédérales. Il est divisé en cinquante titres qui représentent les secteurs sujets au

règlement fédéral. Les réglementations de la FDA se trouvent dans le Titre 21, noté « 21 CFR ». Chaque titre est divisé en chapitres, eux-mêmes subdivisés en pièces couvrant des domaines de normalisation spécifiques. Les chapitres sont regroupés dans neuf volumes différents.

Les chapitres relatifs à notre périmètre sont (cf. Figure III-1) :

- le **21 CFR 58**, qui s'applique à la recherche en laboratoire,
- le **21 CFR 312**, qui s'applique aux études cliniques,
- les **21 CFR 210 et 211 (current Good Manufacturing Practices)**, qui décrivent les attendus en matière de production de médicaments,
- **le 21 CFR 11**, qui regroupe les exigences qui permettent d'assurer l'authenticité, l'intégrité et si nécessaire la confidentialité des enregistrements électroniques. Il régleme également l'utilisation de la signature électronique.

Présenter de manière exhaustive le contenu de ces textes serait utopique : le périmètre couvert est très large et très technique. Nous détaillerons dans la partie 3 de ce chapitre les paragraphes qui nous concernent.

Présentons maintenant synthétiquement les autorités réglementaires relatives à la zone européenne et à la France.

1.2. L'autorité européenne : European Medicines Agency (EMA)¹ et l'autorité française : Agence Française de Sécurité Sanitaire des Produits de Santé (AFSSAPS)

L'EMA et l'AFSSAPS sont les autorités qui réglementent l'industrie pharmaceutique respectivement au niveau européen (EMA) et français (AFSSAPS).

L'EMA a été instituée par le règlement CEE n° 2309/93 du Conseil du 22 juillet 1993, puis officiellement créée en 1995. Elle est installée à Londres.

L'EMA a trois tâches principales :

- la coordination de l'évaluation scientifique des médicaments dans le cadre de la procédure d'AMM centralisée² et la responsabilité de l'arbitrage dans le cadre de la procédure de reconnaissance mutuelle,

¹ Les informations sur l'EMA sont, pour la plupart, directement issues de : <http://www.senat.fr/rap/r05-382/r05-3821.html>

² « La procédure centralisée est une procédure « monolithique » dans laquelle un dossier est soumis directement à l'EMA et analysé par le CSP, la Commission d'AMM de l'EMA. Ce Comité émet un avis qui est transmis à la Commission européenne de Bruxelles. Cette Commission délivre finalement l'AMM sur avis de l'EMA. Ce processus est apparemment rapide puisque l'accord du CSP doit être donné en 210 jours (7 mois). En revanche, le processus décisionnel de la Commission européenne, de nature administrative, est parfois assez long. La procédure

- la coordination des activités liées à la pharmacovigilance à l'échelon européen,
- la coordination des activités d'inspection, notamment celles en rapport avec la vérification du respect des Bonnes Pratiques de Fabrication (BPF), des Bonnes Pratiques de Laboratoire (BPL), et des Bonnes Pratiques Cliniques (BPC).

En outre, le règlement n° 2309/93 lui confère une mission de conseil scientifique dans la conduite des tests et des essais thérapeutiques. L'agence publie également des directives sur les exigences en matière de tests de qualité, de sécurité et d'efficacité.

En général, chaque pays dispose d'une autorité réglementaire. De nombreux exemples sont fournis dans [Wertheimer *et al.*, 2007]. Par exemple, le marché français est ainsi régulé par l'AFSSAPS.

La réglementation imposée par l'EMA et l'AFSSAPS relative à notre périmètre est composée de quatre ensembles de textes :

- les **Bonnes Pratiques de Laboratoires** (BPL ou GLP) : ayant pour application initiale le contrôle des produits chimiques, les GLP ont été mises en place pour promouvoir la qualité et la validité des données d'essai servant à établir la sûreté des produits chimiques. Ce guide a été produit par l'Organisation de Coopération et de Développement Economique¹ (OCDE),
- les **Bonnes Pratiques Cliniques** (BPC ou GCP) : axé sur la qualité, l'éthique et les aspects scientifiques, ce référentiel s'applique à la conception et à la réalisation d'essais auxquels participent des sujets humains. La conformité aux BPC apporte au public la garantie que les droits, le bien-être, et la confidentialité des sujets d'essai sont protégés et que les données d'essai sont crédibles,
- les **Bonnes pratiques de Fabrication** (BPF ou GMP) : les BPF sont l'élément de l'assurance de la qualité qui garantit que les médicaments sont fabriqués et contrôlés de façon cohérente. Le guide des Bonnes Pratiques de Fabrication définit une liste

centralisée aboutit à une AMM unique, valable le même jour dans l'ensemble des États membres. Cependant, la commercialisation dans l'ensemble des États membres dépend de la deuxième étape d'évaluation des médicaments, l'obtention d'un prix de remboursement qui ne peut se faire que selon des procédures nationales » [Colin, 2001]. La procédure de reconnaissance mutuelle « est d'une grande souplesse, permettant de ne choisir que quelques États membres. La procédure elle-même est très rapide (quatre vingt dix jours) et adaptée aux médicaments déjà enregistrés dans un État membre. En revanche, le manque d'harmonisation entraîne la réception d'une longue liste de questions additionnelles de la part des États membres auxquelles le laboratoire doit impérativement répondre. La phase de rédaction des AMM nationales se fait selon un timing aléatoire » [Colin, 2001].

¹ Organisme international pluridisciplinaire qui offre à ses trente pays membres une structure et une tribune pour se consulter et pour collaborer entre eux afin d'élaborer et de perfectionner les politiques économiques et sociales. Contrairement à d'autres organismes internationaux, l'OCDE n'établit pas de règles ni de règlements pour résoudre les conflits, mais elle encourage la négociation d'accords et la promotion de codes juridiques dans certains secteurs. Son action se traduit par la conclusion d'accords ayant force obligatoire ou non en ce qui concerne les mesures à prendre par les pays membres. L'OCDE est surtout connue pour ses publications et ses statistiques.

d'exigences, qui servent de base pour les inspections auprès des fabricants de médicaments dans toute l'Union Européenne,

- les **Bonnes pratiques de Distribution** (BPD ou GDP) : ce référentiel a été établi afin de maintenir le niveau de qualité dans tout le réseau de distribution pharmaceutique, de sorte que la qualité des médicaments soit préservée tout au long de la chaîne de distribution : du producteur aux grossistes, puis aux pharmaciens, pour finalement être mis en vente au grand public.

Que ce soit au niveau européen ou américain, les textes sont donc organisés par phase du cycle de vie du médicament. Nous replaçons ces textes par rapport à un cycle de vie simplifié sur la Figure III-1 (rappelons qu'un cycle de vie plus exhaustif a fait l'objet de la Figure 0-1).

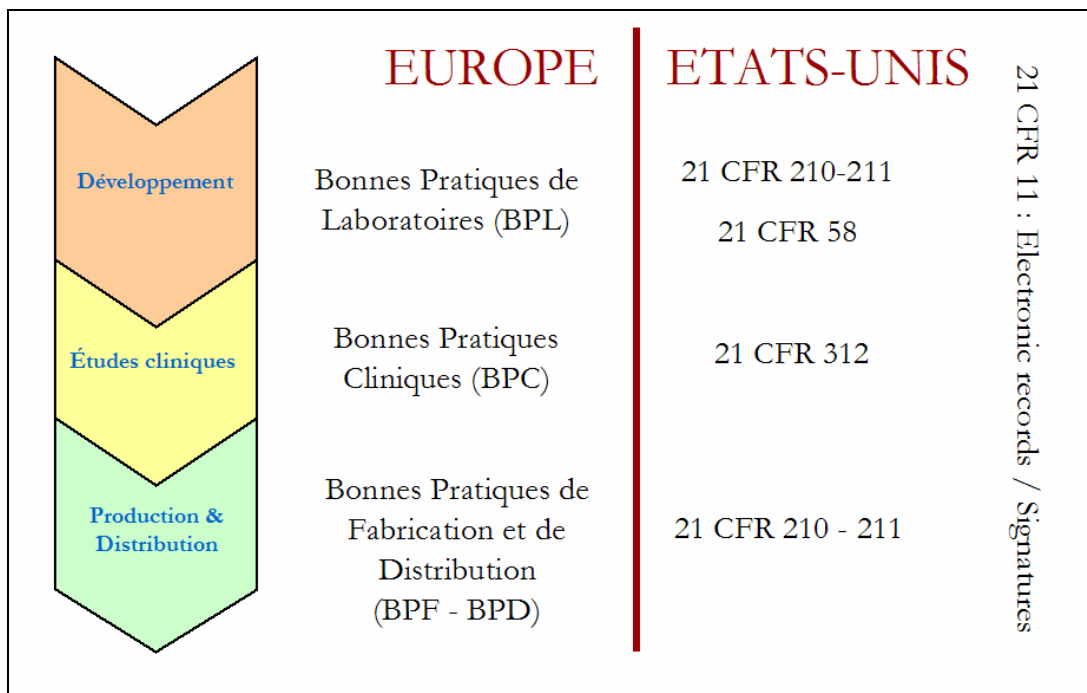


Figure III-1 : cycle de vie simplifié du médicament et textes réglementaires

Le 21 CFR 11 est un document « transverse », puisqu'il peut s'appliquer à toutes les phases du cycle de vie. Ceci explique son positionnement particulier sur la figure précédente.

Nous avons vu que ces textes couvrent un périmètre large, qui inclut les systèmes informatisés. Ceux-ci sont au cœur de notre problématique. Dans le contexte pharmaceutique, qu'est ce qu'un système informatisé ? Est-il possible de rapprocher cette notion de celle de systèmes d'information ? Nous étudions ces questions dans la partie suivante.

2. Périmètre

Par système informatisé, nous entendons l'ensemble [SOFAQ, 1991] comprenant :

- la source d'information (les données qui seront traitées par le programme, et donc également leurs supports de stockage par exemple),
- les connexions (alimentation, réseau...),
- le matériel (serveurs, baies...),
- le système d'exploitation,
- les outils progiciels (environnement de travail ou de développement),
- les programmes (partie visible de l'ensemble par l'utilisateur. Il matérialise l'implémentation des spécifications).

Il s'agit du « système informatique » décrit dans [Morley *et al.*, 2005].

Il faut rajouter à ces éléments le « contexte » dans lequel les systèmes vont être amenés à être utilisés. Il s'agit de l'environnement, des utilisateurs et de la documentation (technique, utilisateurs, formation, procédures opératoires...).

Ce contexte fait donc plutôt référence au système de traitement de l'information, qui complète le système informatique pour donner naissance au concept de système d'information.

3. La réglementation pharmaceutique applicable aux systèmes informatisés

3.1. Textes américains

Nous avons vu que c'est le Titre 21 des CFR qui réglemente l'industrie pharmaceutique. Nous allons maintenant donner quelques détails sur les chapitres qui nous intéressent [Joly, 2001], [Hoffmann *et al.*, 1998] :

- le **21 CFR 58** présente trois paragraphes qui s'appliquent aux systèmes informatisés
 - § 58.61 : « Equipment design »,
 - § 58.63 : « Maintenance and calibration of equipment »,
 - § 58.130 : « Conduct of a non-clinical laboratory study ».

Les deux premiers évoquent de manière indirecte la nécessité d'utiliser des systèmes sous contrôle et donc ayant fait l'objet d'une validation pharmaceutique¹, tandis que le dernier insiste sans ambiguïté sur la nécessité, dans le cas des données électroniques, de pouvoir identifier la personne ayant effectué la saisie, ainsi que de pouvoir tracer toutes les modifications apportées à des données.

- le **21 CFR 312**, qui s'applique aux études cliniques. Il n'y a pas de mention particulière concernant les systèmes informatisés, mais un article à lui seul induit la nécessité de respecter les règles relatives à ces systèmes et regroupées dans d'autres documents. Il s'agit du § 312.45, qui mentionne simplement la nécessité de se reporter aux guides applicables.
- les **21 CFR 210 et 211** (*current Good Manufacturing Practices*), et plus particulièrement le paragraphe 211.68 « Automatic, mechanical, and electronic equipment », qui énumère les principales problématiques en lien avec les systèmes informatisés : validation, gestion des changements, contrôle, archivage et restauration des données critiques...
- le **21 CFR 11** définit les critères selon lesquels les dossiers et signatures électroniques seront considérés comme équivalents à des dossiers sur support papier et des signatures manuscrites. La réglementation s'applique aux dossiers sous forme électronique, qui sont créés, modifiés, maintenus, archivés, récupérés ou transmis dans le cadre des exigences liées à tout dossier décrit dans la réglementation de la FDA. Cette réglementation implique donc des contrôles et actions en termes de sécurité, de privilèges, de signatures électroniques et d'audit trail.

Outre ces textes, la FDA propose de nombreux guides méthodologiques, qui illustrent les attendus de la réglementation par rapport aux systèmes informatisés : « *Guide to Inspection of Computerized Systems in Drug Processing* », « *Guidance for Industry: General Principles of Software*

¹ La « validation pharmaceutique » est une activité d'ingénierie spécifique au monde pharmaceutique. C'est un sujet vaste et complexe présenté dans l'Annexe 0.

Validation», « *Guidance for Industry : Computerized systems Used in Clinical Trials* », « *Guidance for Industry : 21 CFR Part 11 ; Electronic Records ; Electronic Signatures Validation (Draft)* ».

Les attendus au niveau européen sont tout à fait comparables, bien que présentés différemment. La partie suivante est consacrée à l'applicabilité de ces textes aux systèmes informatisés.

3.2. Textes européens

3.2.1. Les Bonnes Pratiques de Laboratoires

En plus du guide des BPL que nous avons déjà présenté, l'OCDE a également produit un document nommé « *Document consensus sur les BPL – Application des BPL aux systèmes informatiques* ». Ce document interprète et détaille les BPL relatives aux systèmes informatisés.

La Figure III-2 présente très synthétiquement l'information contenue dans ce document.

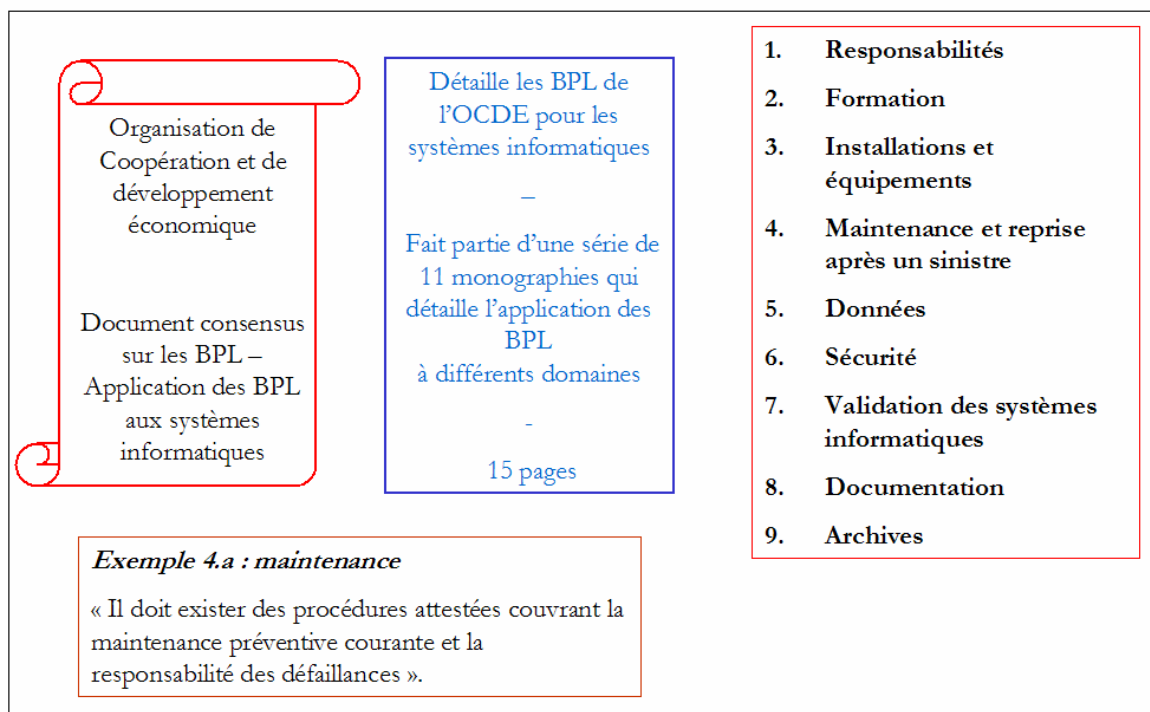


Figure III-2 : les BPL applicables aux systèmes informatisés

Ainsi, le « *Document consensus sur les BPL – Application des BPL aux systèmes informatiques* » aborde neuf points, parmi lesquels : la formation des ressources qui vont utiliser, concevoir ou exploiter les systèmes informatisés, la sécurité des données qui « circulent » à travers les systèmes informatisés, la gestion de la documentation... (cf. liste de droite dans la Figure III-2).

L'article [Dent, 2007] fournira aux lecteurs souhaitant approfondir le sujet de plus amples informations sur les BPL.

3.2.2. Les Bonnes Pratiques Cliniques

Les éléments relatifs aux systèmes informatisés sont regroupés dans l'article 5.5 (en anglais) :

- **5.5.** : « Trial management, data handling and record keeping ». Dans ce paragraphe, les problématiques de conservation et d'enregistrement des informations critiques sont abordées.
- **5.5.3** : « Validation, audit trail, backup ». Ce paragraphe s'intéresse aux mécanismes de validation (annexe) des systèmes informatisés ainsi qu'à la traçabilité des opérations effectuées sur et par un système informatisé (audit trail).

Pour information, une analyse détaillée de ce texte est proposée par l'article [Guarino, 2007].

3.2.3. Les Bonnes Pratiques de Fabrication

Le guide des BPF est divisé en chapitres, chacun d'entre eux reposant sur un principe, qui est développé sur les aspects essentiels à prendre en considération pour son application. Le guide présente également des annexes qui correspondent à des domaines d'activités spécifiques, dont l'annexe 11, intitulée « Systèmes informatisés » (Figure III-3).

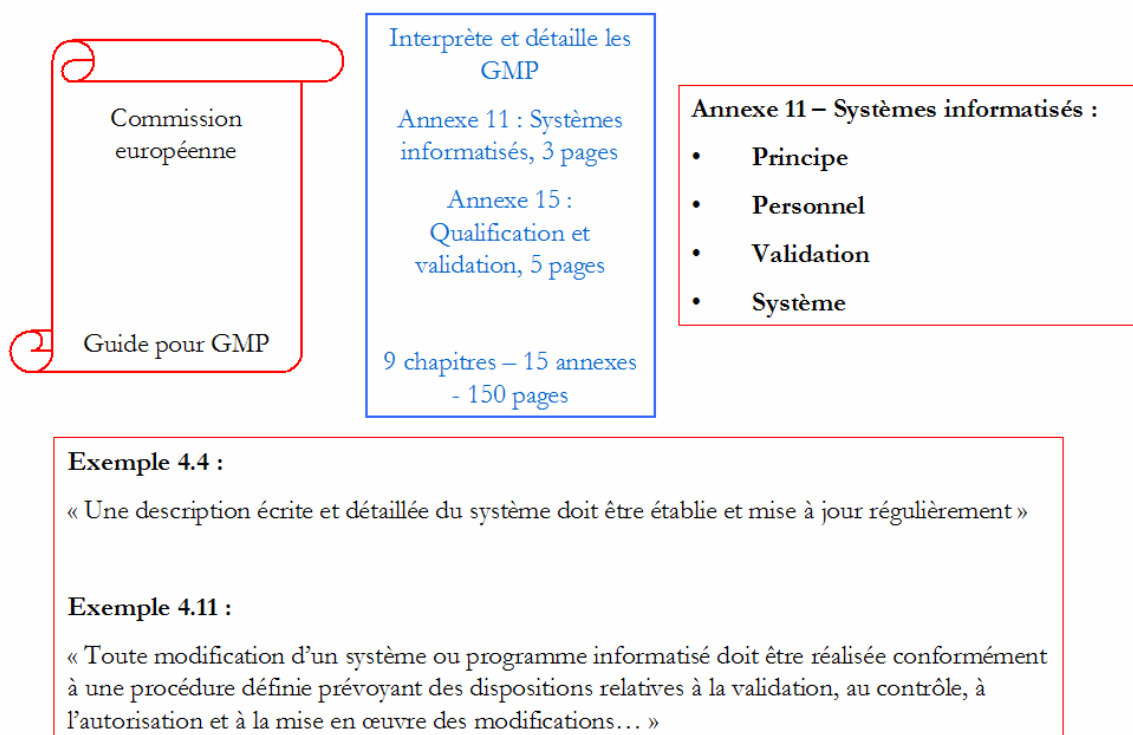


Figure III-3 : les BPF applicables aux systèmes informatisés (Annexe 11)

Ainsi, l'annexe 11 aborde quatre points critiques relatifs aux systèmes informatisés : Principe, Personnel, Validation et Système. La partie « système » est la plus vaste : elle couvre aussi bien la documentation que des fonctionnalités précises ou encore la gestion des changements. Il s'agit du document le plus détaillé de la réglementation concernant les systèmes informatisés (liste de droite sur la Figure III-3).

Pour information, l'article [Berry, 2007] propose une analyse plus fine de ce texte.

3.2.4. Les Bonnes Pratiques de Distribution

Le thème des systèmes informatisés est abordé superficiellement dans le chapitre III des BPD. Huit points sont décrits, parmi lesquels : les principes de validation, la nécessité de posséder une description écrite et détaillée des systèmes, la sécurité des données, la gestion des modifications. Notons qu'il s'agit simplement d'une simplification des attendus des BPF.

A ces textes, il faut rajouter deux ouvrages produits par une association : International Society for Pharmaceutical Engineering.

3.3. Le GAMP et le Volume 5

L'International Society for Pharmaceutical Engineering (ISPE) est une association regroupant plus de 6 000 professionnels (1 400 en Europe) travaillant dans le domaine du génie pharmaceutique, dans des laboratoires, des sociétés fabriquant des équipements et des prestataires de service de l'industrie pharmaceutique. Les activités de l'ISPE consistent à développer le savoir-faire de ses membres à travers des séminaires, des congrès et la publication d'un journal bimestriel.

Cette association a réalisé deux ouvrages qui nous intéressent particulièrement :

- le **GAMP** : il propose une méthode pour la validation des systèmes automatiques (dont les systèmes informatisés). Outre cette méthode, le GAMP propose une liste de bonnes pratiques concernant la gestion de la documentation et la réalisation des tests (nous le référençons par : [ISPE, 2001a]),
- le guide « **Commissioning and Qualification** », qui s'intéresse à la validation des systèmes, du point de vue « ingénierie » [ISPE, 2001b]. Il propose une liste de bonnes pratiques d'ingénierie informatique,

Nous les avons intégrés au périmètre réglementaire, car ce sont des ouvrages de référence, très largement utilisés, qui permettent de comprendre certaines exigences des textes présentés précédemment, en les traduisant en termes « métier ».

L'analyse de l'ensemble des textes montre à la fois une communauté de perception des problèmes et une certaine analogie entre les différents acteurs, mais qui ne va pas jusqu'à un unique document de référence. La diversité reste un handicap pour l'industriel ayant une surface internationale. Cela nous amène à nous poser une question : comment, à partir des textes, obtenir une liste d'exigences applicables aux DSI, c'est-à-dire, comment construire un **référentiel d'exigences** couvrant toute la réglementation ?

4. Vers un référentiel d'exigences

L'élaboration d'un référentiel d'exigences, à partir des textes réglementaires, n'est pas une formalité. Cela demande d'analyser, comprendre et maîtriser les exigences réglementaires, afin de pouvoir les classer et les hiérarchiser.

La première étape qui nous semble nécessaire est de définir clairement la notion d'exigence, comme nous l'avons entrepris pour la notion de pratique au chapitre précédent. C'est ce que nous proposons dans la partie suivante. Nous utilisons les définitions les plus communes, émises par des sources de référence.

4.1. Qu'est-ce qu'une exigence ?

L'ISO 9000 : 2000 [ISO, 2000] définit une exigence comme étant « un besoin¹ ou une attente formulée, ou, en général, implicite ». Cette définition a le mérite d'être concise ! Cependant, elle ne nous semble pas assez précise. Le dictionnaire nous fournit plus d'informations.

Ainsi, une exigence est [Le Petit Larousse, 2001] :

1. Ce qu'un personnage réclame, exige d'autrui.
2. Ce que l'être humain réclame comme nécessaire à la satisfaction de ses besoins, de ses désirs, de ses aspirations.
3. Ce qui est imposé par une discipline, une soumission.

Les notions d'émetteur et de cible apparaissent ici : une exigence est émise par un personnage (1) – un être humain (2) – une autorité (3) et porte sur les activités d'autrui (1).

Le SEI propose également les définitions suivantes [SEI, 2002a]. Une exigence est :

1. Une condition (état) ou une capacité réclamée par un utilisateur dans le but de résoudre un problème ou d'atteindre un objectif.
2. Une condition (état) ou une capacité que doit posséder un produit (ou un de ces composants) pour satisfaire un contrat, un standard, des spécifications ou tout autre document « formellement imposé ».
3. Une représentation documentée d'une condition ou d'une capacité.

Il s'agit d'une définition beaucoup plus pragmatique, avec une forte vision « métier de l'informatique ». Le second point est une application du premier pour un produit (ou un de ses composants). Il est intéressant de constater que pour le SEI, une exigence est obligatoirement

¹ Un besoin est « ce qui est nécessaire, ou perçu comme tel » [ISO, 2000].

formulée, puisqu'elle fait l'objet d'un contrat (ou d'un standard, de spécifications...), alors que pour l'ISO, une exigence peut être implicite.

Ces propositions apportent une information que nous jugeons très pertinente : une exigence est une condition ou une capacité ou une représentation documentée de celles-ci. Dans une vision « produit », une exigence peut donc décrire une fonctionnalité attendue. Dans une vision « processus », une exigence peut porter sur les capacités de ces derniers et donc sur les activités associées.

L'Association Française d'Ingénierie Système (AFIS) adopte, quant à elle, une vision assez proche de celle du SEI [AFIS, 2004] : « quelque chose qui prescrit ce qu'un produit doit faire, avec quelles performances et sous quelles conditions, pour atteindre un but donné ». L'AFIS précise qu'une exigence est un « énoncé qui prescrit une fonction, une aptitude, une caractéristique ou une limitation à laquelle doit satisfaire un produit ou un processus dans des conditions d'environnement données » et qu'elle résulte d'une « négociation client-fournisseur ».

Nous souhaitons affiner la première définition du SEI, afin d'y faire apparaître explicitement certaines notions critiques. La définition que nous proposons est donc la suivante (illustrée par la Figure III-4) :

Une exigence :

- émane d'un émetteur, qui peut être soit : un être humain, une organisation (client, fournisseur...) ou une autorité (choisie ou subie¹)
- vise une cible, qui peut être soit : autrui, un produit, une organisation et ses processus
- est une condition ou une capacité nécessaire pour la satisfaction : d'un besoin (ce qui est nécessaire), d'un désir (ce qui est voulu), d'une attente (ce qui est souhaité).

Notons enfin que la notion d'exigence englobe généralement ce que la bibliographie définit comme des « contrainte » ou des « pré-requis ».

De plus, par la suite, nous appellerons « référentiel d'exigences » un ensemble d'exigences cohérent et structuré.

¹ Cette différenciation nous permet de préciser le fait qu'être soumis à des exigences peut être un choix volontaire (dans le cas d'une certification par exemple) ou non (dans le cas réglementaire par exemple).

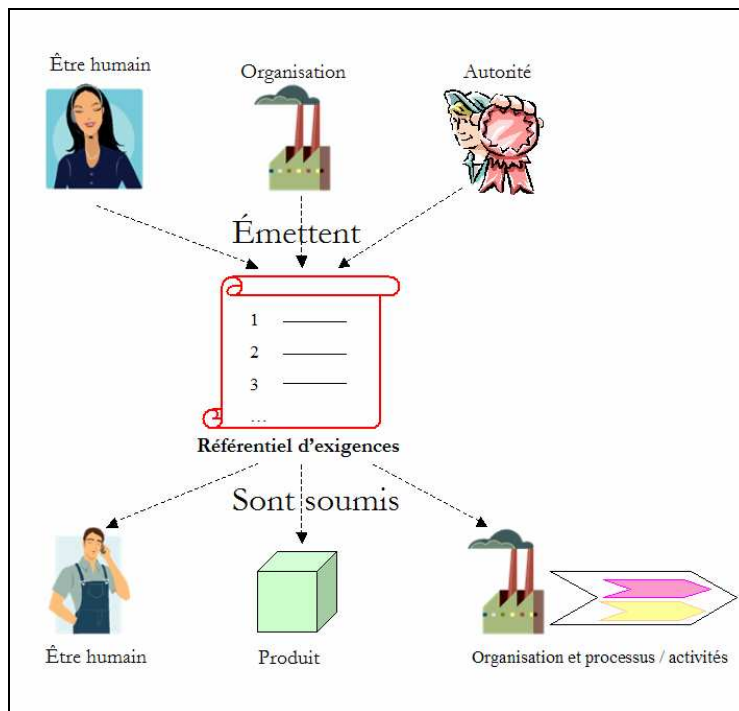


Figure III-4 : qui peut émettre des exigences ? Qui peut y être soumis ?

La figure suivante (Figure III-5) montre les éléments sur lesquels une exigence peut porter, avec une vision processus (formalisme SADT).

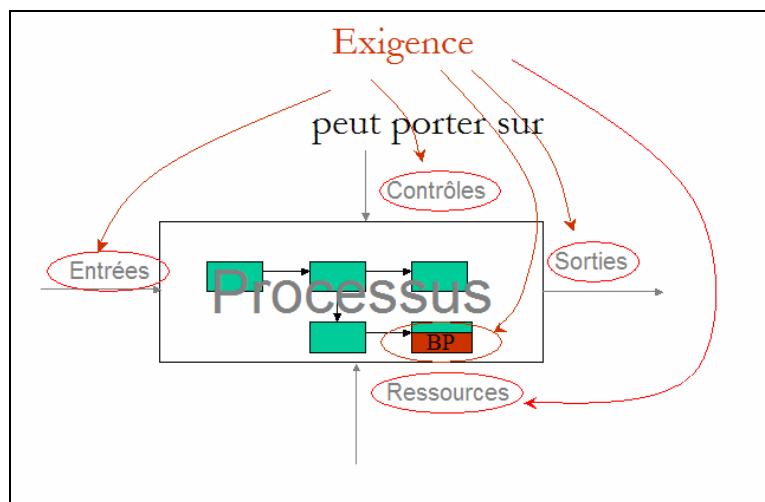


Figure III-5 : exigence et processus

Ainsi, comme le montre la Figure III-5, lorsqu'une exigence prend pour cible un processus, elle peut porter sur :

- les entrées du processus (niveau de qualité, caractéristiques physiques...),
- les sorties du processus (il peut alors s'agir d'une exigence portant sur le produit, puisqu'un produit est obtenu en sortie de processus),

- les mécanismes qui contrôlent l'exécution du processus (fréquences, types de contrôles...),
- les ressources utilisées par le processus (formation, qualification...),
- les activités qui composent le processus, et dans le cas qui nous concerne, les bonnes pratiques à respecter pour réaliser ces activités. Le fait de **devoir** respecter les bonnes pratiques relatives à un métier est désigné par l'AFSSAPS, par exemple, comme étant une « *exigence de bonnes pratiques* » ou une « *règle de bonnes pratiques* » [AFSSAPS, 2007].

Lorsque, par la suite, nous utiliserons le terme « exigence réglementaire », il faudra comprendre qu'il s'agit d'une exigence émise par une autorité, qui vise une organisation (et donc ses processus et ses activités) et ses produits, et qui est une condition nécessaire pour la satisfaction d'un besoin « réglementaire » (c'est-à-dire le respect de la réglementation). Dans le cadre réglementaire, la bonne pratique n'est plus une préconisation, c'est une **obligation**.

Nous avons donc un ensemble de textes réglementaires, qui proposent, chacun, un certain nombre d'exigences relatives aux systèmes informatisés. Comme nous allons le voir, il est nécessaire de faire un travail de tri et d'organisation, afin d'obtenir un référentiel d'exigences satisfaisant.

4.2. L'élaboration du référentiel d'exigences à partir des textes réglementaires

4.2.1. Pourquoi ?

Il faut bien comprendre que les textes que nous venons de présenter sont le résultat de plusieurs dizaines d'années d'analyse et de retour d'expérience. Les textes évoluent « en continu », en fonction des nouvelles méthodes, des technologies, des problématiques liées au métier... Il est donc logique que l'information qu'ils contiennent soit relativement « désorganisée ». L'ensemble des exigences obtenu en additionnant le contenu de tous les textes n'est pas exploitable tel quel :

- les exigences traitent toutes des systèmes informatisés, mais certaines sont adaptées à la vision de l'utilisateur (maîtrise d'ouvrage), c'est-à-dire qu'elles s'intéressent plus à l'utilisation des systèmes et à leur validation, alors que d'autres sont adaptées à la vision du fournisseur (maîtrise d'œuvre) : elles s'intéressent à la conception, l'exploitation et la maintenance des systèmes,
- certaines exigences sont redondantes entre elles, totalement ou partiellement (nous avons vu que les BPD sont issues d'une simplification des BPF par exemple). Il n'est pas étonnant que les textes relatifs aux systèmes informatisés demandent que les ressources humaines qui vont utiliser les systèmes soient formées, que l'on soit durant la phase d'essais cliniques ou durant la phase de production !
- Le vocabulaire utilisé n'est pas universel. Il dépend du métier concerné. Une « donnée critique » pour les BPL n'est pas nécessairement aussi critique pour les

BPF. Cette connaissance ne peut être fournie que par des experts des domaines concernés, rompus à ce champ sémantique,

- les exigences, telles que présentées dans les textes, ne sont pas « unitaires ». L'annexe 11 des BPF, par exemple, est découpée en quatre parties, elles-mêmes découpées en paragraphes. Chaque paragraphe contient plusieurs exigences, mais ces exigences ne sont pas identifiées « formellement ». Ainsi l'exemple 4.11 donné sur la Figure III-3 illustre ceci : il ne s'agit pas d'une seule exigence mais de quatre différentes : une concernant la procédure de validation des modifications, une autre concernant la procédure de contrôle des modifications, une troisième concernant la procédure d'autorisation des modifications en enfin une dernière concernant la procédure de mise en œuvre des modifications. Divisées de cette façon, les exigences sont plus simples à comprendre et à manipuler.

D'une manière très schématique, il nous faut donc passer de la situation de gauche sur la Figure III-6, à la situation de droite.

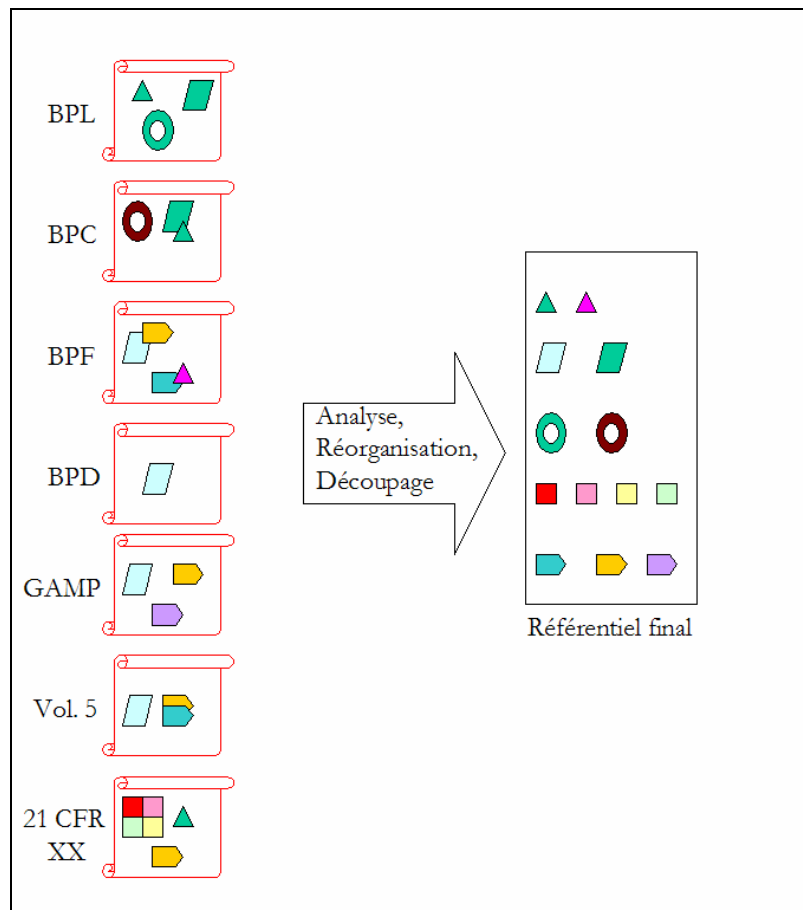


Figure III-6 : travail d'analyse, de réorganisation et de découpage des textes réglementaires

A partir des textes réglementaires (partie de gauche de la Figure III-6), par un travail d'analyse, de découpage et de réorganisation, il nous faut obtenir un référentiel exploitable (partie de droite). Chaque symbole du référentiel final représente alors une exigence unitaire. Elles sont regroupées par catégories (sur cette figure, une catégorie correspond à une forme).

4.2.2. Méthode de conception d'un référentiel unifié

Un travail d'organisation et de traitement est donc nécessaire afin d'obtenir un référentiel d'exigences réglementaires « exploitable ». Concrètement, les activités réalisées afin d'obtenir ce référentiel sont listées dans la Figure III-7.

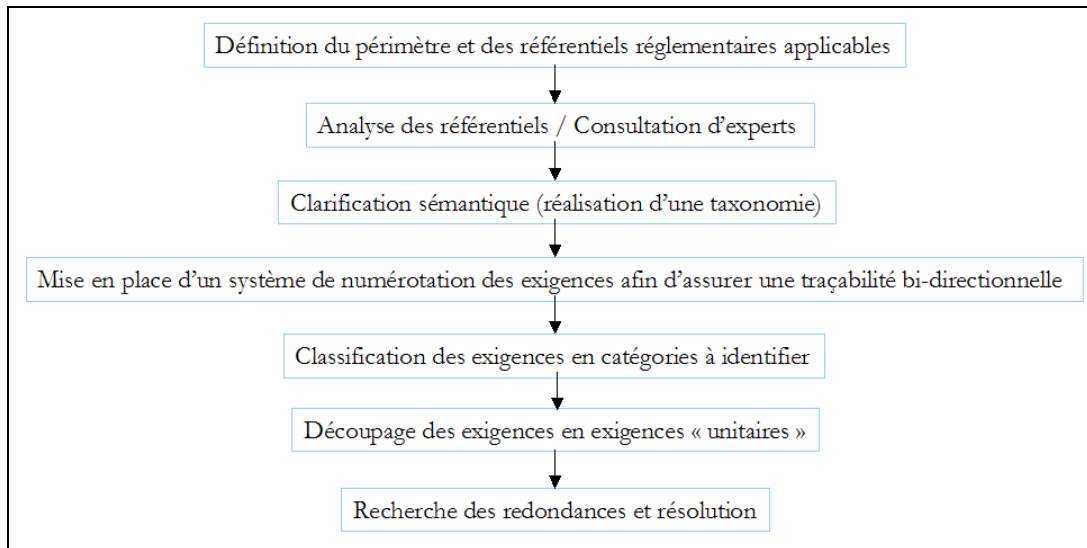


Figure III-7 : l'élaboration du référentiel d'exigences unifié

Ainsi, dans notre cas, les référentiels applicables sont ceux présentés dans la partie 3 de ce chapitre. L'analyse de ces textes a été réalisée grâce à la consultation des nombreux experts, tant dans le domaine pharmaceutique en général, que dans celui des systèmes informatisés. Ils nous ont également permis d'établir une taxonomie des termes utilisés par la réglementation. Nous avons mis en place un système de numérotation qui permet de savoir, à partir d'une exigence donnée, le texte dont elle provient et les « exigences-filles » qu'elle a permis de créer. Comme il est possible que des exigences soient redondantes, certaines exigences peuvent devenir obsolètes. Le système de numérotation permet de les identifier.

De plus, l'analyse rigoureuse des référentiels nous a permis d'identifier trois « familles » d'exigences.

La première correspond aux exigences qui s'appliquent directement à la MOA ou qui concernent des besoins de celle-ci :

- **fonctionnelles**, qui décrivent des fonctionnalités que doivent posséder des systèmes informatisés soumis à la réglementation,
- **pour MOA**, qui regroupe les exigences qui ne s'appliquent pas directement à la maîtrise d'œuvre.

Exemple : « lorsque les données électroniques de deux études et plus sont stockées sur un seul support (disque ou bande), on prévoira un index détaillé ».

La seconde concerne les activités relatives à l'obtention d'un certain niveau de qualité, qui se rapproche de la notion d'activité générique :

- **système de management de la qualité**, qui regroupe les exigences relatives aux pratiques d'assurance et de contrôle de la qualité (notamment, la gestion de la documentation) et à la gestion du SMQ,
- **cycle de vie**, qui regroupe les exigences qui décrivent les attendus pharmaceutiques en termes de description des cycles de vie des systèmes informatisés et de la gestion des déviations par rapport aux cycles de vie planifiés,
- **formation et gestion du personnel**, qui regroupe les exigences relatives à la gestion des ressources humaines.

La dernière famille concerne les activités d'ingénierie :

- **validation**, regroupe les exigences relatives à la validation pharmaceutique (voir annexe),
- **ingénierie informatique** : cette catégorie regroupe les exigences qui concernent l'ingénierie de conception et de réalisation et d'exploitation des systèmes informatisés,
- **gestion des changements** : cette catégorie rassemble les exigences qui traitent de la maîtrise des changements apportés aux systèmes informatisés. Il s'agit pour nous de la gestion des changements au sens « large », puisqu'elle vise toutes les étapes du cycle de vie : conception, réalisation, exploitation et retrait. Les changements peuvent être des modifications documentaires, des corrections ou évolutions apportées aux systèmes, que ce soit au niveau physique ou au niveau logique, des déclarations d'anomalies...
- **temps de rétention des informations** : la réglementation pharmaceutique impose des durées de conservation aux documents critiques,
- **sécurité de l'information** : cette catégorie regroupe aussi bien les exigences relatives à la sécurité « physique » : accès aux locaux, protection contre les intempéries, *etc.*, que la sécurité logicielle.

Le travail d'analyse des textes et de tri des exigences réglementaires en catégories, et du découpage en exigences unitaires, nous a permis de passer d'un référentiel initial de deux cent vingt et une exigences à un référentiel final contenant cent soixante quinze exigences (référentiel unifié). Celles-ci se répartissent dans les dix catégories que nous venons de citer de la manière suivante (

Figure III-8). En théorie, ce travail de classification et de tri des exigences réglementaires pharmaceutiques doit permettre la construction d'un modèle conceptuel générique. Dans la pratique, ces deux tâches ont été réalisées en parallèle.

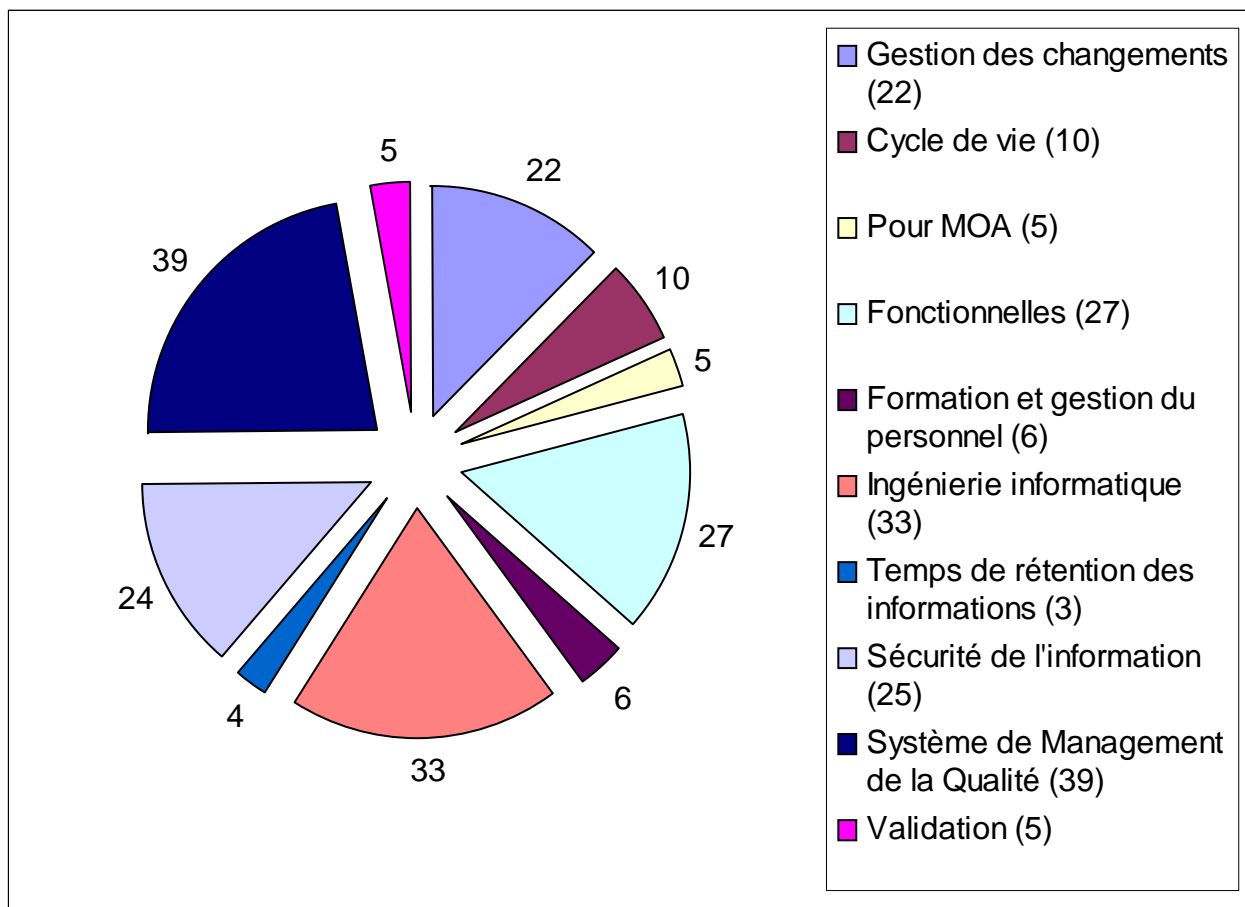


Figure III-8 : répartition des exigences par catégories

Le nombre très important d'exigences pour la catégorie « Système de Management de la qualité » s'explique car la plupart des exigences qui composent cette catégorie proviennent du GAMP. Cet ouvrage propose des exigences plus opérationnelles et détaillées que les textes réglementaires et donc, par conséquent, très nombreuses. Il s'agit notamment des exigences relatives à la gestion de la documentation et à la définition des responsabilités.

4.2.3. Le modèle conceptuel

Finalement, il est possible, à partir des informations présentées dans ce chapitre, d'établir un modèle conceptuel du référentiel d'exigences que nous avons obtenu, en utilisant le langage UML (diagrammes de classes). La Figure III-9 positionne tout d'abord ce modèle dans l'architecture à quatre niveaux de l'OMG.

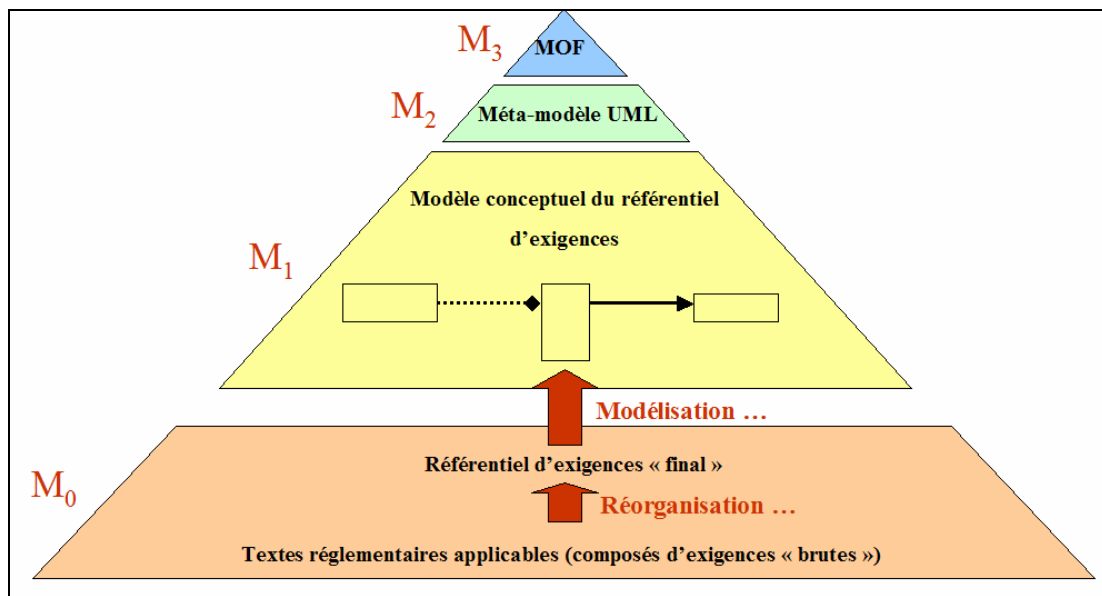


Figure III-9 : positionnement du modèle conceptuel de référentiel d'exigences réglementaires dans l'architecture de l'OMG

Ainsi, de notre point de vue, le niveau M_0 est composé d'une part des textes réglementaires applicables aux systèmes informatisés, et d'autre part, du référentiel d'exigences unifié obtenu à partir de ces textes. Le modèle conceptuel appartient au niveau M_1 . Nous l'exprimons sous forme de diagramme de classes UML (niveau M_2).

Nous proposons le diagramme de classes UML suivant (Figure III-10).

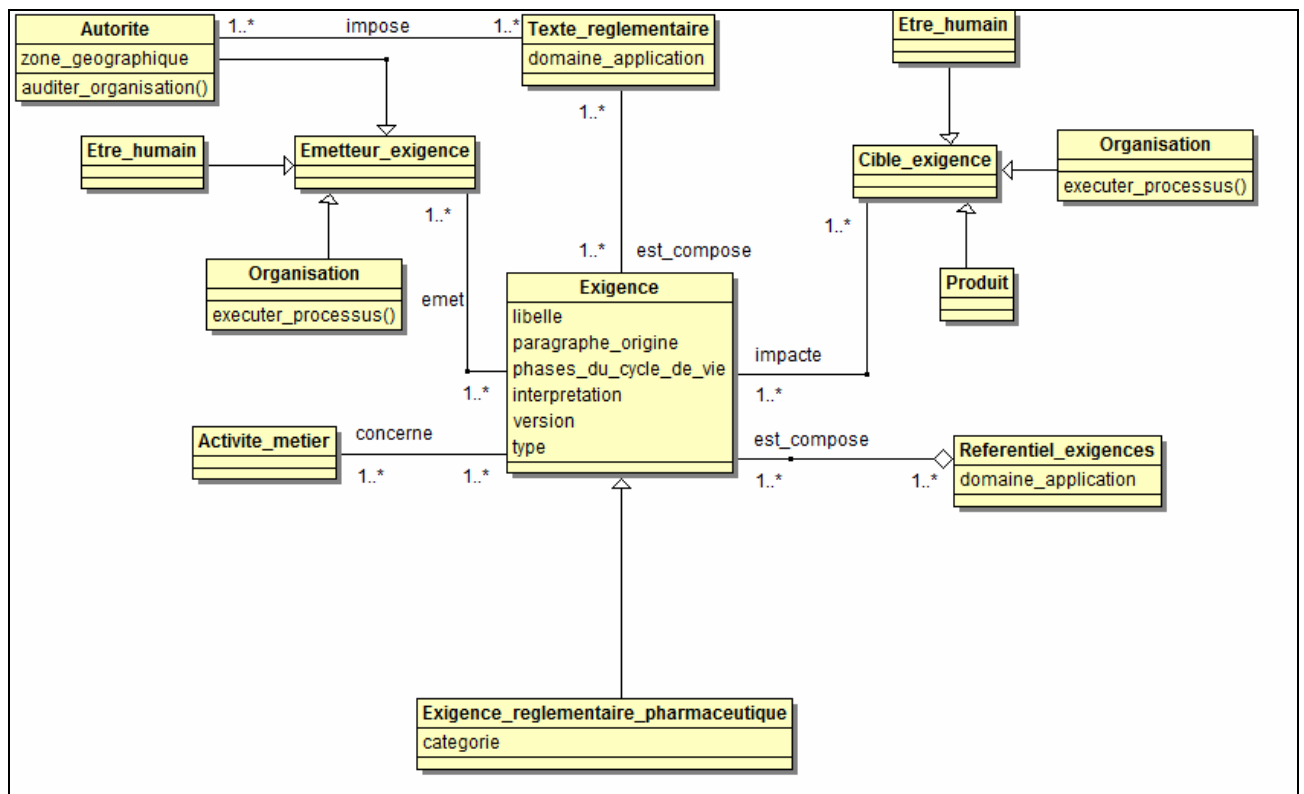


Figure III-10 : modèle conceptuel du référentiel d'exigences

Une exigence est donc caractérisée par six attributs :

- son libellé,
- son origine : il permet de savoir de quel texte provient chaque exigence,
- les phases du cycle de vie concernées (le cycle de vie des systèmes informatisés dans notre cas),
- une interprétation : c'est-à-dire une explication des conséquences de l'exigence sur le produit, les processus...
- sa version, qui est une liste contenant les informations relatives aux différents états successifs de l'exigence et qui précise donc si elle est obsolète ou non,
- son type : entrée, sortie, mécanisme de contrôle ou ressources (cf. Figure III-5),

Nous avons donc obtenu le modèle conceptuel que nous recherchions. Il va nous permettre de caractériser le mapping que nous avons à réaliser.

5. Synthèse

Ce troisième chapitre était dédié au monde réglementaire pharmaceutique, et plus particulièrement aux impacts de la réglementation sur les systèmes informatisés. Après avoir présenté les quelques autorités réglementaires qui nous intéressaient, nous avons défini la notion de systèmes informatisés. Nous avons ensuite présenté les textes réglementaires relatifs à notre problématique industrielle.

Ces parties nous ont permis d'obtenir un modèle conceptuel autour de la notion de référentiel d'exigences. Il s'agit de la seconde « brique » nécessaire à la réalisation du mapping (la première étant le modèle conceptuel des référentiels de bonnes pratiques obtenu dans le chapitre précédent).

Le chapitre suivant va donc nous permettre de présenter le mapping réalisé, ainsi que le problème de génie industriel rencontré et sa méthode de résolution.

Chapitre IV. LA MISE EN RELATION DES REFERENTIELS

Nous avons établi dans le premier chapitre que la réalisation d'un mapping entre deux « mondes » comparables exige :

- l'existence et la possession des modèles des deux « mondes » à mettre en relation : ces modèles sont obtenus par passage du niveau M_0 au niveau M_1 de l'architecture proposée par l'OMG,
- l'identification des éléments des modèles à mettre en relation (définir et établir la comparabilité entre certains éléments, ce qui équivaut à identifier les « règles de mise en relation »),
- la déduction, à partir des règles établies, des relations dans le « monde réel » (repasser au niveau M_0).

Concrètement, les second et troisième chapitres nous ont permis d'obtenir les modèles de nos deux « mondes » : celui des référentiels de bonnes pratiques informatiques et celui des exigences réglementaires pharmaceutiques. Il nous reste donc à établir la comparabilité entre les éléments de ces modèles et à « tirer » les liens au niveau M_0 . C'est ce que nous proposons de faire dans ce chapitre.

Ce chapitre est découpé en deux parties complémentaires. La première va nous permettre de faire un état de l'art synthétique sur les mappings entre référentiels de bonnes pratiques informatiques existants et de faire une synthèse sur les approches utilisées afin de situer nos travaux dans le paysage académique. Nous présenterons également les caractéristiques du mapping que nous souhaitons réaliser. La seconde partie présentera la formalisation du problème, ainsi qu'une comparaison des méthodes de résolution envisagées.

1. Présentation générale du mapping réalisé

1.1. Etat de l'art sur les mappings entre référentiels de bonnes pratiques informatiques existants

L'idée de réaliser un mapping entre deux référentiels n'est pas nouvelle. La bibliographie en propose de nombreux et nous souhaitons ici présenter les principaux travaux dans ce domaine, en apportant une attention particulière à ceux qui mettent en jeu le CMMI.

Plusieurs travaux visent à étudier les correspondances entre les bonnes pratiques du CMMI et le référentiel ISO 9001. Citons tout d'abord les travaux de [Mutafelija, 2001], [Mutafelija *et al.*, 2003a] et [Mutafelija *et al.*, 2003b], dont un des principaux résultats est une liste de correspondances entre les paragraphes de la norme ISO 9001 : 2000 et les pratiques génériques et spécifiques du CMMI. Il s'agit d'un mapping annoté (au sens où nous l'avons défini dans le chapitre 1), car à chaque correspondance est associée une « force » : S (pour « Strong ») si le lien est particulièrement pertinent, M (pour « Medium ») si la correspondance n'est pas totale et enfin W (pour « Weak »), si la correspondance est faible.

Ces travaux ont ensuite servi de base pour [Yoo *et al.*, 2006]. Ce document propose un modèle unifié de ces deux référentiels, et pour [Ferchichi, 2008] qui propose un cadre afin de générer un référentiel qualité conforme à ces deux référentiels. Moins récemment, les problématiques de mappings avaient déjà été abordées entre l'ISO 9001 : 1994 et le CMM : [Bamford *et al.*, 1993], [Paulk, 1993], [Paulk, 1994], [Paulk, 1995] et [Jalote, 1999].

Le SEI propose un mapping entre les modèles CMM et CMMI [SEI, 2002d]. L'objectif de ce travail est d'aider les organisations qui ont atteint un certain niveau de maturité du CMM à définir leur niveau équivalent dans le modèle CMMI.

D'autres études proposent des mappings entre les pratiques du modèle de maturité de l'ISO SPICE (Software Process Improvement and Capability dEtermination) ([ISO, 1995a], [ISO, 1995b] et [ISO, 1995c]) et celles du CMMI [SQI, 2001a], [SQI 2001b]. Ces référentiels sont très clairement comparables, puisqu'ils visent au même objectif (amélioration du processus logiciel), et de la même manière (préconisation de bonnes pratiques).

Certains travaux proposent également des mappings entre le CMMI et des méthodologies de développement. C'est le cas de [Marçal *et al.*, 2007] qui étudie les équivalences entre une approche de développement AGILE (le SCRUM) et les pratiques des secteurs clés de la catégorie « Gestion de projet » de l'architecture continue de CMMI. Le SEI propose d'autres mappings entre le CMMI et des méthodologies de développement¹.

Nous proposons, dans le Tableau IV-1, une synthèse des caractéristiques de ces travaux.

¹ Voir : <http://www.sei.cmu.edu/cmmi/adoption/comparisons.html>

	Éléments comparés	Direction	Cardinalité	Complexité des relations
Mapping CMMI / ISO 9001 : 2000 [Ferchichi, 2008]	Pratiques du CMMI / paragraphes et sous paragraphes de l'ISO 9001 : 2000	Bi-directionnel (deux mappings)	Relations « n : m »	Mapping annoté sur trois niveaux + commentaires sur les écarts
Mapping ISO SPICE / CMMI [SQI, 2001a] et [SQI, 2001b]	Pratiques du CMMI / pratiques de l'ISO SPICE	Uni-directionnel : de CMMI vers ISO SPICE	Relations « 1 : n »	Quelques commentaires sur les écarts
Mapping CMM / CMMI [SEI, 2002d]	Pratiques et sous-pratiques du CMMI / pratiques du CMM	Bi-directionnel (deux mappings)	Relations « n : m »	Commentaires sur les écarts
Mapping CMMI / SCRUM [Marçal et al., 2007]	Pratiques spécifiques des secteurs clés de la catégorie gestion de projet / pratiques de SCRUM	Uni-directionnel : de CMMI vers SCRUM	Relations « 1 : n »	Annotation sur 3 niveaux (« Satisfied », « Partially Satisfied » et « Unsatisfied ») + explications très détaillées des écarts

Tableau IV-1 : synthèse des travaux existants sur les mappings CMMI

Dans un premier temps, nous constatons qu'aucune des démarches précédentes n'est basée sur la définition et l'utilisation de méta-modèles, afin d'établir la comparabilité des éléments mis en relation. Notre démarche, qui se justifie par la définition même de la notion de mapping, est donc originale et cohérente.

Dans un second temps, notons que ces mappings répondent tous au même type de problématique : identifier les « équivalences » entre deux référentiels. Certains sont bi-directionnels, d'autres ne couvrent qu'une seule direction. Dans tous les cas, au minimum, un élément d'un référentiel est mis en relation avec plusieurs éléments de l'autre référentiel (relations « 1 : n »). Le fait de réaliser un mapping bi-directionnel implique que les relations deviennent du type « n : m ».

Notre problématique est légèrement différente, puisque nous ne recherchons pas les équivalences entre deux référentiels, mais entre deux « ensembles de référentiels ». Un ensemble de référentiels étant une « union » de plusieurs référentiels, des problématiques de « couverture » peuvent apparaître. En d'autres termes, que faire si une pratique du CMMI et une pratique de l'ISO 20000 permettent d'arriver au même résultat ? De plus, *a fortiori* de par la structure même du modèle CMMI, certaines pratiques sont redondantes (nous donnerons des exemples par la suite).

Cela introduit le besoin de manipuler des relations « évoluées », c'est-à-dire la possibilité de **pouvoir choisir** des combinaisons d'éléments dans les référentiels. Nous détaillons ceci dans la partie suivante.

1.2. Notre mapping

Repositionnons visuellement notre démarche dans la pyramide de l'OMG.

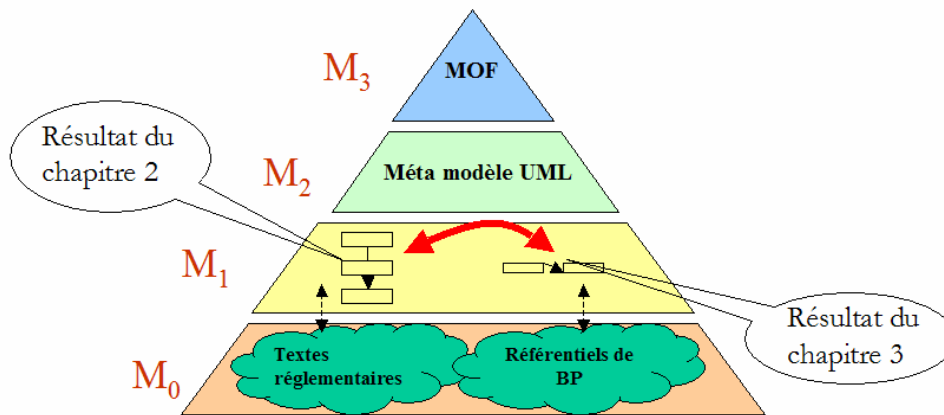


Figure IV-1 : positionnement des modèles dans l'architecture de l'OMG

Le niveau M_0 (le monde réel) est composé de l'ensemble des référentiels que nous souhaitons mettre en relation. Nous différencions les référentiels de bonnes pratiques informatiques et les textes réglementaires pharmaceutiques. Dans les deux cas, il s'agit de descriptions de règles qui s'appliquent aux activités « métier » (à titre « informatif » lorsqu'il s'agit des référentiels de bonnes pratiques et à titre « obligatoire » lorsqu'il s'agit des textes réglementaires). Nous avons également fait apparaître sur la Figure IV-1 le deuxième niveau du monde réel : les activités métier. En effet, sur les Figure II-20 et Figure III-10 les activités métier étaient liées aux notions de bonne pratique et d'exigence par une relation d'association. En les rajoutant à la pyramide de l'OMG, nous matérialisons le fait que les exigences et les bonnes pratiques s'appliquent aux mêmes entités. C'est ce caractère qui va nous permettre de réaliser la mise en relation.

Le niveau M_1 est alors composé des modèles conceptuels en langage UML que nous avons présenté dans les chapitres précédents. Le langage utilisé pour les représenter étant UML, il s'agit du niveau M_2 . Le niveau M_3 est alors, comme d'habitude, le MOF.

Les carrés (« filtre industriel ») sur la Figure IV-1 mettent en valeur le fait que ce ne sont pas tous les textes réglementaires qui sont sélectionnés pour réaliser le mapping. En effet, il ne faut inclure que ceux qui sont en rapport avec l'activité de la DSI et ceux qui correspondent à une réalité (si l'entreprise ne souhaite pas vendre de médicaments en Ouzbekistan, rien n'impose le respect des exigences réglementaires relatives à cette zone géographique !). Il s'agit donc d'un « filtre », défini par le périmètre du cas applicatif industriel. Rappelons que la définition mathématique de mapping fournie dans le premier chapitre autorise l'utilisation de « sous-parties » des modèles (le « $Sub(A)$ » de la définition). C'est exactement le but du « filtre industriel » : nous ne souhaitons pas utiliser l'ensemble des textes réglementaires, mais seulement ceux qui nous intéressent. Il s'agit donc de préciser les « domaines de définition » des ensembles à mettre en relation.

Les modèles conceptuels nous permettent d'identifier les éléments qui seront mis en relation entre référentiels de bonnes pratiques et textes réglementaires (double flèche sur la Figure IV-1).

Une étude fine des référentiels de bonnes pratiques et des textes réglementaires nous a permis d'établir que l'information que propose chaque exigence réglementaire est comparable à celle fournie, soit par :

- une **bonne pratique générique** (d'après la sémantique introduite dans le chapitre 2),
- une **bonne pratique spécifique**,
- une **combinaison de bonnes pratiques**,
- un **objectif spécifique attendu**, ce qui finalement est équivalent à une combinaison de bonnes pratiques spécifiques prédéfinie.

De plus, le chapitre 3 nous a permis d'établir qu'une exigence réglementaire peut porter soit sur (Figure III-5) :

- les entrées du processus ou d'une activité de celui-ci (niveau de qualité, caractéristiques physiques...),
- les sorties du processus (ou d'une activité) : il peut alors s'agir d'une exigence portant sur le produit, puisqu'un produit est obtenu en sortie de processus,
- les mécanismes qui contrôlent l'exécution du processus ou d'une activité qui le compose (fréquences, types de contrôles...),
- les ressources utilisées par le processus ou l'activité (formation, qualification...),
- les activités qui composent le processus.

Finalement, il est possible de construire le tableau de correspondances suivant (Tableau IV-2), qui se base sur les définitions proposées dans les chapitres précédents. Il nous permet d'établir la comparabilité au niveau « contextuel ».

Modèle générique des exigences réglementaires (chapitre 3)		Conditions	Modèle générique des référentiels de bonnes pratiques (chapitre 2)
Exigence		Une exigence <i>est respectée par</i> une combinaison de bonnes pratiques si :	Bonne pratique (BP)
Premier niveau de mise en relation			
Libellé	▶ la combinaison des libellés des BP permet de respecter l'exigence		Libellé
Phases du cycle de vie applicatif	▶ toutes les phases du cycle de vie applicatif concernées par l'exigence sont couvertes par la combinaison des phases des cycles de vie des BP		Phases du cycle de vie applicatif
Second niveau de mise en relation			
Type	Ressources	▶ la combinaison de BP porte sur les activités qui permettent d'obtenir les ressources	Activité
	Contrôle	▶ la combinaison de BP porte sur les activités qui permettent d'obtenir le mécanisme de contrôle	
	Entrée/sortie	▶ la combinaison de BP porte sur les activités qui permettent d'obtenir l'entrée / la sortie	
	BP	▶ la combinaison de BP porte directement sur les activités concernées par l'exigence	

Tableau IV-2 : correspondances entre exigences et bonnes pratiques

Le Tableau IV-2 montre que nous différencions deux niveaux pour la mise en relation. Dans un premier temps, s'il existe pour une exigence donnée une bonne pratique ou une combinaison de bonnes pratiques qui permettent de la couvrir complètement, et que ceci peut être établi directement à partir des libellés respectifs, alors la relation est établie. Dans un second temps, si la relation n'a pas pu être établie directement à partir des libellés, il faut alors passer par les activités métier concernées, afin d'identifier, par exemple, les processus, les procédures, les outils, *etc.*, concernées par une exigence et d'en déduire les bonnes pratiques liées.

Ainsi, lorsqu'une exigence porte sur l'entrée d'une activité, la bonne pratique à mettre en relation doit être relative à l'activité qui fournit cette entrée. Lorsqu'une exigence est une exigence de « bonne pratique », la bonne pratique à mettre en relation doit être cette bonne pratique. Il faut, bien entendu, que les phases du cycle de vie correspondent.

Lorsqu'une mise en relation adéquate est réalisée, la sémantique portée par le lien est « *une exigence est respectée par une combinaison de bonnes pratiques* ». Nous avons introduit le fait que les liens à manipuler doivent être « évolués ». La présence d'éventuelles redondances entre bonnes pratiques nous amène à devoir utiliser les opérateurs booléens classiques pour réaliser des combinaisons. Le formalisme utilisé est présenté dans le Tableau IV-3.

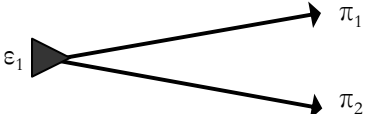
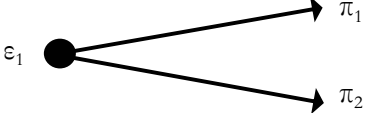

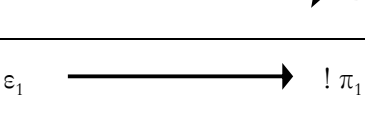
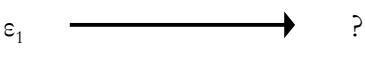
Opérateur	Représentation logique	Représentation graphique	Interprétation sémantique
ET	$\varepsilon_1 = \pi_1 \text{ ET } \pi_2$		L'exigence ε_1 est respectée par le déploiement des deux bonnes pratiques π_1 et π_2
OU	$\varepsilon_1 = \pi_1 \text{ OU } \pi_2$		L'exigence ε_1 est respectée par le déploiement d'une des deux bonnes pratiques π_1 ou π_2 (déploiement simultané possible)
XOU	$\varepsilon_1 = \pi_1 \text{ XOU } \pi_2$		L'exigence ε_1 est respectée par le déploiement d'une des deux bonnes pratiques π_1 ou π_2 (déploiement simultané impossible)
NON	$\varepsilon_1 = ! \pi_1$		L'exigence ε_1 est respectée si la bonne pratique π_1 n'est pas déployée
?	$\varepsilon_1 = ?$		Il n'y a pas de bonne pratique qui permet le respect de ε_1

Tableau IV-3 : les opérateurs et leur représentation

Lorsque toutes les relations sont représentées sous cette forme, nous obtenons un graphe global, composé d'autant de sous-graphes qu'il y a d'éléments ε_i .

Il est possible qu'un élément ε_i ne soit couvert que partiellement par une combinaison de π_j . Dans ce cas, nous découpons ε_i en deux nouveaux éléments, le premier, qui est couvert et le second qui

ne l'est pas. Cela nous dédouane des activités particulières liées à la gestion des couvertures partielles.

Si à un élément ε correspondent plusieurs combinaisons logiques d'éléments π , alors il est possible d'identifier plusieurs ensembles d'éléments π permettant de couvrir tous les éléments ε : c'est l'intérêt d'utiliser les opérateurs **OU** et **XOU**. Il est légitime de se poser la question « comment identifier la meilleure solution, c'est-à-dire l'ensemble **optimisé** (selon un certain critère) des éléments π ? ». La partie 2 de ce chapitre présente deux méthodes que nous avons identifiées pour répondre à cette question.

De plus, notre mapping se rapproche d'un mapping « exogène », puisque les deux modèles que nous souhaitons mettre en relation sont exprimés en UML, grâce à notre travail préliminaire.

Dans notre cas, manipuler un mapping bi-directionnel serait intéressant. Cela permettrait de répondre simultanément aux deux questions :

- quelles sont les bonnes pratiques informatiques à déployer afin de respecter l'ensemble ou un sous-ensemble des exigences réglementaires ?
- quelles sont les exigences réglementaires qui sont respectées par un ensemble de bonnes pratiques pré-définies ? Par exemple, si une organisation pharmaceutique a atteint le niveau 2 du CMMI, quelles sont les exigences pharmaceutiques qu'elle respecte ?

Le formalisme proposé permet la bi-directionnalité. Par contre, les méthodes d'exploitation que nous allons présenter dans la partie suivante ne permettent de « lire » le mapping que dans le sens « exigences \rightarrow bonnes pratiques ». La problématique industrielle à l'origine des travaux nous impose de travailler, au minimum, dans ce sens. L'obtention de la bi-directionnalité ne fait pas l'objet de cette étude et il s'agit de l'une des perspectives de nos recherches.

2. Mise en relation

Avant de présenter les méthodes d'exploitation des relations entre les deux référentiels, il nous faut préciser la notation utilisée, les variables et leurs domaines de définition.

2.1. Notations, variables et indices

Afin de rendre cette partie plus accessible, la signification de chaque variable sera illustrée par un exemple tiré de notre cas industriel.

- ▶ Ref_1 : un des ensembles à utiliser dans le mapping
 - | Dans notre cas, Ref_1 désigne le référentiel d'exigences réglementaires.

- ▶ Ref_2 : le second ensemble à utiliser dans le mapping
 - | Dans notre cas, Ref_2 désigne l'ensemble des bonnes pratiques qui appartiennent au modèle CMMI, à l'ISO 9001 : 2000 et à l'ISO 20000.

- ▶ $n = \text{Card}(Ref_1)$
 - | Il s'agit du nombre total d'exigences qui composent le référentiel d'exigences.

- ▶ $m = \text{Card}(Ref_2)$
 - | Il s'agit du nombre total de bonnes pratiques.

- ▶ $Ref_1 = \{ \varepsilon_i / i \in \mathbf{C}]0 ; n] \}$
 - | ε_i désigne le libellé de l'exigence réglementaire i .
 - | Ref_1 désigne donc l'ensemble des libellés des exigences.

- ▶ $E_i = \{ 0 ; 1 \}$
 - | Nous posons $E_i = 0$ si l'exigence E_i de libellé ε_i n'est pas couverte et $E_i = 1$ si elle l'est.

- ▶ $Ref_2 = \{ \pi_j / j \in \mathbf{C}]0 ; m] ; !\pi_j / j \in \mathbf{C}]0 ; m] \}$
 - | π_j désigne le libellé de la bonne pratique j .
 - | Ref_2 désigne donc l'ensemble des libellés des bonnes pratiques et de leurs complémentaires (sa « négation »).

- ▶ $P_j = \{ 0 ; 1 \}$
 - | Nous posons $P_j = 0$ si la pratique P_j de libellé π_j n'est pas à déployer (c'est-à-dire qu'il faut déployer son complémentaire) et $P_j = 1$ s'il faut la déployer.

De plus, nous souhaitons utiliser une propriété particulière des expressions logiques : **la propriété des mintermes** [Faure, 1979]. En assimilant le « OU logique » à un « + » et le « ET logique » à un « * », la propriété des mintermes affirme que toute expression logique peut s'exprimer comme somme de produits des variables et de leurs complémentaires. Il s'agit de la **Forme Canonique Disjonctive** (FCD) des expressions logiques. Nous illustrons ceci par les figures suivantes.

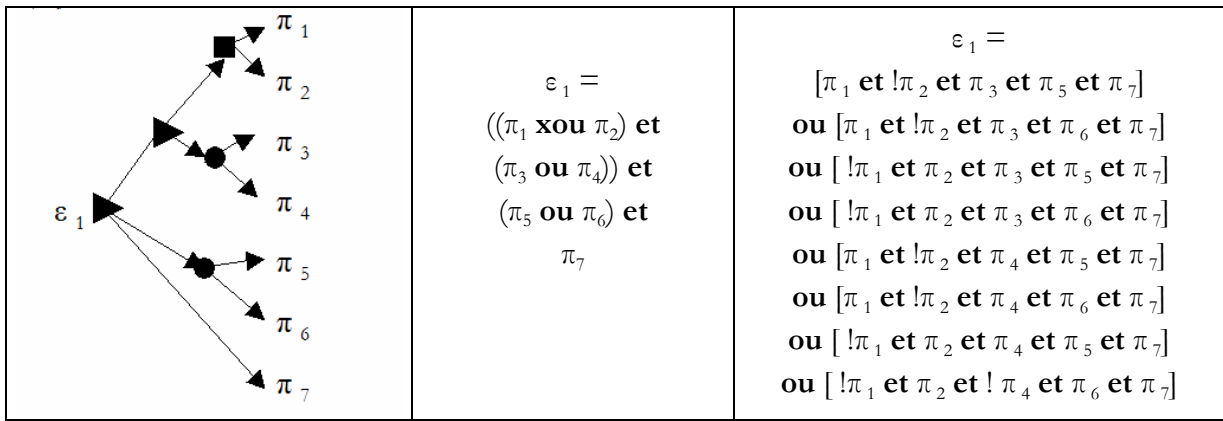


Figure IV-2 : exemple de combinaisons d'éléments de Ref₂ (P_j) équivalentes à un élément de Ref₁ (E_i)

Figure IV-3 : représentation logique de l'exemple de la Figure IV-2

Figure IV-4 : FCD de l'expression de la Figure IV-2 – forme littérale

$$E_1 =$$

$$[P_1 * !P_2 * P_3 * P_5 * P_7]$$

$$+ [P_1 * !P_2 * P_3 * P_6 * P_7]$$

$$+ [!P_1 * P_2 * P_3 * P_5 * P_7]$$

$$+ [!P_1 * P_2 * P_3 * P_6 * P_7]$$

$$+ [P_1 * !P_2 * P_4 * P_5 * P_7]$$

$$+ [P_1 * !P_2 * P_4 * P_6 * P_7]$$

$$+ [!P_1 * P_2 * P_4 * P_5 * P_7]$$

$$+ [!P_1 * P_2 * P_4 * P_6 * P_7]$$

Figure IV-5 : FCD de l'expression de la Figure IV-2 – forme algébrique

La Figure IV-2 montre un exemple de relation entre un élément de Ref₁ (E_i) et un ensemble d'éléments de Ref₂. La Figure IV-3 est l'équivalent de cette relation sous forme logique. Enfin, la Figure IV-4 correspond à la forme canonique disjonctive de cette relation. La Figure IV-5 est l'équivalent sous forme algébrique de la Figure IV-4. Les éléments mis en jeu ne sont plus les libellés des exigences (ε_i), mais leur valeur booléenne (E_j). Il en va de même pour les bonnes pratiques.

L'utilisation de la propriété des mintermes nous permet d'affirmer qu'il est toujours possible d'exprimer un élément de Ref₁ comme étant équivalent à une somme de produits d'éléments de Ref₂ et de leurs complémentaires. Sous forme graphique, cela veut dire que chaque élément E_i est relié à un nœud **OU**. Ce nœud est ensuite relié à un ensemble de nœuds **ET**. Le nœud **OU** sera noté X_i et les nœuds **ET** : Y_{i,k}. En théorie des graphes, il s'agit donc d'une arborescence¹, à trois niveaux de profondeur. Le premier niveau de profondeur correspond au premier sommet de l'arborescence, qui, d'après la propriété des mintermes, porte la sémantique « OU ». Cette même propriété nous permet d'affirmer que le second niveau de profondeur sera le sommet, portant la sémantique de l'opérateur logique « ET ». Enfin, le troisième niveau (les feuilles) sont des éléments de Ref₂. Nous illustrons ceci par la Figure IV-6. L'élément du premier niveau (nœud

¹ Une arborescence est un graphe orienté, où chaque sommet possède un précédent, sauf un qui n'en a pas (*la racine*) [Lopez, 2005].

OU) sera caractérisé par l'indice i : il s'agit du même indice que pour les éléments de Ref_i , car l'utilisation des formes canoniques disjonctives nous permet d'affirmer que chaque E_i sera en lien avec un et un seul nœud **OU** (X_i). Les éléments du second niveau seront caractérisés par un couple d'indices $(i ; k) : Y_{i,k}$. Enfin, les éléments du troisième niveau seront caractérisés par le triplet $(i , k , l) : P_{i,k,l}$.

Nous utilisons également la notion de degré sortant d'un nœud. Cette notion est couramment utilisée en théorie des graphes [Charon *et al.*, 1996].

Si x est un sommet d'un graphe G orienté, on appelle **degré sortant de x** , noté $d^+(x)$, le nombre d'arcs d'origine x .

Ainsi, appliqué à notre cas, le degré sortant d'un sommet correspond au « nombre de branches » qui « partent » d'un opérateur.

Les variables sont donc les suivantes :

- ▶ $A = \{ 0 ; 1 \}$, avec $A \in \{ X_i ; Y_{i,k} ; P_{i,k,l} \}$ et $i \in]0 ; n]$, $k \in [1 ; d^+(X_i)]$ et $l \in [1 ; d^+(Y_{i,k})]$
 - └ Le parcours de graphe va consister à définir par quels sommets la solution finale doit passer. Nous proposons la convention suivante : si la solution passe par un sommet, alors sa valeur est à « 1 », sinon elle est à « 0 ».

La Figure IV-6 positionne ces variables sur un exemple d'arborescence. L'exemple que nous proposons sur cette figure est un exemple fictif, directement sous forme canonique disjonctive.

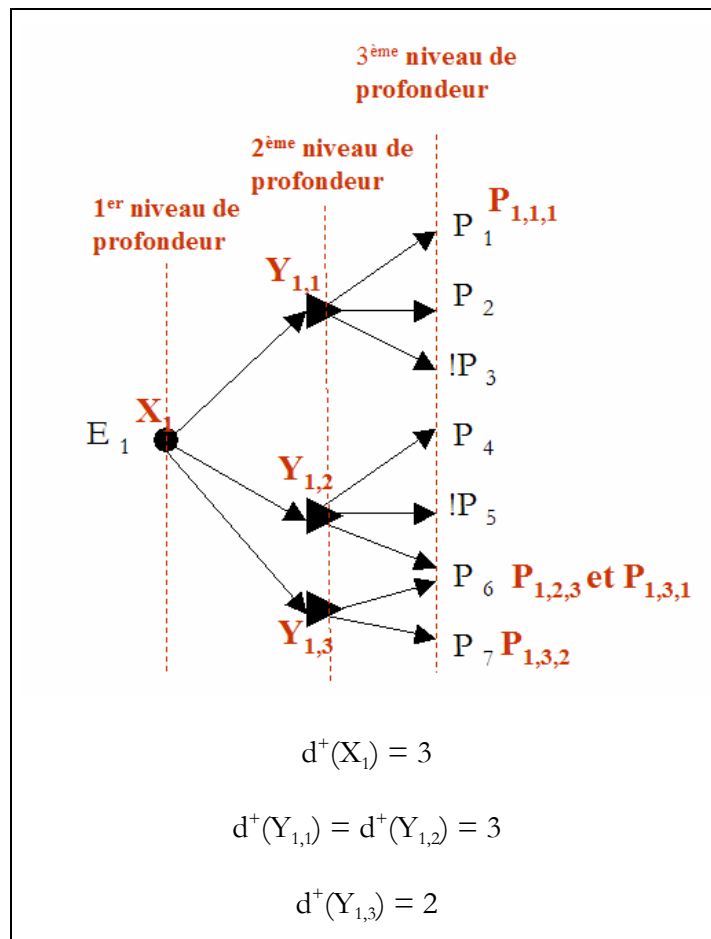


Figure IV-6 : illustration de la notation utilisée sur une FCD – notation algébrique (exemple fictif)

Comme le montre la Figure IV-6, un même élément de Ref_2 (P_i) est donc identifié par plusieurs notations différentes. Par exemple, P_6 correspond à la notation $P_{1,2,3}$, mais également à la notation $P_{1,3,1}$. Il convient donc de maintenir, tout au long de la résolution du problème, une « table des équivalences » des notations.

2.2. Deux méthodes de résolution

Notre objectif premier va être de minimiser le nombre total d'éléments des Ref_2 à déployer afin de couvrir tous les éléments de Ref_1 . Nous reviendrons sur le choix de cet objectif par la suite. Nous avons identifié deux méthodes concurrentes qui nous permettent de l'atteindre. La première est une méthode d'optimisation [Deguil *et al.*, 2007a] [Deguil *et al.*, 2007c], qui va mettre en jeu une fonction objectif et des contraintes linéaires. La seconde est un algorithme de recherche, qui va assimiler le système à une simple équation à résoudre [Deguil *et al.*, 2007b].

2.2.1. Méthode A : méthode d'optimisation

► Fonction objectif

L'objectif est de minimiser le nombre total d'éléments de Ref_2 à choisir afin de couvrir la totalité des éléments de Ref_1 . Or, nous avons vu qu'un même P_i pouvait être défini par plusieurs $P_{i,k,l}$ différents. La fonction qui établit les équivalences entre P_i et $P_{i,k,l}$ sera appelée « Équivalence ». Dans un premier temps, il faut appliquer cette fonction et ensuite rechercher l'objectif :

$$\text{Objectif} = \min\left(\sum_{j=0}^m P_j\right)$$

► Contraintes

Les contraintes qui définissent notre système sont relatives aux règles qui s'appliquent aux nœuds **OU** et **ET**, dans le parcours des graphes.

La première définit le comportement que doit suivre le parcours des nœuds **ET**. Le nœud aura pour valeur « 1 » si les $d^+(Y_{i,k})$ éléments liés par un arc sortant ont la valeur « 1 ». Cela se traduit par l'inéquation suivante :

$$\forall i \in [0; n], \forall k \in [0; d^+(X_i)], \sum_{l=1}^{d^+(Y_{i,k})} P_{i,k,l} \geq d^+(Y_{i,k}) * Y_{i,k}$$

Ainsi, quel que soit le nœud **ET** traité, si tous les $P_{i,k,l}$ en lien avec ce nœud sont égaux à « 1 », alors la somme de leurs valeurs est forcément égale au degré sortant du sommet. La valeur du booléen $Y_{i,k}$ passe alors à « 1 ».

De la même manière, si tous les $P_{i,k,l}$ ne sont pas égaux à « 1 », alors la somme de leurs valeurs est forcément inférieure au degré sortant du sommet (nœud **ET**). Sa valeur passe à « 0 ».

La seconde contrainte est relative aux nœuds **OU**. Un nœud **OU** (X_i) aura pour valeur 1, si parmi les $d^+(X_i)$ sommets **ET** qui y sont liés, il y en a au moins un à la valeur « 1 ». Cela se traduit par l'inéquation :

$$\forall i \in [0; n], \sum_{k=1}^{d^+(X_i)} Y_{i,k} \geq 1$$

La somme des valeurs des nœuds **ET** associés à un même nœud **OU**, quelque soit ce dernier, doit toujours être supérieure à « 1 ».

Notre système final est donc décrit par l'ensemble d'équation et d'inéquations suivant :

$$(1) \text{Objectif} = \min\left(\sum_{j=0}^m P_j\right)_{X_i, Y_{i,k}, P_{i,k,l}}$$

$$(2) \forall i \in [0; n], \forall k \in [0; d^+(X_i)], \sum_{l=1}^{d^+(Y_{i,k})} P_{i,k,l} \geq d^+(Y_{i,k}) * Y_{i,k}$$

$$(3) \forall i \in [0; n], \sum_{k=1}^{d^+(X_i)} Y_{i,k} \geq 1$$

Il s'agit d'un problème d'optimisation à fonction objectif et contraintes linéaires. La résolution d'un tel système peut se faire grâce à l'utilisation d'un logiciel d'optimisation. Il suffit d'une part de coder l'ensemble des relations entre Ref_1 et Ref_2 et d'autre part de coder la fonction objectif et les contraintes précédentes.

2.2.2. Méthode B : algorithme de recherche

La seconde méthode que nous présentons est plus « intuitive ». Elle est basée sur l'assimilation de notre système à une équation logique et sur la résolution de celle-ci.

Nous avons vu qu'il est possible d'exprimer chaque élément de Ref_1 comme étant une somme de produits, dans lesquels les variables sont des éléments (ou leurs complémentaires) de Ref_2 .

Il est donc possible de déterminer un système (S) d'équations logiques du type :

$$(S) = \begin{aligned} E_1 &= \sum \prod P_j \\ E_2 &= \sum \prod P_j \\ &\dots \end{aligned}$$

L'objectif étant la couverture totale des éléments de Ref_1 , nous souhaitons résoudre l'équation suivante :

$$\prod_{i=1}^n E_i = 1$$

Il suffit maintenant de substituer, dans l'équation précédente, chaque E_i par son expression logique et d'exprimer ce résultat sous forme canonique disjonctive. Chaque « somme » dans l'expression finale sous forme canonique disjonctive correspondra à une solution possible de l'équation. Il suffit alors de choisir la somme la plus adaptée, en fonction du critère à optimiser. Par exemple, si l'on souhaite minimiser le nombre total de pratiques à déployer, la somme qu'il faudra choisir sera celle de moindre dimension. Présenté de cette manière, notre problème s'apparente à une problématique de « choix d'investissements », tel que présenté dans [Faure, 1979]. Il est possible de rechercher une solution correspondant à un critère différent (par exemple, affecter un « poids » à chaque élément de Ref_2 et minimiser ce poids).

Nous illustrons ce mécanisme par un exemple simple. La situation que nous proposons dans la Figure IV-7 est déjà sous forme canonique disjonctive.

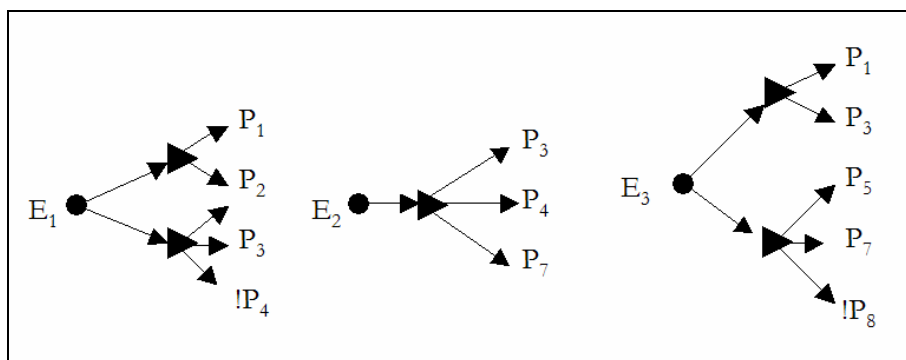


Figure IV-7 : exemple de FCD pour la seconde méthode

Ceci correspond au système d'équations :

$$\begin{aligned} E_1 &= P_1 * P_2 + P_2 * P_3 * !P_4 \\ E_2 &= P_3 * P_4 * P_7 \\ E_3 &= P_1 * P_3 + P_5 * P_7 * !P_8 \end{aligned}$$

L'équation générale à résoudre est donc :

$$E_1 * E_2 * E_3 = 1$$

La substitution donne les résultats suivants :

$$\begin{aligned} & [P_1 * P_2 + P_2 * P_3 * !P_4] * [P_3 * P_4 * P_7] * [P_1 * P_3 + P_5 * P_7 * !P_8] = 1 \\ \Leftrightarrow & P_1 * P_2 * P_3 * P_4 * P_7 * P_1 * P_3 + P_2 * P_3 * !P_4 * P_3 * P_4 * P_7 * P_1 * P_3 + \\ & P_1 * P_2 * P_3 * P_4 * P_7 * P_5 * P_7 * !P_8 + P_2 * P_3 * !P_4 * P_3 * P_4 * P_7 * P_5 * P_7 * !P_8 = 1 \end{aligned}$$

$$\Leftrightarrow \begin{cases} P_1 * P_2 * P_3 * P_4 * P_7 = 1(a) \\ \text{ou} \\ P_1 * P_2 * P_3 * P_4 * !P_4 * P_7 = 1(b) \\ \text{ou} \\ P_1 * P_2 * P_3 * P_4 * P_5 * P_7 * !P_8 = 1(c) \\ \text{ou} \\ P_2 * P_3 * P_4 * !P_4 * P_5 * P_7 * !P_8 = 1(d) \end{cases}$$

Les équations (b) et (d) ne sont pas solvables, puisqu'il est impossible d'avoir $P_i * !P_i = 1$! (il est impossible de déployer et ne pas déployer un même élément). Il ne nous reste donc plus que deux équations :

$$\begin{cases} P_1 * P_2 * P_3 * P_4 * P_7 = 1(a) \\ \text{ou} \\ P_1 * P_2 * P_3 * P_4 * P_5 * P_7 * !P_8 = 1(c) \end{cases}$$

Ces deux équations représentent chacune une solution qui répond à la situation de départ décrite par la Figure IV-7.

Les deux façons de respecter E_1 , E_2 et E_3 sont donc soit :

- de déployer les éléments P_1, P_2, P_3, P_4 , et P_7 ,
- de déployer les éléments $P_1, P_2, P_3, P_4, P_5, P_7$, mais en aucun cas il ne faut déployer P_8 .

Notre objectif étant de minimiser le nombre total d'éléments de Ref_2 à déployer, la première solution est la meilleure.

3. Synthèse et choix

Nous avons donc présenté deux méthodes de résolution différentes, qui permettent de résoudre notre système. Nous les comparons dans le Tableau IV-4 selon six critères différents :

- **principales étapes** : nous listons ici les principaux temps forts des deux méthodes de résolution envisagées,
- **compétences exigées** : il s'agit des diverses compétences techniques à posséder afin de mettre en œuvre les méthodes,
- **possibilité d'amélioration** : à travers ce critère, nous estimons le degré de complexité lié à l'implémentation éventuelle de nouvelles fonctionnalités,
- limites / inconvénients,
- avantages.

	Méthode A : optimisation par des variables booléennes	Méthode B : algorithme de recherche
Principales étapes	Conception d'une base de données pour stocker les relations Conception et réalisation d'un programme permettant le passage vers les formes canoniques disjonctives Utilisation d'un moteur d'optimisation pour coder le système et migration des données vers ce moteur Conception d'une base de données pour stocker les résultats obtenus par le moteur Conception d'une interface afin d'exploiter ces résultats	Conception d'une base de données pour stocker les relations Conception et réalisation d'un programme permettant le passage vers les formes canoniques disjonctives Conception et réalisation d'un programme permettant d'identifier le noyau d'éléments de Ref_2 à sélectionner Conception d'une base de données pour stocker les résultats obtenus par le moteur Conception d'une interface afin d'exploiter ces résultats
Compétences exigées	Maîtrise d'un SGBD (type MySQL) Maîtrise d'un moteur d'optimisation (type LINGO) Maîtrise d'un langage de programmation (type Perl)	Maîtrise d'un SGBD (type MySQL) Maîtrise d'un langage de programmation (type Perl)
Possibilité d'amélioration	Transition aisée vers l'optimisation d'une autre fonction objectif (charge de déploiement par exemple)	Transition plus complexe vers l'optimisation d'un autre critère Couplage aisé avec des méthodes type « <i>Branch and Bound</i> »

Limites / inconvenients	Le fonctionnement d'un moteur d'optimisation peut paraître « hermétique » : l'utilisateur du moteur ne connaît pas toutes les subtilités du code De nombreuses interfaces sont à prévoir Le temps de calcul augmente considérablement avec le nombre de variables	Demande de générer simultanément toutes les solutions possibles afin d'identifier la meilleure
Avantages	Ré-utilisabilité	Maîtrise quasi-totale du processus utilisé Peu d'interfaces à réaliser
Estimation de la charge de mise en œuvre	35 j.h	25 j.h

Tableau IV-4 : comparaison des deux méthodes envisagées

Le principal défaut de la première méthode réside dans l'utilisation même du moteur d'optimisation. Un tel moteur est généralement complexe et l'utilisateur ne maîtrise que les données d'entrée et n'a finalement que très peu la main sur l'optimisation elle-même.

Le principal défaut de la seconde méthode réside dans le fait qu'il faut générer toutes les solutions possibles afin d'identifier la plus pertinente. Si le cas applicatif comporte beaucoup de nœuds OU et XO, la génération de toutes les solutions possibles peut prendre du temps. Afin, d'y remédier, il est possible d'utiliser des méthodes d'optimisation comme le « *Branch and Bound* », qui permettent de filtrer d'emblée les solutions les plus mauvaises.

Les estimations¹ des charges de mise en œuvre étant relativement proches, nous avons fait le choix de la seconde méthode sur notre cas applicatif. Le chapitre suivant fournira les informations techniques relatives à la mise en œuvre et présentera les résultats obtenus.

¹ Il s'agit d'estimations intuitives qui n'ont pas été obtenues en utilisant les méthodes d'estimation classiques.

Chapitre V. APPLICATION INDUSTRIELLE ET RESULTATS OBTENUS

Rappelons que l'objectif premier de nos travaux est d'identifier les pratiques du CMMI et d'autres référentiels qualité informatique (Ref₂) qu'il faut déployer au sein d'une organisation qui souhaite ou doit respecter les exigences réglementaires pharmaceutiques (Ref₁). Ainsi, nous avons réalisé un mapping entre ces référentiels. La complexité des relations mises en jeu nous a amené à concevoir des liens utilisant des opérateurs logiques. Nous avons alors spécifié et programmé la seconde méthode présentée dans le chapitre précédent, pour nous permettre d'exploiter les résultats obtenus par les mises en correspondance.

Ce chapitre est organisé en trois parties (Figure V-1) :

- la première présente la méthodologie d'identification et de construction des relations entre bonnes pratiques informatiques et exigences réglementaires, ainsi que les modalités de mise en œuvre de la méthode de résolution (choix technologiques),
- la seconde est dédiée à la présentation des résultats obtenus : identification de ce qu'il faut déployer dans le CMMI et les autres référentiels de bonnes pratiques afin de couvrir un maximum d'exigences réglementaires,
- enfin, dans la troisième partie, nous présentons deux exploitations possibles des résultats obtenus.

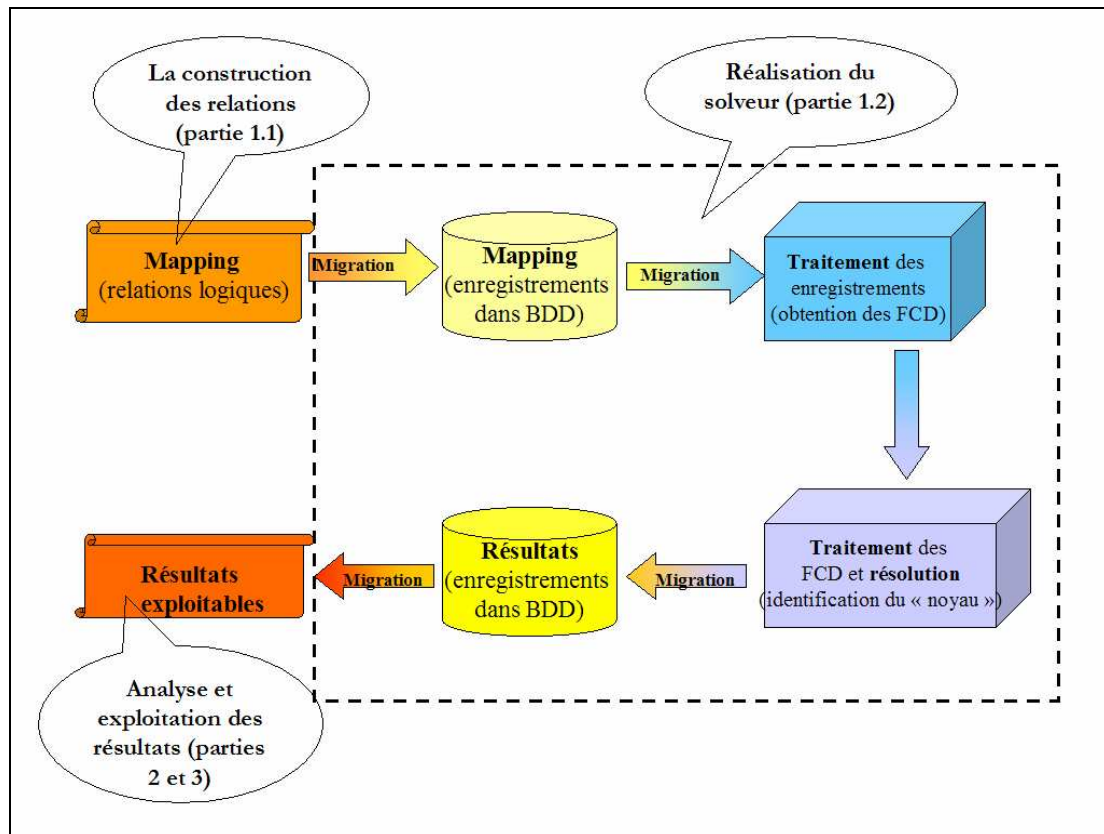


Figure V-1 : plan de la partie

1. La construction des relations et la réalisation du solveur

1.1. Construction des relations

Dans notre cas applicatif, les relations ont été établies « manuellement ». Une analyse en profondeur de chaque exigence réglementaire relative à notre périmètre, ainsi que de chaque pratique des référentiels applicables, a été réalisée. Grâce à la consultation de nombreux experts des différents domaines, nous avons pu établir les correspondances recherchées. Les activités réalisées sont précisées sur la Figure V-2.

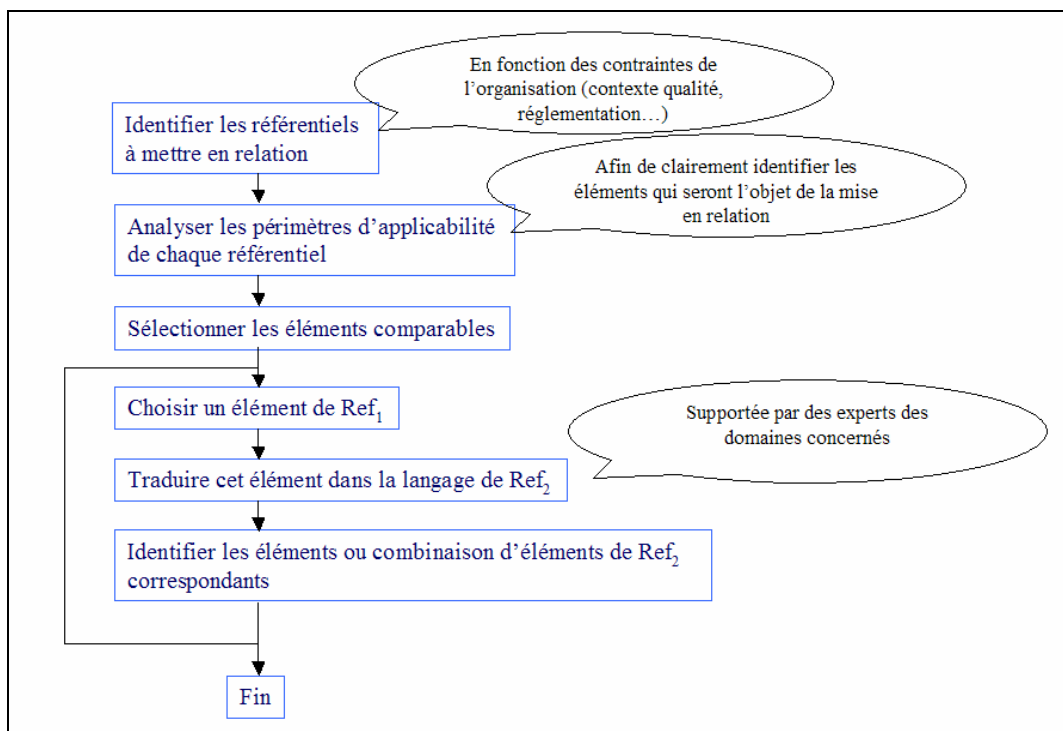


Figure V-2 : activités pour identifier les relations

Ainsi, dans notre cas, le travail consiste à traduire les exigences réglementaires en termes « métier » afin de pouvoir les comparer avec les bonnes pratiques informatiques. Cela a été réalisé avec des experts du domaine pharmaceutique, afin de clairement définir les sens des termes utilisés dans un contexte pharmaceutique, par rapport à leur sens dans le monde qualité.

Par exemple, les exigences réglementaires imposent aux producteurs de médicaments de procéder à des audits de fournisseurs afin de s'assurer que ces derniers travaillent « correctement ». Dans le monde pharmaceutique, « audit fournisseur » possède un sens bien plus vaste que dans le cadre « qualité informatique ». En plus d'évaluer les capacités d'un fournisseur sur certains processus, un audit fournisseur implique également, dans le monde pharmaceutique par exemple, d'analyser les méthodes d'amélioration des processus et implique la mise en place de plans d'amélioration et les activités de suivi des actions qui en découlent... Ainsi, un audit n'est pas seulement une

« photographie » à un instant « t » d'une organisation, mais il consiste aussi à évaluer un écart entre la maturité requise et la maturité actuelle de cette organisation et à proposer un « chemin » à suivre.

Un autre exemple significatif est celui de la « validation ». Dans le contexte pharmaceutique, le processus de validation est un processus formel, clairement défini, dont le but est d'autoriser ou non l'utilisation d'un système dans un processus pharmaceutique. Dans cet objectif, de nombreux tests doivent être réalisés (au niveau physique, au niveau fonctionnel ainsi qu'au niveau du processus qui va être supporté par le système informatisé). Pour le CMMI, la « validation » consiste simplement à comparer un système par rapport au besoin initial du client (« est-ce que j'ai construit ce qui était attendu ? »). Il y a donc une réelle différence entre les deux visions. Ainsi, lorsque les textes réglementaires pharmaceutiques imposent de valider les systèmes utilisés dans la production informatique, le déploiement des bonnes pratiques relatives au secteur clé « Validation » du CMMI n'est donc pas suffisant « tel quel ».

Ces « écarts lexicaux » sont à l'origine de bien des incompréhensions entre experts de domaines différents. Tout le travail consiste à pouvoir faire communiquer les deux mondes, grâce à l'identification claire de ces différences.

La recherche de correspondances entre ces exigences et les bonnes pratiques a été réalisée grâce à la participation de différents experts qualité de la DSI des Laboratoires Pierre Fabre. Une fois que toutes les relations ont été identifiées, il faut les traiter informatiquement, grâce à la seconde méthode présentée dans le chapitre précédent.

En termes de volumétrie, nous rappelons que le référentiel d'exigences est composé de cent soixante quinze éléments. Le CMMI est composé de cent quatre vingt cinq pratiques spécifiques et douze pratiques génériques. L'ISO 20000 propose environ deux cent trente pratiques. Pour des raisons de droits, nous ne pouvons reproduire ici les textes réglementaires et les référentiels de bonnes pratiques utilisés. Il nous est donc impossible de proposer l'ensemble des relations établies, qui composent le mapping.

La réalisation technique du solveur est présentée dans la partie suivante.

1.2. Réalisation du solveur

La solution technique se divise en six temps forts :

- élaboration des relations du mapping sous Excel,
- stockage des relations dans une base MySQL,
- traitement des enregistrements grâce à un programme développé en langage Perl,
- identification du noyau de pratiques à choisir grâce à un programme développé en langage Perl,

- stockage des résultats dans une base MySQL,
- exploitation des résultats sous Excel.

La figure suivante (Figure V-3) présente cette logique de résolution.

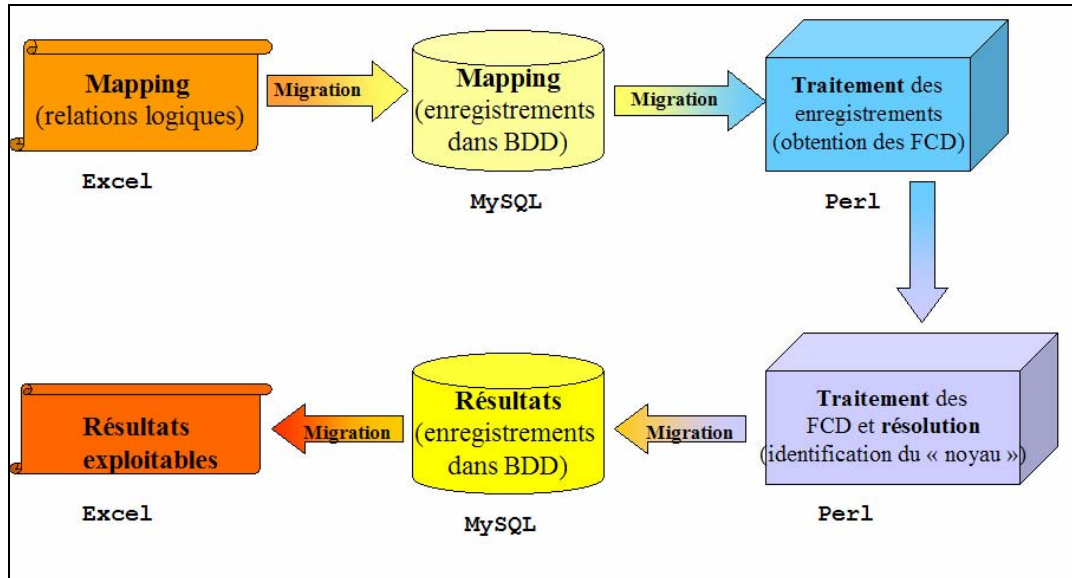


Figure V-3 : méthodes de résolution

Ainsi, le mapping a tout d'abord été réalisé sous Excel¹. Des requêtes SQL permettent de remplir une première base de données. Les enregistrements sont alors traités par un programme conçu en langage Perl², qui permet d'aboutir dans un premier temps aux formes canoniques disjonctives des relations logiques. Dans un second temps, le programme Perl déduit des formes canoniques disjonctives le noyau de pratiques à déployer dans le but de respecter toutes les exigences réglementaires. Ce résultat est alors stocké dans une nouvelle base de données afin d'être sauvegardé dans un format facilement lisible. Des requêtes SQL permettent finalement d'envoyer ces résultats sous Excel afin d'en faciliter l'exploitation.

¹ Le choix d'utiliser Excel se justifie aisément : logiciel quasi-universel, il permet d'obtenir des fichiers faciles à communiquer et à faire évoluer.

² Le choix du langage Perl s'explique également simplement : c'est un langage très adapté, grâce à sa syntaxe simple, pour des « petits » développements et des prototypes. Son utilisation est combinée avec MySQL. De plus, les compétences du Centre de Génie Industriel de l'Ecole des Mines d'Albi-Carmaux dans ce langage sont indéniables.

2. Résultats du mapping

Une fois que les relations sont construites et traitées par le solveur, le noyau minimal de bonnes pratiques est identifié. Nous accordons une importance particulière à la présentation des résultats du mapping entre le référentiel d'exigences et les pratiques du CMMI. En effet, comme il s'agit des activités relatives au cœur de métier de la DSI, il est logique d'y consacrer un effort plus conséquent. Nous présenterons ensuite plus synthétiquement les résultats du mapping entre le référentiel d'exigences réglementaires pharmaceutiques et l'ensemble des référentiels de bonnes pratiques informatiques relatifs à notre périmètre.

2.1. Mapping entre les pratiques du CMMI et les exigences réglementaires

Nous rappelons que le CMMI est composé de vingt cinq secteurs clés (PA) qui proposent au total cent quatre vingt cinq pratiques spécifiques (SP) et douze pratiques génériques (GP). La répartition de ces pratiques par niveaux de maturité de l'architecture étagée du CMMI est illustrée par la Figure V-4.

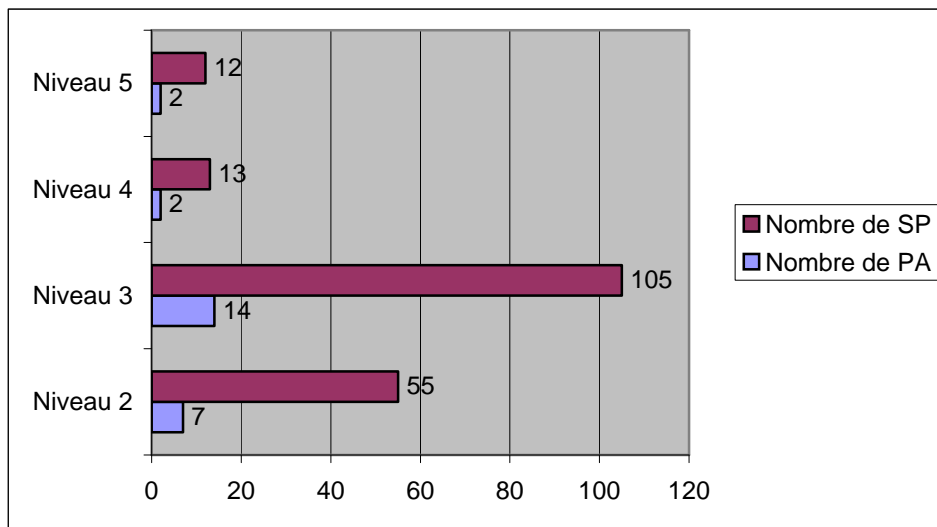


Figure V-4 : répartition des PA et SP par niveaux

L'exploitation des relations a permis d'établir que 63 % des exigences du référentiel pharmaceutique sont couvertes par les pratiques du CMMI (Figure V-5).

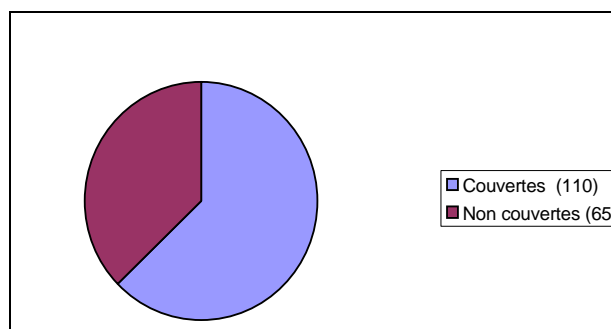


Figure V-5 : couverture des exigences par les pratiques du CMMI

Ainsi, sur les cent soixante quinze exigences du référentiel pharmaceutique, il est possible de trouver des équivalents dans le CMMI pour cent dix d'entre elles. La mise en place d'une démarche basée sur le CMMI ne permet donc pas de couvrir toutes les exigences réglementaires pharmaceutiques. Ces cent dix exigences sont couvertes soit :

- par une combinaison de pratiques qui appartiennent exclusivement au niveau de maturité 2 (soixante trois exigences),
- par une combinaison de pratiques qui appartiennent au niveau 2 et au niveau 3, ou exclusivement au niveau 3 (quarante sept exigences).

La Figure V-6 montre la répartition des exigences couvertes entre ces deux catégories.

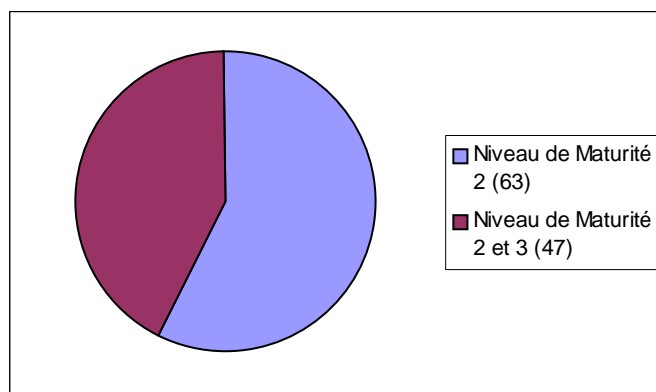


Figure V-6 : répartition des exigences couvertes par niveaux de maturité du CMMI

Plus particulièrement, le mapping met en jeu 84 % des pratiques du niveau 2 (Figure V-7) et 55 % des pratiques du niveau 3 (Figure V-8).

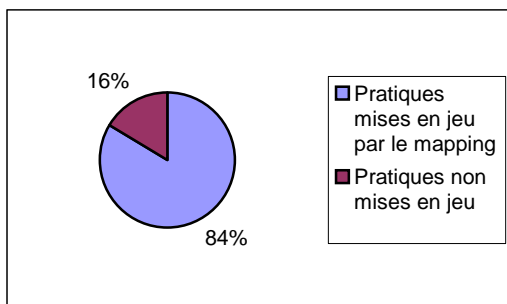


Figure V-7 : les pratiques du niveau 2 mises en jeu par le mapping

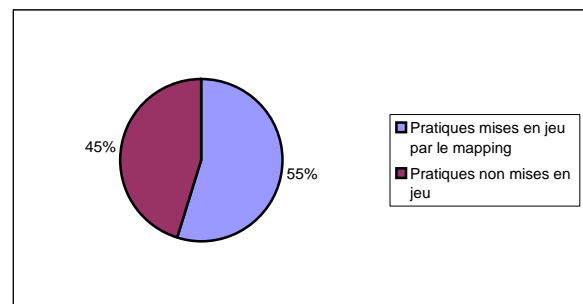


Figure V-8 : les pratiques du niveau 3 mises en jeu par le mapping

Le nombre de pratiques mises en jeu par secteur clé est proposé par la Figure V-9.

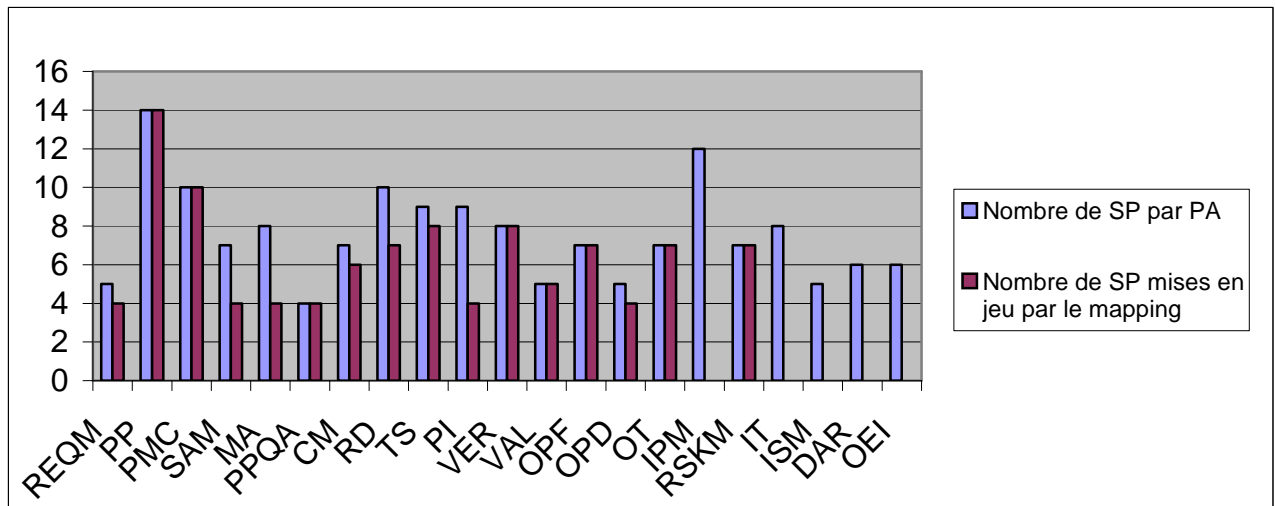


Figure V-9 : les pratiques mises en jeu par PA

Notons que les secteurs clés relatifs à la gestion de projet et ceux relatifs à l'ingénierie sont particulièrement représentés. Enfin, il nous a paru intéressant d'étudier le nombre de fois que les pratiques d'un secteur clé sont citées dans les relations établies. Ceci est l'objet de la Figure V-10. Cette figure propose, pour chaque secteur clé, la somme des citations des pratiques clés qui le composent.

Les secteurs clés PP (planification de projet), CM (gestion de la configuration), RD (développement des exigences), VER (vérification) et VAL (validation) sont les plus cités. Compte tenu de la répartition des exigences par catégories, ceci n'est pas étonnant. En effet, le secteur clé PP propose des pratiques qui permettent de s'assurer de la maîtrise des projets (formation des ressources, gestion de la documentation...). Le secteur clé CM propose les pratiques propres à la gestion de configuration et donc à la gestion des changements. Enfin, les secteurs clés RD, VER et VAL permettent de produire un système et de s'assurer qu'il fonctionne et qu'il correspond au besoin.

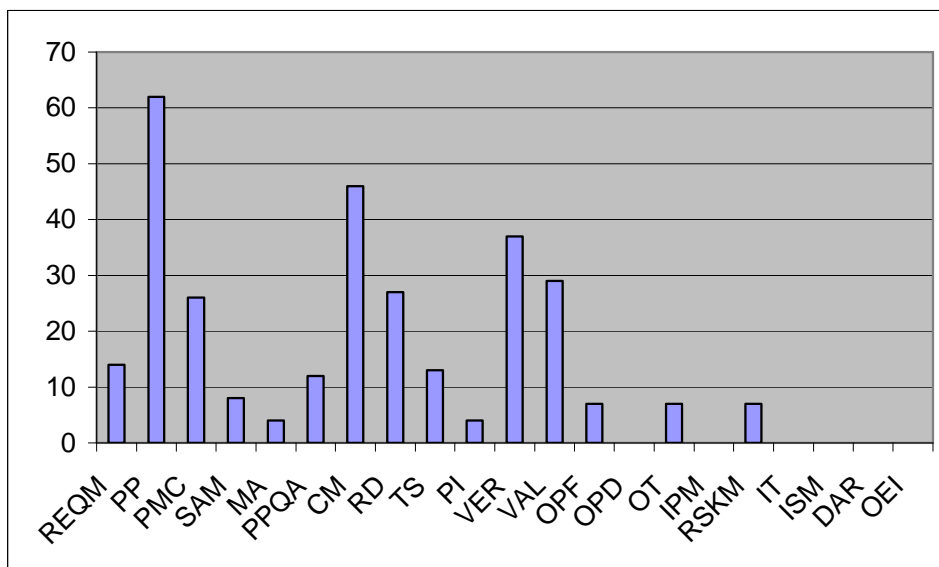


Figure V-10 : nombre de références par PA

Les exigences non-couvertes par le CMMI (qui sont au nombre de soixante cinq) se répartissent de la manière suivante (Figure V-11) :

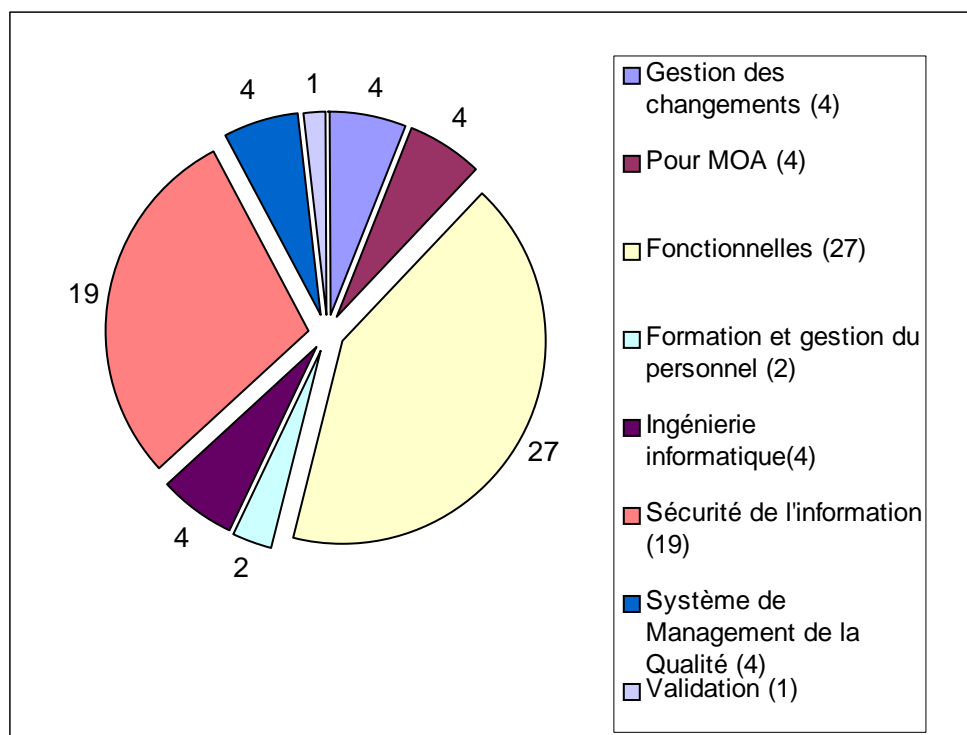


Figure V-11 : répartition des non-couvertures par catégories

D'une manière générale, les exigences non-couvertes concernent la sécurité des données (puisque c'est une problématique qui n'est que superficiellement abordée par le CMMI) ; l'exploitation des systèmes informatisés (la gestion des profils d'utilisateurs, des administrateurs, la maintenance des

serveurs, la supervision...) et enfin les fonctionnalités des systèmes. Ceci n'est pas une surprise, car ce ne sont pas des domaines couverts par le CMMI. Il faut pour cela intégrer les référentiels :

- ISO 20000,
- ISO 9000 : 2000,
- ISO 27000.

2.2. Résultats du mapping global

La généralisation de notre démarche à l'ensemble du contexte qualité nous a permis de conclure sur la complémentarité des différents référentiels par rapport aux exigences réglementaires. Ainsi, l'intégration de l'ISO 9000 : 2000 dans l'ensemble du cycle de vie des systèmes informatisés est une bonne solution :

- d'une part, pour les étapes de conception et de réalisation/suivi des systèmes informatisés, car elle est complémentaire au CMMI [Ferchichi, 2008]. Elle précise largement les rôles et caractéristiques d'un système qualité. C'est en ce sens qu'elle est d'ailleurs conseillée par l'ISPE dans [ISPE, 2001].
- d'autre part, pour les phases d'exploitation, car elle est intégrée dans l'ISO 20000.

L'ISO 20000 est une piste tout à fait raisonnable dans l'objectif de respect des exigences réglementaires. Notons que cette dernière est plus exigeante que la réglementation, notamment au niveau de la gestion du budget (ce qui est logique : pour la réglementation, qu'importe que le budget soit mal géré, du moment qu'il permet d'obtenir et de maîtriser le système désiré).

L'ISO 27000, qui a pour objet la sécurité, est une référence majeure de ce domaine. Les exigences réglementaires étant très « haut niveau » à ce sujet, un mapping « fin » semble être très complexe. Au vu des attendus de la norme, la complémentarité semble être totale.

Enfin, sur la Figure V-12, la zone « Spécifiques Pharma. », regroupe l'ensemble des exigences pour lesquelles il n'y a pas de correspondances dans les référentiels étudiés. Il s'agit principalement des exigences propres à la validation pharmaceutique, des exigences fonctionnelles et enfin des exigences MOA.

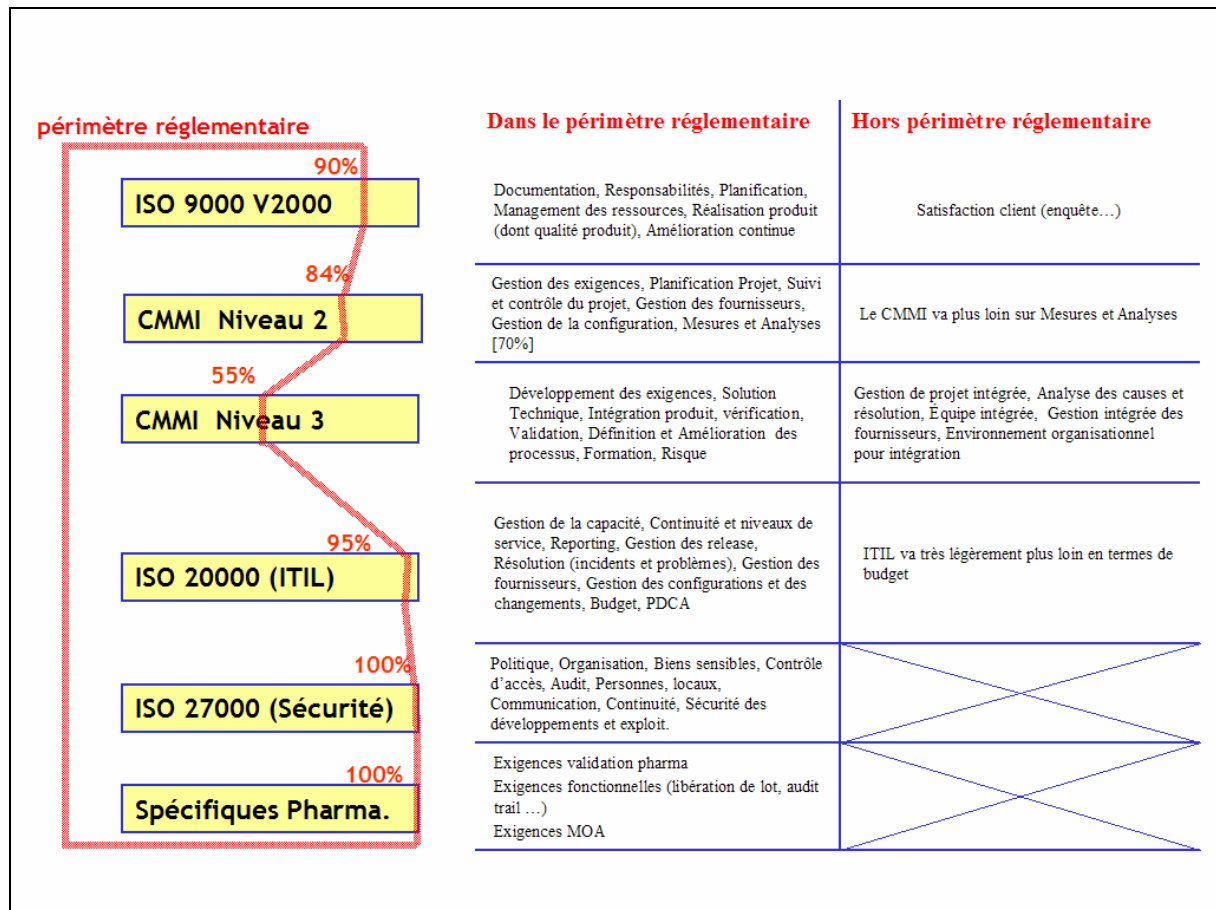


Figure V-12 : cadre qualité

Ce profil qualité définit donc la cible qu'il faut atteindre, en utilisant le CMMI pour la conception et la réalisation des systèmes informatisés, l'ISO 20000 pour leur exploitation, l'ISO 27000 pour la gestion de la sécurité et enfin l'ISO 9000 : 2000 pour le système de management de la qualité.

Ainsi, le respect de la réglementation pharmaceutique passe par le déploiement de 84 % des pratiques qui composent les secteurs clés du niveau de maturité 2 du CMMI. Les 16 % qui ne sont pas nécessaires sont principalement relatifs au secteur clé MA : « Mesures et Analyses ». De même, il faut déployer 55 % des pratiques qui composent les secteurs clés du niveau de maturité 3 du CMMI. De plus, 90 % de l'ISO 9000 : 2000, 95 % des pratiques de l'ISO 20000 et enfin 100 % de l'ISO 27000. Enfin, certaines exigences réglementaires ne sont pas couvertes par tous ces référentiels. Il s'agit principalement d'exigences fonctionnelles, qui sont par définition propres aux domaines pharmaceutiques.

La suite logique du travail est de comparer cette cible avec la situation actuelle de la DSI, ce qui permettra de définir les écarts et ensuite de proposer un plan d'action, permettant de combler ces écarts. L'élaboration de ce plan est détaillée dans la partie suivante.

3. Exploitation des résultats du mapping et du travail d'analyse des exigences réglementaires

3.1. Plan Qualité de la DSI

Cette partie décrit le travail que nous avons réalisé, au sein de la DSI des Laboratoires Pierre Fabre, afin d'identifier et de combler les écarts entre la cible (le respect des exigences réglementaires) et l'existant. Nous nous plaçons dans l'objectif de faire évoluer le Système de Management de la Qualité (SMQ), afin qu'il réponde aux exigences réglementaires pharmaceutiques.

Nous identifions deux dimensions à toute action d'amélioration :

- les actions qui consistent à apporter des modifications au Système de Management de la Qualité au niveau « documentaire » (création d'un nouveau document à instancier sur les projets, modification...) ou au niveau « processus » (ajout d'une description d'activité « attendue » que les équipes doivent réaliser...),
- les actions qui consistent à faire évoluer les pratiques au niveau « terrain » (formation des équipes projet à une activité technique précise, à une nouvelle technologie, déploiement de nouveaux processus...).

C'est ce que nous représentons sur la Figure V-13. L'existant est décrit dans la partie de gauche et la cible est décrite dans la partie de droite. L'existant se compose du SMQ (version 1) et des pratiques « terrain » (version 1). Notons qu'il est envisageable qu'à un instant t , le SMQ et les pratiques « terrain » ne soient pas alignés : certains processus peuvent être décrits théoriquement par le SMQ et non réalisés au niveau « terrain ». La cible est, quant à elle, construite à partir des exigences réglementaires pharmaceutiques du référentiel et des résultats du mapping.

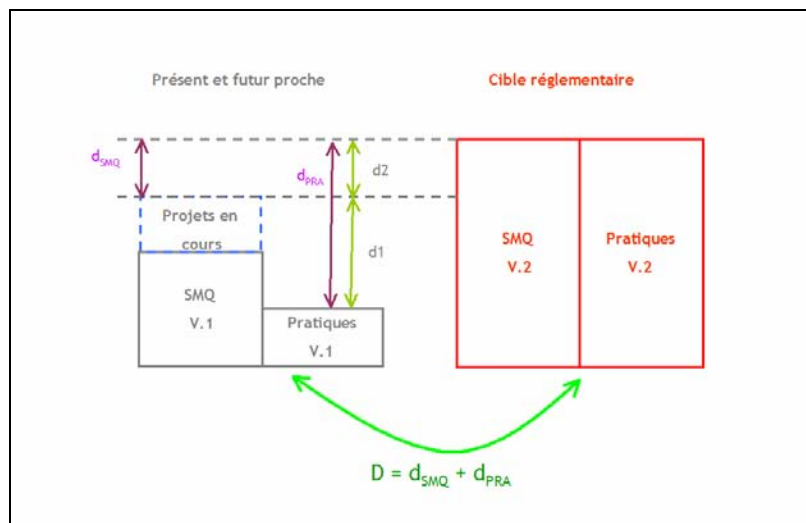


Figure V-13 : les dimensions de l'amélioration

Ainsi, l'exigence de « conformité réglementaire » impose d'une part, d'identifier ce qu'il faut faire évoluer dans le SMQ (faire passer le SMQ de la version 1 à la version 2 sur la Figure V-13, c'est-à-dire parcourir l'écart d_{SMQ}), pour ensuite faire évoluer le niveau terrain (parcourir l'écart d_{PRA}). L'objectif final est donc de faire parcourir à l'organisation les écarts d_{SMQ} et d_{PRA} .

Dans le cadre de nos travaux, nous nous concentrons sur l'écart d_{SMQ} : nous travaillons au niveau du SMQ. Le déploiement du SMQ (version 2) au niveau des pratiques « terrain » (c'est-à-dire combler l'écart d_{PRA}) n'est pas notre objectif.

Le SMQ de la DSI des Laboratoires Pierre Fabre est composé des éléments suivants :

- du Manuel Qualité (MAQ) : c'est un document qui décrit la politique qualité de l'organisation,
- des descriptions des processus métier de la DSI,
- des procédures, qui permettent de cadrer des activités à réaliser,
- des modes opératoires,
- des formulaires et documents types,
- des enregistrements (instanciations des documents types et des formulaires).

Ceci est représenté par la pyramide documentaire proposée dans la Figure V-14.

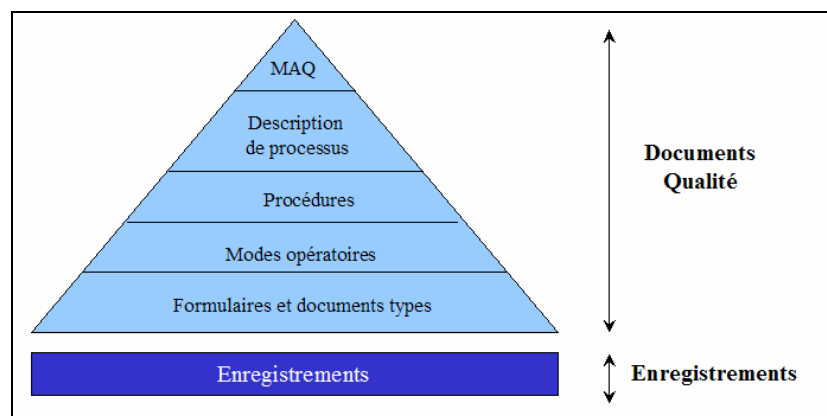


Figure V-14 : le système documentaire de la DSI [Carrasco, 2006]

Le MAQ, en tant que document « chapeau », est situé à la pointe de la pyramide. Les autres composants « héritent » de ce dernier, puisqu'il décrit les politiques qualité applicables.

Le mapping entre les pratiques du CMMI et les exigences réglementaires nous a permis d'identifier l'ensemble des pratiques nécessaires dans le but de respecter la réglementation

pharmaceutique. La DSI a déjà atteint le niveau 2 CMMI, donc son Système de Management de la Qualité est censé être compatible avec les exigences réglementaires liées, dans le mapping, aux pratiques associées aux secteurs clés du niveau 2. Il ne nous reste donc qu'à analyser les exigences qui ont été liées, en partie ou totalement, à des pratiques associées à des secteurs clés du niveau de maturité 3.

L'objectif est donc, pour chaque exigence restante, de comprendre son impact sur le SMQ en termes de :

- politique qualité,
- description de processus, des rôles et des responsabilités,
- procédures et modes opératoires (création, modification, suppression),
- formulaires et documents types (création, modification, suppression).

C'est un travail laborieux, mais nécessaire. Celui-ci a été réalisé en exploitant, dans un premier temps, les relations établies. En effet, le CMMI propose, par exemple, pour chaque pratique, une liste de produits du travail et un léger détail de ceux-ci. Il est donc possible d'utiliser ces informations afin de caractériser la situation cible désirée. Dans un second temps, nous avons consulté des experts du domaine pharmaceutique et des experts qualité. Nous avons également exploité les guides disponibles, qui proposent des documents types (le GAMP propose par exemple, un modèle de cahier des charges et un modèle de dossier de spécifications techniques).

Ainsi, nous avons pu identifier une liste d'actions à réaliser dans le but de faire évoluer le système de management de la qualité. Ces actions ont été regroupées par chantier, qui ont ensuite été priorisés les uns par rapport aux autres. Enfin, les chantiers ont été planifiés et budgétés, ce qui a permis de construire le plan qualité de la DSI à cinq ans. Il s'agit de l'application principale du résultat industriel de nos travaux. A celui-ci s'ajoute un résultat indirect, qui se base sur l'exploitation du référentiel d'exigences réglementaires pharmaceutiques.

3.2. Exploitation du référentiel d'exigences réglementaires : construction des diagrammes d'Ishikawa

Le référentiel d'exigences réglementaires pharmaceutiques que nous avons obtenu présente deux avantages :

- il peut servir de base de communication entre les différentes parties prenantes d'un projet pharmaceutique mettant en jeu un ou plusieurs systèmes informatisés. C'est en effet un outil plus simple à diffuser qu'un ensemble de textes réglementaires, qui peuvent apparaître comme « hermétiques » aux « non-initiés »,
- il est facile à manipuler. A ce titre, il est aisé de s'en servir afin d'identifier, par exemple, les risques pharmaceutiques associés à un système informatisé. C'est ce que nous proposons de faire.

La validation d'un système informatisé (annexe) consiste à s'assurer que les risques pharmaceutiques en lien avec ce système sont maîtrisés. La validation d'un système se fait en regard des exigences réglementaires, c'est-à-dire qu'il faut vérifier que la réglementation qui s'applique à ce système a été respectée. C'est donc le non-respect des exigences qui est à l'origine des risques, c'est-à-dire, qui en est la cause.

La méthode Ishikawa [Ishikawa, 1984] permet de recenser l'ensemble des causes aboutissant à un effet défini, en les représentant sous forme d'arêtes de poisson. Chaque arête va représenter une famille de causes pouvant aboutir à l'effet principal. Les ramifications peuvent se faire sur plusieurs niveaux, c'est-à-dire que les diagrammes font apparaître des causes, des sous-causes...

De plus, il est habituel de coupler cette méthode avec la méthode des 5M, qui consiste à organiser les causes d'un effet, en fonction de cinq composantes : milieu, méthodes de travail, matériel, main-d'œuvre, matière. Cela permet de standardiser la classification des causes. C'est le choix que nous avons fait (cf. Figure V-15).

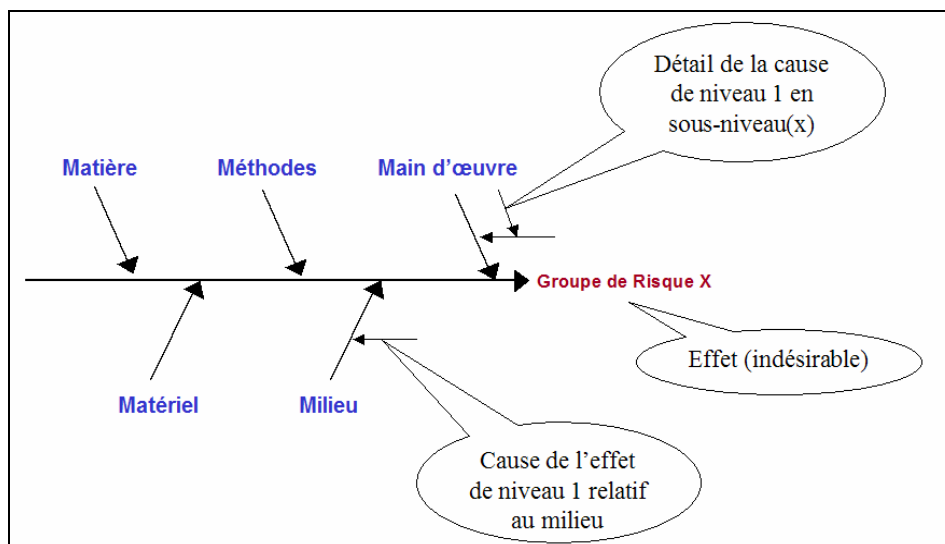


Figure V-15 : diagramme d'Ishikawa et 5M

Une analyse fine des exigences nous a permis d'identifier trois risques pharmaceutiques principaux, qui seront les « effets » des diagrammes d'Ishikawa. Nous différencions donc :

- (1) Risques liés aux données et enregistrements critiques (perte, altération, diffusion...)
- (2) Risques liés aux processus de gestion des changements (changements non maîtrisés...)
- (3) Risques liés aux processus qualité lors des différentes phases du cycle de vie du système (déviations, non-conformités...).

Le Tableau V-1 présente, pour chaque groupe de risques, les significations des 5M.

	Groupe 1 (perte, altération ou diffusion des données critiques)	Groupe 2 (gestion des changements)	Groupe 3 (écarts qualité)
Milieu	Emplacement géographique : où sont stockées / traitées les données ?	Gestion des accès aux systèmes de gestion des changements	Ce dans quoi les systèmes qualité sont déployés
Matériel	Systèmes qui traitent les données	Fonctionnalités des systèmes de gestion des changements et d'audit trail	
Matières	Les données considérées comme critiques par la réglementation	Les données considérées comme critiques par la réglementation	Enregistrements qualité
Méthodes	Procédures / listes / méthodologies	Procédures / listes / méthodologies	Procédures / listes / méthodologies
Main-d'œuvre	hommes	hommes	hommes

Tableau V-1 : les 5M et les risques

Pour le risque 1, les résultats obtenus sont montrés sur la Figure V-16.

De tels diagrammes peuvent facilement être exploités pour la validation d'un système informatisé : il suffit d'identifier les branches des diagrammes qui s'appliquent au système à valider et ensuite de les parcourir afin de vérifier que chaque cause a bien été couverte par un test de qualification d'installation (QI), opérationnelle (QO) ou de performance (QP).

Dans le cadre de la thèse, nous avons utilisé les diagrammes d'Ischikawa en phase opérationnelle sur un projet de validation d'un système informatisé. Il s'agissait d'un système complexe de supervision de l'infrastructure informatique, des applications informatiques et du réseau. La comparaison des résultats obtenus par l'utilisation de nos diagrammes avec ceux obtenus par la méthode actuellement utilisée par la DSI des Laboratoires Pierre Fabre (basée sur l'expérience des experts et sur le retour d'expérience des projets précédemment réalisés) nous permet de conclure sur la pertinence de notre démarche. Ainsi, notre méthode permet d'identifier un spectre plus large de tests et apparaît donc comme un outil pertinent vis-à-vis de la réalisation des protocoles de tests.

Nous avons fait un travail similaire pour chaque groupe de risques. Les autres résultats sont présentés dans en annexe.

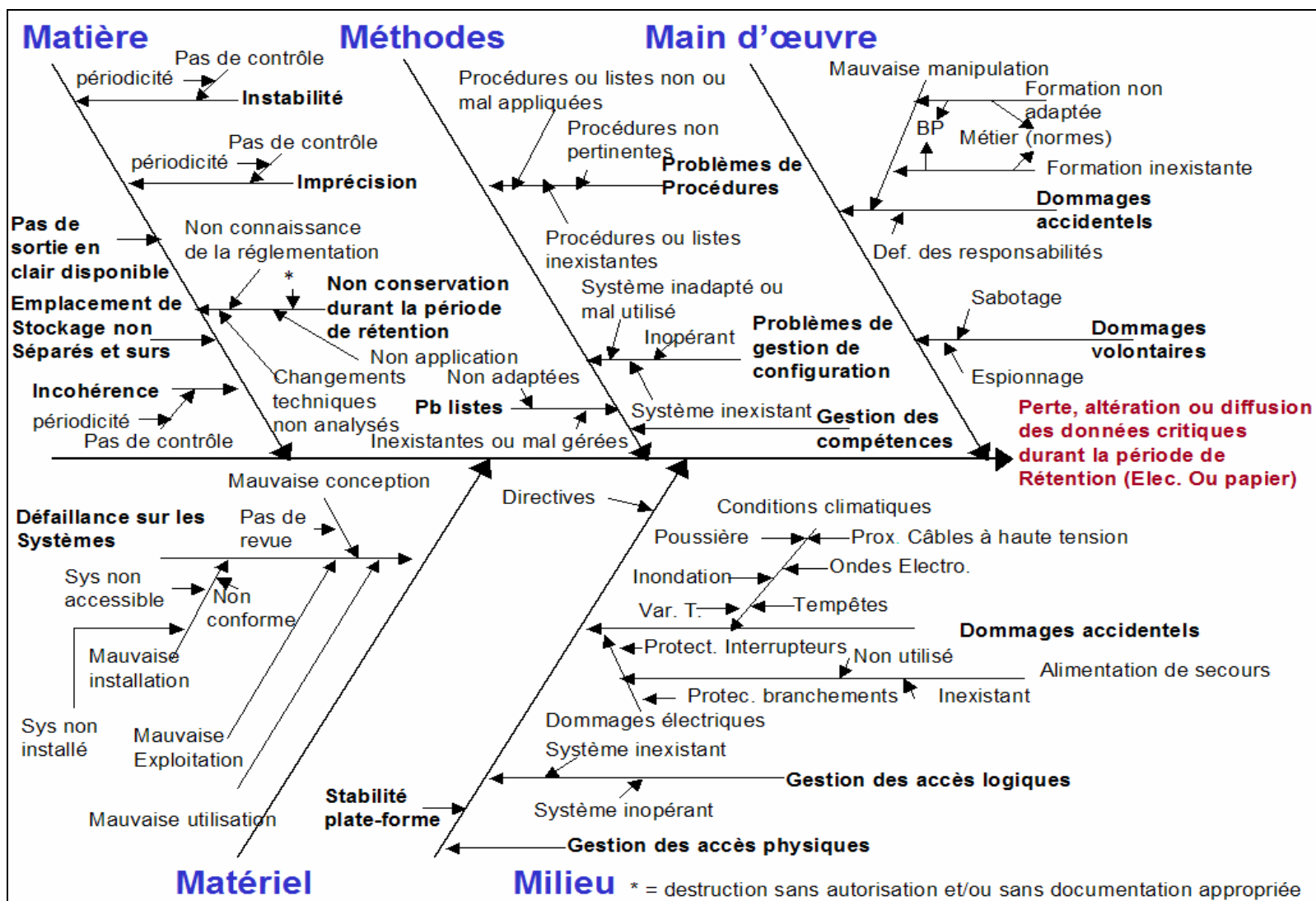


Figure V-16 : exemple de diagramme d'Ishikawa

CONCLUSION ET PERSPECTIVES

Rappelons que nos travaux étaient motivés par la problématique industrielle suivante : « jusqu'à quel point les exigences réglementaires pharmaceutiques sont-elles couvertes par les référentiels de bonnes pratiques informatiques que sont le CMMI, l'ISO 20000, l'ISO 27000 et l'ISO 9000 : 2000 ? ».

Pour répondre à cette question, nous nous sommes rattachés à l'IDM, ce qui nous a permis d'utiliser la notion de mapping de modèles. Nous avons mis en valeur le fait que la réalisation d'un mapping demande d'avoir, au préalable, des modèles des deux mondes à mettre en relation (chapitre 2). L'obtention de ces modèles a donc été l'objectif des chapitres 2 et 3. Nous avons tout d'abord cherché à obtenir un modèle générique des référentiels de bonnes pratiques informatiques, puis un modèle des exigences réglementaires pharmaceutiques. Enfin, grâce à ces modèles, nous avons pu caractériser le mapping voulu (chapitre 4). Le dernier chapitre (chapitre 5) nous a permis de présenter les résultats industriels recherchés, ainsi que deux applications de ceux-ci.

La démarche proposée se veut être suffisamment générique pour pouvoir être appliquée à d'autres référentiels. Le choix de référentiels concurrents permettrait alors de générer un plus grand nombre de liens OU et XO, ce qui mettrait davantage en valeur la pertinence de la méthode d'identification du noyau de pratiques à déployer.

De nombreuses perspectives sont identifiées pour la suite de nos travaux. Nous classons ces perspectives selon les trois parties de la Figure V-1. Ainsi, au niveau de la construction des relations du mapping, deux perspectives nous semblent particulièrement intéressantes :

- il serait intéressant d'affiner les mises en relation entre les exigences réglementaires et les pratiques proposées par l'ISO 27000 et l'ISO 20000. En effet, n'étant pas spécialisés sur ces référentiels très spécifiques, les relations que nous proposons sont parfois trop globales, ce qui a logiquement un impact sur la qualité des résultats obtenus,
- un moteur d'analyse lexicale pourrait être conçu afin d'automatiser la mise en relation des différents concepts. Cette perspective est particulièrement intéressante, car elle permettrait de comparer les liens établis « manuellement » et ceux générés « automatiquement ».

Les trois perspectives suivantes sont relatives aux fonctionnalités du solveur :

- concernant le choix du noyau de pratiques à déployer, une piste pertinente serait de rechercher un optimum différent. En effet, dans le chapitre IV, nous avons précisé que nous souhaitions choisir la solution qui propose le nombre minimum de pratiques à déployer. Nous pourrions envisager, par exemple, d'affecter à chaque pratique une

évaluation de « l'effort de déploiement nécessaire », qui serait donc une valeur relative à l'organisation. L'optimisation pourrait alors consister à rechercher la solution la moins « coûteuse ». Une autre piste pourrait être de faire le lien entre les pratiques mises en jeu par le mapping et les risques pharmaceutiques listés dans les diagrammes d'Ishikawa, à travers les exigences réglementaires. Suite à une analyse risques (analyse d'impact, probabilité d'occurrence...), il serait alors possible, pour une organisation, de prioriser les pratiques entre-elles et donc de leur affecter un poids relatif à leur importance. L'optimisation pourrait alors consister à chercher le meilleur compromis entre effort de déploiement et couverture des risques pharmaceutiques,

- nous avons travaillé sur la notion de liens « intra-référentiels », c'est-à-dire des liens entre éléments du même référentiel (par exemple, le déploiement de la pratique X du CMMI impose le déploiement préalable de la pratique Y). Cette réflexion reste à préciser, mais cela semble être une piste de travail tout à fait pertinente,
- une fois que la meilleure solution a été choisie, il faudrait enregistrer les nœuds par lesquels celle-ci passe. Ceci permettrait de pouvoir « remonter » dans l'arbre des relations et ainsi obtenir un premier degré de bi-directionnalité. Il se pose alors des questions de robustesse : si une exigence réglementaire est annulée, quel est l'impact de cette modification sur le nœud de pratiques à déployer ? Les nœuds par lesquels la meilleure solution passe peuvent avoir changés, et l'optimisation est alors à refaire. Il serait intéressant de développer cette fonctionnalité dans le prototype. De plus, un second niveau de bi-directionnalité est envisageable : il s'agit de reconstruire tous les liens dans le sens bonnes pratiques → exigences afin de disposer de deux mappings indépendants. La différence de granularité des informations comparées pose alors un problème : dans ce sens, les liens vont-ils avoir une signification réelle ?

Enfin, la dernière perspective concerne l'exploitation des résultats. Un travail de veille réglementaire et scientifique est obligatoire afin de maintenir la cohérence entre notre noyau de bonnes pratiques et la réglementation pharmaceutique applicable.

LISTE DES ACRONYMES UTILISES

5M	Milieu, Méthodes de travail, Matériel, Main-d'œuvre, Matière
AFIS	Association Française d'Ingénierie Système
AFSSAPS	Agence Française de Sécurité Sanitaire des Produits de Santé
AMM	Autorisation de Mise sur le Marché
AQ	Assurance Qualité
BDD	Base De Données
BP	Bonne Pratique
BPC	Bonnes Pratiques Cliniques
BPD	Bonnes Pratiques de Distribution
BPF	Bonnes Pratiques de Fabrication
BPL	Bonnes Pratiques de Laboratoire
CAR	Causal Analysis and Resolution
CDC	Cahier des charges
CEE	Code of Federal Regulations
CETIC	Centre d'Excellence en Technologies de l'Information et de la Communication
CFR	Center for Food Safety and Applied Nutrition
cGMP	Current Good Manufacturing Practices
CIM	Computation Independent Model
CM	Configuration Management
CMM	Capability Maturity Model
CMMI	Capability Maturity Model Integration
CMMM	Configuration Management Maturity Model
CNRS	Centre national de la recherche scientifique
CQ	Contrôle Qualité
DAR	Decision Analysis and Resolution
DOD	Department of Defence
DSI	Direction du Système d'Information
DSL	Domain Specific Language
EMA	European Medicines Agency

FCD	Forme Canonique Disjonctive
FDA	Food and Drug Administration
FR	Federal Register
GAMP	Good Automated Manufacturing Practice
GCP	Good Clinical Practices
GDP	Good Distribution Practices
GG	Generic Goal
GLP	Good Laboratory Practices
GMP	Good Manufacturing Practices
GP	Generic Practice
IDEF0	Integration Definition for Function Modeling
IDM	Ingénierie Dirigée par les Modèles
IEEE	Institute of Electrical and Electronics Engineers
IPPD	Integrated Project and Product Management
ISM	Integrated Supplier Management
ISO	International Organization for Standardization
ISPE	International Society for Pharmaceutical Engineering
IT	Integrated Teaming
ITIL	Information Technology Infrastructure Library
MA	Measurement and Analysis
MAQ	MANuel Qualité
MCD	Modèle Conceptuel de Données
MDA	Model Driven Architecture
MOA	Maîtrise d'ouvrage
MOF	Meta Object Facility
OCDE	Organisation de Coopération et de Développement Économique
OEI	Organizational Environment for Integration
OGC	Office of Government Commerce
OID	Organizational Innovation and Deployment
OMG	Object Management Group
OPD	Organizational Process Definition

OPF	Organizational Process Focus
OPM	Organizational Process Management
OPM3	Organizational Project Management Maturity Model
OPP	Organizational Process Performance
OT	Organizational Training
OWPL	Observatoire Wallon des Pratiques Logicielles
PA	Process Area
PDCA	Plan, Do, Check, Act
PDM	Platform Dependant Model
PI	Product Integration
PIM	Platform Independent Model
PM	Platform Model
PMC	Project Monitoring and Control
PP	Project Planning
PPQA	Process and Product Quality Assurance
PSM	Platform Specific Model
QC	Qualification de Conception
QI	Qualification d'Installation
QO	Qualification Opérationnelle
QP	Qualification de Performance
RD	Requirements Development
REQM	Requirements Management
RSQM	Risk Management
SADT	Structured Analysis and Design Technic
SAM	Supplier Agreement Management
SCAMPI	Standard CMMI Appraisal Method for Process Improvement
SD	Spécifications Détaillées
SEI	Software Engineering Institute
SFD	Spécifications Fonctionnelles Détaillées
SFG	Spécifications Fonctionnelles Générales
SG	Specific Goal

SGBD	Système de Gestion de Bases de Données
SLA	Service Level Agreement
SMQ	Système de Management de la Qualité
SMR	Service Médical Rendu
SP	Specific Practice
SPEM	Software Process Engineering Metamodel
SPI	Software Process Improvement
SPICE	Software Process Improvement and Capability dEtermination
SQL	Structured Query Language
SSII	Société de Services en Ingénierie Informatique
ST	Spécifications Des Tests
TMM	Testing Maturity Model
TS	Technical Solution
TU	Tests Utilisateurs
UE	Union Européenne
UML	Unified Modeling Language
VAL	Validation
VER	Verification

BIBLIOGRAPHIE

A

- [AFIS, 2004] Association Française d'Ingénierie Système, *Glossaire de base de l'ingénierie des systèmes - Version expérimentale*, 48 p., 2004
- [AFNOR, 2000] Agence Française de NORmalisation, *Norme NF EN ISO 9001*, 52 p., 2000
- [AFSSAPS, 2007] Agence Française de Sécurité Sanitaire des Produits de Santé, *Glossaire des vigilances*, [En-ligne], URL : http://afssaps.sante.fr/pdf/5/glossaire_vigilances.pdf. Consulté en 2007, 19 p., 2007
- [Akehust *et al.*, 2002] D.H. Akehust, S. Kent, *A relational approach to defining transformations in a metamodel*, Proceedings of the 5th International Conference on the UML 2002, Vol. 2460, pp. 243-258, 2002
- [Alexandre *et al.*, 2003] S. Alexandre, N. Habra, *UML Modeling of five Process Maturity Models*, [En ligne] : URL : <http://www.cetic.be/IMG/pdf/UMLModelingOfFiveProcessModels-V1-3.pdf>. Consulté en 2006, 42 p., 2003
- [ANM, 1998] Ateliers Nationaux du Médicament, *La qualité*, Editions : John Libbey Eurotext, 109 p., 1998
- [Armatte, 2005] M. Armatte, *La notion de modèle dans les sciences sociales : anciennes et nouvelles significations*, Mathématiques et sciences humaines, spécial modèles et méthodes mathématiques dans les sciences sociales : apports et limites, n° 172, 33 p., 2005, [En ligne], mis en ligne le 22 avril 2006. URL : <http://msh.revues.org/document2962.html>. Consulté le 7 avril 2008
- [Arnold, 1999] J. Arnold, *Les nouvelles normes ISO 9000 : 2000*, Transfusion Clinique et Biologique, volume 6, Issue 4, pp. 247 – 253, 1999

B

- [Bamford *et al.*, 1993] R.C. Bamford, W.J. Deibler II, *Comparing, contrasting ISO 9001 and the SEI capability maturity model*, IEEE Computer, Volume 26, pp. 68–70, 1993
- [Basavapathruni, 2007] O. Basavapathruni, *Computers in Pharmaceutical Technology*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 732 – 741, 2007
- [Basque, 1993] R. Basque, *Modèle d'évolution des capacités logiciel (CMM) du SEI : un guide de pratiques clés pour les informaticiens soucieux de qualité !*, [En ligne] : URL : http://w3.crim.ca/files/documents/services/pratiquesLogicielles/modele_evolution_capacites.pdf, Consulté en 2006, 1993
- [Bataille *et al.*, 2001] V. Bataille, X. Castellani, *Métamodélisation et ingénierie des systèmes d'information*, dans C. Cauvet, C. Rosenthal-Sabroux, *Ingénierie des systèmes d'information*, pp.149 – 174, Editions : Hermes, 2001
- [Bénard *et al.*, 2005] J.L. Bénard, F. Merand, *Approches pragmatiques pour industrialiser le développement d'applications – Gérer le cycle de vie du développement logiciel : une mise en œuvre concrète*, 88 p., 2005
- [Berry, 2007] I. Berry, *Good Manufacturing Practices (GMPs) : An Overview*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 732 – 741, 2007
- [Berthier *et al.*, 2005] D. Berthier, C. Morley, M. Maurice-demourieux, *Enrichissement de la modélisation des processus métiers par le paradigme des systèmes multi agents*, 21p., 2005
- [Bézivin *et al.*, 2004] J. Bézivin, M. Blay, M. Bouzhegoub, J. Estublier, J.M. Favre, S. Gérard, J.M. Jézéquel, *Rapport de synthèse de l'AS CNRS sur le MDA (Model Driven Architecture)*, 16 p., 2004, [En ligne] URL : <http://www.irisa.fr/triskell/publis/2004/Jezequel04e.pdf>. Consulté le 7 avril 2008
- [Bézivin, 2004] J. Bézivin, *Sur les principes de base de l'ingénierie des modèles*, RSTI – L'objet – 10/2004, Où en sont les objets ?, pp. 124 - 156, 2004, [En ligne] URL : <http://www.sciences.univ-nantes.fr/lina/atl/www/papers/LObjet2004.pdf>. Consulté le

13 avril 2008

- [Boltzmann, 1902] L. Boltzmann, *Models*, Encyclopedia Britannica, 1902
- [Bottoni *et al.*, 2006] P. Bottoni, F. D'Antonio, M. Missikoff, *Towards a unified view of model mapping and transformation*, EMOI-INTEROP 2006, 10 p., 2006

C

- [Cabanne, 2001] M.D. Cabanne, *Points d'ancrage de la validation des systèmes : cycle de vie et cahier des charges*, STP pharma pratiques, Volume 11 (5), pp. 240 – 249, 2001
- [Caplat *et al.*, 2003] G. Caplat, J.L. Sourrouille, *Considerations about model mapping*, Wisme, 6 p., 2003
- [Carrasco, 2006] D. Carrasco, *Manuel Qualité, Version 1.1*, document interne – Direction du Système d'Information des Laboratoires Pierre Fabre, 24 p., 2006
- [Chadonnet *et al.*, 2002] A. Chardonnet, D. Thibaudon, *Le guide du PDCA de Deming : progrès continu et management (Plan Do Check Act)*, Editions : Eyrolles, 368 p., 2002
- [Chandelon, 2002] M. Chandelon, *Convergence : Rêve ou Réalité*, [En ligne], URL : www.lux-ias.lu/PDF/Chandelon.pdf. Consulté en 2005, 24 p., 2002
- [Chapron, 2006] J. Chapron, *L'urbanisation organisationnelle : méthodes et aides à la décision pour piloter l'évolution du système d'information de l'entreprise*, Thèse de doctorat, Ecole Nationale Supérieure des Mines de Saint Etienne & Université Jean Monnet de Saint Etienne, 259 p., 2006
- [Charon *et al.*, 1996] I. Charon, A. Germa, O. Hudry, *Méthodes d'optimisation combinatoire*, Editions : Masson, 268 p., 1996

- [Chast, 1995] F. Chast, *Histoire contemporaine des médicaments*, Editions : La découverte, 385 p., 1995
- [Chaveau, 2004] S. Chaveau, *Genèse de la « sécurité sanitaire » : les produits pharmaceutiques en France aux XIXE et XXE siècles*, Expertise et protection, Issue 51, 30 p., 2004
- [Colin, 2001] J.N. Colin, *Stratégies d'enregistrement en Europe*, Sang Thrombose Vaisseaux, Volume 13, pp. 67 - 71, 2001
- [Combemale *et al.*, 2006] B. Combemale, X. Crégut, A. Caplain, B. Coulette, *Modélisation rigoureuse en SPEM de procédé de développement*, LMO 2006, 16 p., 2006
- [CRAM, 1993] Centre de Recherche Informatique de Montréal, *Modèle d'évolution des capacités logiciels, version 1.1*, [en ligne] ; adresse URL : http://w3.crim.ca/files/documents/services/pratiquesLogicielles/modele_evolution_capacites.pdf. Consulté en 2006, 93 p., 1993
- [Crosby, 1979] P. Crosby, *Quality is free : The art of making quality certain*, Editions : McGraw-Hill, 309 p., 1979

D

- [D'Antonio, 2005] F. D'Antonio, *Deliverable MoMo.2 TG MoMo Roadmap*, Rapport INTEROP (Interoperability Research for Network Enterprises Applications and Software), 72 p., 2005, [En ligne] URL : http://interop-vlab.eu/ei_public_deliverables/interop-nœ-deliverables/tg3-model-morphisms/DTG3.1/?searchterm=morphism. Consulté le 12 décembre 2007
- [Degroote *et al.*, 2003] D. Degroote, L. Benaïche, *Responsabilité civile du fait d'un produit de santé défectueux : nouvelles perspectives au regard des arrêtés du 25 avril 2002 de la Cour de justice des communautés européennes*, Médecine & Droit, Issue 58, pp. 73 - 84, 2003

- [Deguil *et al.*, 2006] R. Deguil, D. Gourc, H. Pingaud, *Etude bibliographique des modèles de maturité*, Rapport semestriel de thèse, Centre de Génie Industriel de l'Ecole des Mines d'Albi Carmaux, 87 p., 2006
- [Deguil *et al.*, 2007a] R. Deguil, D. Gourc, H. Pingaud, *Détermination d'un ensemble minimal de bonnes pratiques informatiques pour couvrir un référentiel réglementaire de l'industrie pharmaceutique*, 7^{ème} édition du congrès international pluridisciplinaire QUALITA'2007, Tanger, pp. 329 – 336, 2007
- [Deguil *et al.*, 2007b] R. Deguil, D. Gourc, H. Pingaud, *A first study about the determination of the overlap between different quality repositories : problem formulation and methodology*, 20th International Conference on Software & Systems Engineering and their Applications ICSSEA'07, Paris, 8 p. , 2007
- [Deguil *et al.*, 2007c] R. Deguil, D. Gourc, H. Pingaud, *Détermination d'un ensemble minimal de bonnes pratiques informatiques pour couvrir un référentiel réglementaire de l'industrie pharmaceutique*, 7^{ème} congrès international de génie industriel « L'entreprise innovante en réponse à la mondialisation : défis et perspectives en génie industriel », Trois-rivières (Canada), 2007
- [Deguil *et al.*, 2008] R. Deguil, D. Gourc, H. Pingaud, *Determining the overlap between quality repositories and regulatory requirements for the pharmaceutical industry : problem formulation and application*, INSIGHT – Special Feature : The Best of France – Forum académique et RobAFIS, p. 17, 2008
- [Dent, 2007] N. Dent, *Good Laboratory practices (GLPs) : An Overview*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 1931 – 1940, 2007
- [DiMasi *et al.*, 2003] J. DiMasi, R. Hansen, *The price of innovation : New estimates of drug developing costs*, Journal of Health Economics, 22(2), pp. 151 – 185, 2003
- [Doumeingts, 1990] G. Doumeingts, *Méthodes pour concevoir et spécifier les systèmes de production*, actes du colloque international CIM'90, Bordeaux, pp. 89-103, 1990
- [Dumont, 2006] C. Dumont, *ITIL pour un service informatique optimal*, Editions Eyrolles, 317 p., 2006

E

- [Eurotil, 2004] Eurotil, *ITIL : Introduction et historique*, [En ligne] ; URL : http://www.eurotil.com/pages/itil/00-docs/01-itil_presentation.pdf. Consulté en 2005, 19 p., 2004

F

- [Faure, 1979] R. Faure, *Précis de recherche opérationnelle*, Editions : DUNOD, 466 p., 1979
- [Féraud *et al.*, 2001] S. Féraud, P. André, *Analyse de risque et stratégie de validation des systèmes informatisés en recherche et développement pharmaceutique*, STP Pharma Pratiques, Volume 11 (5), pp. 250 – 254, 2001
- [Ferchichi, 2008] A. Ferchichi, *Contribution à l'intégration des processus métier : application à la mise en place d'un référentiel qualité multi-vues*, Thèse de doctorat de l'Ecole Centrale de Lille, 348 p., 2008
- [Ferry, 2000] S. Ferry, *L'usage du médicament*, Editions : TEC & DOC – Editions Médicales Internationales, 641 p., 2000
- [Fouassier, 2006] E. Fouassier, *Les fondements juridiques de la responsabilité du producteur de spécialités pharmaceutiques : vers un retour à la raison ?*, Médecine & Droit, Issue 78, pp. 103 - 109, 2006
- [Frazer *et al.*, 2002] P.Fraser, J.Moultrie, M.Gregory, *The use of maturity models / grids as a tool in assessing product development capability*, IEEE International Engineering Management Conference, Cambridge, 19-20 August 2002
- [Frkic, 2000] I. Frkic, *La structure et le fonctionnement du modèle américain de protection sanitaire : Food and Drug Administration*, Notre Europe, 91p., 2000
- [Froman *et al.*, 2003] B. Froman, C. Gourdon, *Dictionnaire de la qualité*, Editions : AFNOR, 225 p., 2003

G

- [Ginisti, 1997] J.P. Ginisti, *La logique combinatoire*, Editions : Presses Universitaires de France, Paris, 126 p., 1997
- [Giunchiglia *et al.*, 2003] F. Giunchiglia, P. Shvaiko, *Semantic matching*, Technical report # DIT-03-013, Université de Toronto, 19 p., 2003, [En ligne] URL : <http://eprints.biblio.unitn.it/archive/00000381/01/013.pdf>. Consulté le 12 septembre 2007
- [Goffinet, 2006] J. Goffinet, *CMMI ou comment maîtriser vos développements*, [En ligne] URL : <http://www.zdnet.fr/techupdate/infrastructure/0,39020938,39148384-1,00.htm>. Consulté en 2005, 2004
- [Gonzales-Ramirez, 2006] N. Gonzales-Ramirez, Thèse ECP/PSA : Mesure de la maturité des projets : une approche pour améliorer le pilotage des projets automobiles, Présentation du GRD-MACS, Valenciennes, 14 p., 2006
- [Gonzalo-Perez, Henderson-Sellers, 2007] C. Gonzalo-Perez, B. Hederson-Sellers, *Modeling software development methodologies : a conceptuel foundation*, The Journal of Systems and Software 80, pp. 1778 – 1796, 2007
- [Guarino, 2007] R. Guarino, *Good Clinical Practices (GCPs) : An overview*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 1925 – 1930, 2007

H

- [Hausmann, 2007] K. Hausmann, *Deliverable DTG 3.2 Report and workshop, Final version of the toolbox*, Rapport INTEROP (Interoperability Research for Network Enterprises Applications and Software), 55 p., 2007, [En ligne] URL : <http://interop->

vlab.eu/ei_public_deliverables/interop-nœ-deliverables/tg3-model-morphisms/dtg3-2-report-and-workshop/?searchterm=morphism. Consulté le 12 décembre 2007

[Hoffmann *et al.*, 1998]

A. Hoffmann, J. Káhny-Simonius, M. Plattner, V. Schmidli-Vckovski, C. Kronseder, *Computer system validation : an overview of official requirements and standards*, Pharmaceutica Acta Helevtia, Volume 72, pp. 317 – 325, 1998

[Huang *et al.*, 2006]

S. Huang, W. Han, *Selection priority of process areas based on CMMI continuous representation*, Information & Management 43, pp. 297 – 307, 2006

I

[Ishikawa, 1984]

K. Ishikawa, *Le tqc ou la qualité à la japonaise*, Editions : AFNOR, 195 p., 1984

[ISO, 1994]

International Organization for Standardization, *ISO 8402 : 1994 : Management de la qualité et assurance de la qualité*, version daté 01/04/1994 (norme annulée), 1994

[ISO, 1995a]

International Organization for Standardization, *SPICE Part 1 : Concepts and Introductory Guide, Version 1.00*, 27 p., 1995

[ISO, 1995b]

International Organization for Standardization, *SPICE Part 9 : Vocabulary, Version 1.00*, 27 p., 1995

[ISO, 1995c]

International Organization for Standardization, *SPICE Part 2 : A Model for Process Improvement, Version 1.00*, 27 p., 1995

[ISO, 2000]

International Organization for Standardization, *ISO 9001 : 2000 : Systèmes de management de la qualité – Exigences*, 24 p., 2000

[ISO, 2002]

International Organization for Standardization, *ISO 19011 : Lignes directrices pour l'audit des systèmes de management de la qualité et/ou de management environnemental*, 2002

- [ISO, 2005a] International Organization for Standardization, *ISO/IEC 20000-1 : Information technology – Service management – Part 1 : Specification*, 24 p., 2005
- [ISO, 2005b] International Organization for Standardization, *ISO/IEC 20000-1 : Information technology – Service management – Part 2 : Code of practice*, 42 p., 2005
- [ISO, 2005c] International Organization for Standardization, *ISO/IEC 27002 : Information Technology. Code of practices for information security management*, 115 p., 2005
- [ISPE, 2001a] International Society for Pharmaceutical Engineering, *The Good Automated Manufacturing Practice (GAMP) Guide for Validation of Automated Systems in Pharmaceutical Manufacture*, Editions : ISPE, 2001
- [ISPE, 2001b] International Society for Pharmaceutical Engineering, *Pharmaceutical Engineering Guides for New and Renovated Facilities, Volume 5 Commissioning and qualification*, Editions : ISPE, 135 p., 2001
- [itSMF, 2004] itSMF, *An Introductory Overview of ITIL*, Version 1.0a, 42 p., 2004
- [Izza, 2006] S. Izza, *Intégration des systèmes d'information industriels : une approche flexible basée sur les services sémantiques*, Thèse de doctorat de l'Ecole Nationale Supérieure des Mines de Saint-Etienne, Conférence annuelle, 390 p., 2006

J

- [Jalote, 1999] P. Jalote, *CMM in Practice : Processes for Executing Software Projects at Infosys*, The SEI Series in Software Engineering, Editions : Addison-Wesley, 1999
- [Joly, 2001] P. Joly, *Validation des systèmes informatisés : aperçu des réglementations en vigueur et des guides d'application*, STP pharma pratiques, Volume 11 (5), pp. 225 – 228, 2001

- [Jouault, 2006] F. Jouault, *Contribution à l'étude des langages de transformation de modèles*, Thèse de doctorat, Université de Nantes, Ecole Centrale de Nantes & Ecole des Mines de Nantes, 190 p., 2006, [En ligne] URL : http://www.bu.univ-nantes.fr/sciences/docnum/2006/jouault_frederic.pdf. Consulté le 28 avril 2008

K

- [Kalfoglou *et al.*, 2003] Y. Kalfoglou, M. Schorlemmer, *Ontology mapping : the state of the art*, The Knowledge Engineering Review, Volume 18, Issue 1, pp. 1 – 31, 2003
- [Kleppe *et al.*, 2003] A. Kleppe, J. Warmer, W. Bast, *MDA explained : the model driven architecture : practice and promise*, Reading, Addison-Wesley, 154 p., 2003

L

- [Laboratoires Pierre Fabre, 2004] Laboratoires Pierre Fabre, *Les réglementations pour les industries de la santé & la validation des systèmes informatisés*, Document interne, 100 p., 2004
- [Le Moigne, 1990] J.L. Le Moigne, *La modélisation de systèmes complexes*, Editions : Dunod, 192 p., 1990
- [Lemesle, 1998] R. Lemesle, *Transformation rules based on meta-modeling*, IEEE EDCO-98, La Jolla, Californie, pp. 113-122, 1998
- [Lepmets, 2007] M. Lepmets, *Evaluation of Basic Project Management Activities : Study In Software Industry*, Tampere University Of Technology,

- Publication 669, 233 p., 2007
- [Levendovski *et al.*, 2002] T. Levendovskin G. Karsai, M. Maroti, A. Ledeczzi, H. Charaf, *Model reuse with meta-model based transformations*, Proceedings of the 7th International Conference on Software Reuse : Methods, Techniques and Tools, pp. 166 – 178, 2002
- [Liao *et al.*, 2005] L. Liao, Y. Qu, H. Leung, *A Software Process Ontology and its Application*, Proceedings of the 4th International Semantic Web, Galway, Ireland, 10 p., 2005
- [Lipschutz, 1973] S. Lipschutz, *Algèbre linéaire : cours et problèmes*, Editions : Mc Graw-Hill, 334 p., 1973
- [Lissandre, 1990] M. Lissandre, *Maitriser SADT*, Editions : Armand Colin, 219 p., 1990
- [Lopes *et al.*, 2006] D. Lopes, S. Hammoudi, J. Bézivin, F. Jouault, *Mapping Specification in MDA : From theory to Practice*, Interoperability of Enterprise Software and Applications, pp. 253 – 264, 2006
- [Lopez, 2005] P. Lopez, *Cours de graphes*, 50 p., 2005, [En ligne] URL : <http://www.laas.fr/~lopez/cours/GRAPHES/GraphesPDF.zip>
- [Ludewig, 2003] J. Ludewig, *Models on software engineering - an introduction*, Software Systems Modeling 2, pp. 5 – 14, 2003

M

- [Manasse, 2007] H. Manasse, *Health Care Systems : Within the United States*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 1985 – 1995, 2007
- [Marçal *et al.*, 2007] A. Marçal, B. de Freitas, F. Soares, A. Belchior, *Mapping CMMI Project Management Process Areas to SCRUM Practices*, Proceedings of the 31st IEEE Software Engineering Workshop, pp. 13- 22, 2007

- [Marvie, 2003] R. Marvie, *Vers des patrons de méta-modélisation*, Techniques et Sciences Informatiques, Volume 23, num 10/2004, pp. 1355 – 1382, 2004
- [Massot, 1999] F. Massot, *La dynamique PDCA dans une entreprise, Une philosophie de l'action pour le XXIème siècle*, Conférence annuelle, Paris, 8 juin 1999
- [Mathieu, 2000] S. Mathieu, *Normes qualité ISO 9000 – Version 2000*, Techniques de l'ingénieur, 15 p., 2000
- [Mathieu, 2002] S. Mathieu, *Comprendre les normes ISO 9000 Version 2000*, Éditions AFNOR, 195 p., 2002
- [Miler et al., 2004] J. Miler, J. Görski, *Identifying Software Project Risks with the Process Model*, Proceedings of the 17th International Conference « Software & Systems Engineering and their Applications », 2004
- [Morley et al., 2005] C. Morley, J. Hughes, B. Leblanc, O. Hughes, *Processus métiers et Système d'Information*, Editions : Dunod, 240 p., 2005
- [Morley, 2002] C. Morley, *La modélisation des processus : typologie et proposition utilisant UML*, Processus et Systèmes d'information – Journées ADELI, Paris, France, 2002
- [Muller, 1997] P.L. Muller, *Modélisation objet avec UML*, Editions : Eyrolles, 420 p., 1997
- [Mutafelija et al., 2003a] B. Mutafelija, H. Stromberg, *ISO 9001 :2000 – CMMI 1.1 Mappings*, [En ligne], URL : <http://www.sei.cmu.edu/cmml/adooption>, Consulté en 2005, 31 p., 2003
- [Mutafelija et al., 2003b] B. Mutafelija, H. Stromberg, *Systematic Process Improvement Using ISO 9001 : 2000 and CMMI*, Editions : Artech House, Inc., Norwood, MA, 324 p., 2003
- [Mutafelija, 2001] B. Mutafelija, *Software Process Improvement : Synergy Between ISO 9001 : 2000 and CMMI*, SERG Conference, New Orleans, LA, 2001

N

- [Niazi *et al.*, 2005] M. Niazi, D. Wilson, D. Zowghi, *A maturity model for the implementation of software process improvement : an empirical study*, The Journal of Systems and Software 74, pp. 155 - 172, 2005
- [Nomia, 2005] Nomia, *Dossier Itil et CMMi : la qualité au cœur des DSI*, La lettre de Nomia, 7 pages, 2005

O

- [OCDE, 1995] Organisation de Coopération et de Développement Économiques, *Document consensus sur les BPL, Application des principes des BPL aux systèmes informatiques*, 17 p., 1995
- [OMG, 2002] Object Management Group, *Meta Object Facility (MOF) Specification*, Version 1.4, 358 p., 2002
- [OMG, 2003] Object Management Group, *MDA Guide*, Version 1.0.1, 62 p., 2003
- [OMG, 2005A] Object Management Group, *Software Process Engineering Metamodel Specification*, Version 1.1, 99 p., 2005, [En ligne] URL : <http://www.omg.org/docs/formal/05-01-06.pdf>. Consulté le 25 avril 2008
- [OMG, 2005B] Object Management Group, *MOF QVT Final Adopted Specification*, 2005, [En ligne] URL : <http://www.omg.org/docs/ptc/05-11-01.pdf>. Consulté le 29 avril 2008
- [OMS, 1997] Organisation Mondiale de la Santé – Vaccins et Produits biologiques, *Guide OMS des normes relatives aux bonnes pratiques de fabrication (BPF)*, [En ligne] URL : <http://www.who.int/vaccines-documents>, 148 p., 1997

P

-
- [Panetto, 2006] H. Panetto, *Meta-modèles et modèles pour l'intégration et l'interopérabilité des applications d'entreprise de production*, Habilitation à diriger des recherches, Université Henri Poincaré de Nancy, 124 p., 2006
- [Paty, 1994] M. Paty, *Notes sur les modèles et la modélisation*, Journée de l'association française d'informatique théorique (AFIT) n° spécial de la lettre de l'AFIT, Paris, pp. 25 - 34, 1994
- [Paulk et al., 1993] M.C. Paulk , B. Curtis, M.B. Chrissis, C.W. Weber, *The Capability Maturity Model for Software*, 26 p., 1993
- [Paulk, 1993] M.C. Paulk, *Comparing ISO 9001 and the capability maturity model for software*, Software Quality Journal 2, pp. 245 – 256, 1993
- [Paulk, 1994] M.C. Paulk, *A Comparison of ISO 9001 and the capability maturity model for software*, Software Engineering Institute, CMU/SEI-94-TR-12, 1994
- [Paulk, 1995] M.C. Paulk, *How ISO 9001 compares with the CMM*, IEEE Software 12, pp. 74 – 83, 1995

R

-
- [Rahm et al., 2001] E. Rahm, P.A. Bernstein, *A survey of approaches to automatic schema matching*, The VLDB journal 10 / Digital Object Identifier (DOI), pp. 334 – 350, 2001
- [Reifer, 2000] D.J. Reifer, *The CMMi it's formidable*. Journal of Systems and Software 50, Volume 50, Issue 2, pp. 97-98, 2000
- [Renard et al., 2003] L. Renard, G.E. St-Amant, *Capacité, capacité organisationnelle et capacité dynamique : une proposition de définitions*, 18 p., 2003

S

-
- [SEI, 1995] Software Engineering Institute, *A System Engineering Capability Maturity Model*, version .1.1, 297 p., 1995
- [SEI, 1997] Software Engineering Institute, *Integrated Product Development Capability Maturity Model, Draft version 0.98*, 1997
- [SEI, 2001] Software Engineering Institute, *People Capability Maturity Model (P-CMM)*, Version 2.0, Pittsburgh, 735 p., 2001
- [SEI, 2002a] Software Engineering Institute, *Capability Maturity Model Integration, Staged Representation*, Version 1.1, Pittsburgh, 729 p., 2002
- [SEI, 2002b] Software Engineering Institute, *Capability Maturity Model Integration, Continuous Representation*, Version 1.1, Pittsburgh, 724 p., 2002
- [SEI, 2002c] Software Engineering Institute, *Software Acquisition Capability Maturity Model (SA-CMM)*, Version 1.03, Pittsburgh, 134 p., 2002
- [SEI, 2002d] Software Engineering Institute, *CMMI-SE/SW V. 1.1 to SW-CMM V. 1.1 Mapping*, 57 p., 2002
- [Sheard, 1997] S. Shread, *The Frameworks Quagmire, A Brief Look*, Proceedings of INCOSE, 1997
- [SOFAQ, 1991] Société Française d'Assurance de la Qualité, *Validation informatique*, 24 p., 1991
- [SQI, 2001a] Software Quality Institute, *CMMI – Continuous representation Mapping to ISO 15504 – 2*, [En ligne], URL : <http://www.sqi.gu.edu.au/cmmi/indexFrameset.html>, 29 p., 2001
- [SQI, 2001b] Software Quality Institute, *CMMI – Staged representation Mapping to ISO 15504 – 2*, [En ligne], URL : <http://www.sqi.gu.edu.au/cmmi/indexFrameset.html>, 50 p., 2001

- [Studer, 2006] S. Studer, *La prise en charge du risque de développement par l'assurance*, Mémoire rédigé en vue de l'obtention du Master 2 Droit de la prévention des risques et responsabilités, Université de haute Alsace, 135 p., 2006

T

- [Terrasse *et al.*, 2005] M.N. Terrasse, M. Savonnet, E. Leclercq, T. Grison, G. Becker, *Points de vue croisés sur les notions de modèle et métamodèle*, Premières journées sur l'Ingénierie Dirigée par les Modèles, Paris, 13 p., 2005
- [The Standish Group, 1994] The Standish Group, *The Chaos Report*, 9 p., 1994
- [Théroude, 2002] F. Théroude, *Formalisme et système pour la représentation et la mise en œuvre des processus de pilotage des relations entre donneurs d'ordre et fournisseurs*, thèse de doctorat de l'Institut National Polytechnique de Grenoble, 2002
- [Touzi, 2007] J. Touzi, *Aide à la conception de Système d'Information Collaboratif support à l'interopérabilité des entreprises*, Thèse de doctorat, Institut National Polytechnique de Toulouse & Centre de Génie Industriel de l'Ecole des Mines d'Albi Carmaux, 248 p., 2007, [En ligne] URL : <http://www.univ-valenciennes.fr/GDR-MACS/these/TheseJihedTouzi.pdf>. Consulté le 10 janvier 2008

V

- [Vauquier, 1997] D. Vauquier, *Le plan qualité logiciel*, Editions : AFNOR, 228 p., 1997

[Vernadat, 1999] F. Vernadat, *Techniques de modélisation en entreprise : applications aux processus industriels*, Editions Economica, France, 129 p., 1999

[Villeneuve *et al.*, 2001] F. Villeneuve, Y. Soler, Référentiel Qualité, Guide méthodologique, *Présentation du cycle de vie d'un système d'information*, 6 p., 2001, , [En ligne] URL : <http://www.dsi.cnrs.fr/conduite-projet/principes/cycle%20de%20vie/guide-cycle-vie.pdf>, Consulté en juin 2008

W

[Wenger, 2005] E. Wenger, *La théorie des communautés de pratique, apprentissage, sens et identité*, Editions : Les Presses de l'Université, Canada, XX p., 2005

[Wertheimer *et al.*, 2007] A. Wertheimer, S. Kong, *Health Care Systems : Outside the United States*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 1977 – 1984, 2007

[Wikipedia, 2007] Wikipedia, *Object Management Group*, 2007, [En ligne] URL : http://fr.wikipedia.org/wiki/Object_Management_Group, Consulté en juin 2008

[Wikipedia, 2008a] Wikipedia, *Ingénierie Dirigées par les Modèles*, 2008, [En ligne] URL : http://fr.wikipedia.org/wiki/Ing%C3%A9nierie_dirig%C3%A9e_par_les_mod%C3%A8les, Consulté en juin 2008

[Wikipedia, 2008b] Wikipedia, *Modèles*, 2008, [En ligne] URL : <http://fr.wikipedia.org/wiki/Mod%C3%A8le>, Consulté en juin 2008

[Wikipedia, 2008c] Wikipedia, *Bonnes Pratiques*, 2008, [En ligne] URL : http://fr.wikipedia.org/wiki/Bonne_pratique, Consulté en juin 2008

- [Wikipedia, 2008d] Wikipedia, *Patron de Conception*, 2008, [En ligne] URL : http://fr.wikipedia.org/wiki/Patron_de_conception, Consulté en juin 2008
- [Williams, 2007] R. Williams, *Food and Drug Administration : Role in Drug Regulation*, Encyclopedia of Pharmaceutical Technology, Editions : Informa Healthcare, pp. 1779 – 1790, 2007
- [Winter, 2000] S.G. Winter, *The satisficing principle en capability learning*, Strategic Management Journal, 21, pp. 981 – 996, 2000
- [Wreth, 1993] L.H. Wreth, *Introduction to Software Process Improvement*, [En-ligne], URL : <http://www2.umassd.edu/SWPI/curriculummodule/em8.part2.pdf>. Consulté en 2005, 11 pages, 1993

Y

- [Yoo *et al.*, 2006] C. Yoo, J. Yoon, B. Lee, C. Lee, J. Lee, S. Hyun, C. Wu, *A unified model for the implementation of both ISO 9001 : 2000 and CMMI by ISO – certified organizations*, Journal of Systems and Software, Volume 79, Issue 7, pp. 954 - 961, 2006

ANNEXES

LA QMMG.....	165
LE CMM.....	168
LES PLANS DES DEUX LIVRES DE LA NORME ISO 20000	172
LA VALIDATION PHARMACEUTIQUE.....	173
DIAGRAMMES D'ISHIKAWA.....	177

La QMMG

	Incertitude	Réveil	Vision éclairée	Sagesse	Certitude
Attitude de la direction	Ne perçoit pas la qualité comme instrument de management	Comprend l'intérêt, mais ne met rien en œuvre	Réalise son apprentissage et apporte son soutien	Participe directement aux actions qualité	Intègre les outils de la qualité comme éléments essentiels du management
Niveau d'organisation de la qualité	Limité à : la production, la conception, l'évaluation et le tri	Leader qualité en place. Actions limitées à l'évaluation et au tri	Reporting du département qualité vers la Direction, participation au management	Le responsable de la qualité est un cadre. Actions préventives fortes	La responsable appartient à la direction, et est focalisé sur la prévention
La gestion des problèmes	Problèmes traités quand découverts et souvent partiellement résolus	Des équipes traitent les problèmes avec une vision à court terme	Problèmes résolus dans l'ordre et actions correctives régulières	Problèmes identifiés à leur source	A l'exception de rares cas, prévention des problèmes.
Coût de la non-qualité en pourcentage de ventes	Environ 20%	Détectée : 5% Effective : 15%	Détectée : 8% Effective : 12%	Détectée : 6.5% Effective : 8%	Détectée : 2.5% Effective : 2.5%
Actions d'amélioration de la qualité	Pas d'actions planifiées	Réactives et court terme	Programme en 14 étapes en cours d'implantation	Poursuite du programme en 14 étapes et entrée dans la zone des certitudes	Actions continues d'amélioration de la qualité
Bilan de l'attitude de l'entreprise	« Nous ignorons pourquoi nous avons des problèmes de qualité »	« Aurons nous toujours des problèmes de qualité. »	Du fait de l'engagement du management et des programmes d'amélioration de la qualité, identification et traitement des problèmes de qualité	Prévention permanente contre les défauts	« Nous savons pourquoi nous n'avons pas de problèmes relatifs à la qualité »

Tableau A-1 : la QMMG

Le programme en quatorze points, qui permet de progresser à travers la grille est le suivant :

- **engagement de la Direction.** L'ensemble de la Direction de l'entreprise doit être convaincu du besoin de l'amélioration de la qualité et exprimer clairement son engagement à toute la société. Ceci doit être accompagné par un document écrit définissant la politique qualité de l'entreprise et exprimant « ce que chacun doit faire en respect des exigences ou effectuer

officiellement des évolutions de celles-ci de sorte qu'elles atteignent ce que nous pouvons faire et répondons au besoin du client ».

- ***les équipes d'amélioration de la qualité.*** La Direction doit mettre en place dans chaque entité de l'entreprise une équipe qui supervise l'amélioration de la qualité. Le rôle de l'équipe est d'évaluer ce qui est nécessaire au niveau de son département et de le mettre en œuvre tout en respectant la politique générale de qualité de l'entreprise.
- ***la mesure de la qualité.*** Des indicateurs qualité sont à mettre en place de manière à identifier les besoins d'amélioration. En comptabilité, par exemple, un indicateur peut être le pourcentage des retards de traitement, en développement le nombre de litiges techniques remontés par produit, aux achats le nombre de rejets de commandes dus à l'imprécision des spécifications et aux services généraux le nombre d'heures perdues du fait des pannes des équipements.
- ***évaluation du coût de la non-qualité.*** Les équipes d'amélioration de la qualité feront une estimation des coûts de non-qualité de manière à identifier les zones prioritaires où les actions conduites seront immédiatement profitables.
- ***prise de conscience des nécessités de la qualité.*** Une prise de conscience des nécessités de la qualité doit se faire au sein du personnel. Il doit être amené à comprendre l'importance du respect des spécifications, le coût des non-conformités. Ces messages doivent être véhiculés par leurs responsables immédiats (après une formation de ceux-ci) en s'appuyant sur des outils tels que des cassettes vidéo, des livres, et des posters.
- ***actions correctives.*** Les opportunités pour les actions correctives sont déclenchées par les étapes 3 et 4, tout comme par des discussions parmi les employés. L'idée doit toujours être véhiculée par les responsables immédiats et les problèmes traités au sein de l'équipe si possible. Si nécessaire, il ne faut pas hésiter à sortir du cadre du groupe de résolution pour un traitement de plus grande ampleur.
- ***planification du zéro défaut.*** Un comité ad hoc « zéro défaut » doit être constitué dans l'équipe d'amélioration de la qualité. Ce comité mettra en place, un programme « zéro défaut » approprié aux besoins de l'entreprise et à sa culture.
- ***formation des responsables et des contremaîtres.*** Dès le démarrage du programme, à leurs différents niveaux de responsabilité, les responsables doivent être formés pour implanter ce qui leur revient dans le programme global d'amélioration de la qualité.
- ***la journée « zéro défaut ».*** Une journée « zéro défaut » doit être programmée afin que l'ensemble des personnels de l'entreprise soit sensibilisé aux nouveaux standards de performances.

- **définition des objectifs.** Afin de transformer les engagements en action, les individus doivent établir des objectifs d'amélioration pour eux-mêmes et pour leur groupe. Pour cela, chaque responsable définit avec les membres de son équipe des objectifs spécifiques à atteindre dont le résultat soit mesurable. Ces objectifs peuvent être portés à la connaissance de tous et des réunions régulières pourront en apprécier les progrès.
- **éliminer les causes d'erreurs.** Chaque employé se doit d'informer son responsable immédiat de toute anomalie ou problème lui interdisant de traiter une tâche sans erreur. Les employés n'ont pas à prendre des décisions particulières à cet effet. Par contre les problèmes signalés doivent recevoir une réponse du management dans les 24 heures.
- **reconnaissance.** Des incitations publiques et non financières doivent être données à ceux qui atteignent leur objectif de manière régulière.
- **comité qualité.** Les experts qualité et les personnes particulièrement motivées par la démarche de progrès doivent se rencontrer régulièrement afin d'échanger idées et expériences.
- **recommencer et toujours progresser.** Pour mettre l'accent sur l'amélioration permanente de la qualité, l'ensemble des points précédents doit être repris régulièrement. Ceci renouvelle l'engagement des anciens employés et introduit les nouveaux venus dans le processus, c'est le rôle du Comité Qualité

Le CMM

Le CMM est divisé en cinq niveaux de maturité.

Chaque niveau décrit un degré de maturité d'une organisation par rapport à son processus logiciel. C'est à dire :

- **Niveau initial** : le processus logiciel est caractérisé par la prédominance d'interventions ponctuelles, voire chaotiques. Très peu de processus sont définis et la réussite dépend de l'effort individuel.
- **Niveau reproductible** : une gestion de projet élémentaire est définie pour assurer le suivi des coûts, des délais et de la fonctionnalité du produit. La discipline nécessaire au processus est en place pour reproduire la réussite de projets d'un même domaine d'application.
- **Niveau défini** : le processus logiciel des activités de gestion et d'ingénierie est documenté, normalisé et intégré dans le processus logiciel standard de l'organisation. Tout nouveau projet de développement ou de maintenance logiciel fait intervenir une version adaptée et approuvée du processus logiciel standard de l'organisation.
- **Niveau maîtrisé** : des mesures détaillées sont prises en ce qui concerne le déroulement du processus logiciel et la qualité des produits du travail logiciel. Le processus logiciel et le niveau de qualité des produits sont connus et contrôlés quantitativement.
- **Niveau d'optimisation** : une amélioration continue du processus logiciel est mise en œuvre par une rétroaction quantitative émanant du processus lui-même et par l'application d'idées et de technologies novatrices.

La Figure A-1 illustre la vision dont bénéficie la hiérarchie (la direction d'une organisation) sur l'état et la performance d'un projet à chaque niveau de maturité. A chaque niveau successif correspond une amélioration de la vision de la hiérarchie sur les projets.

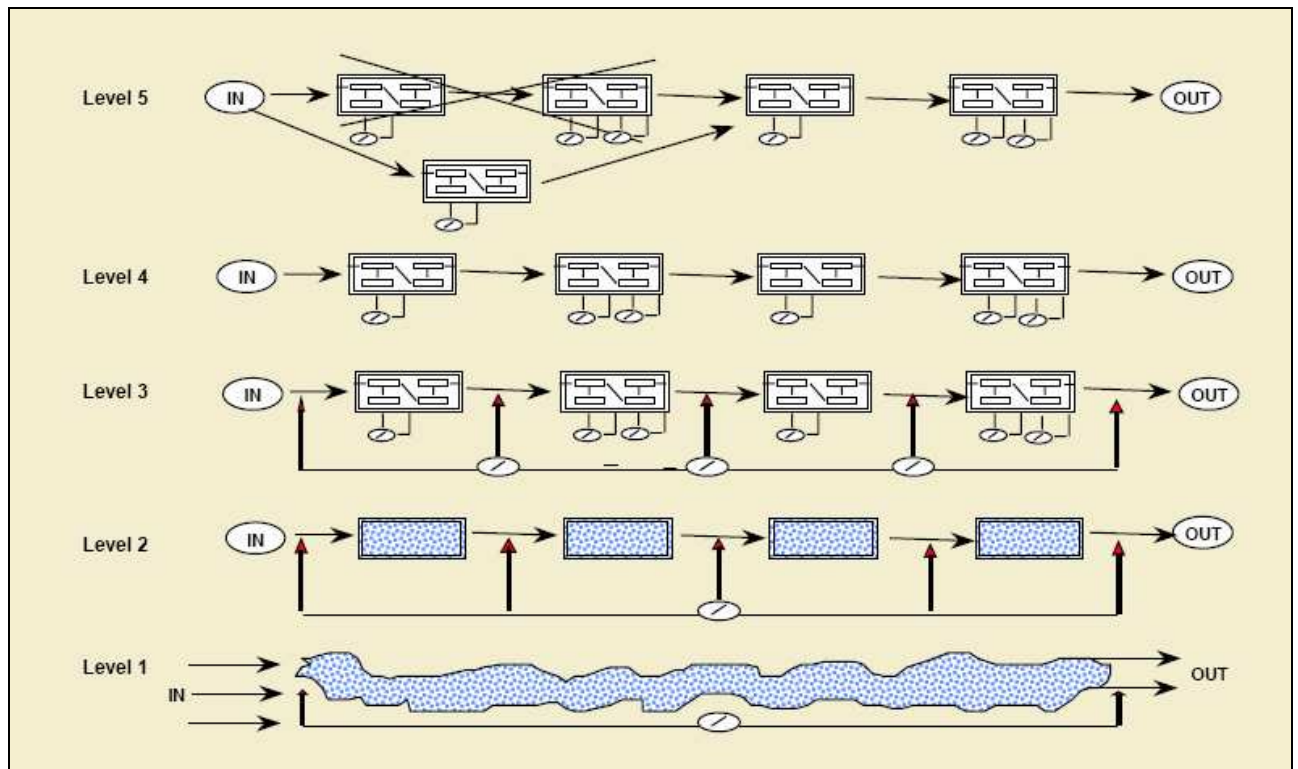


Figure A-1 : représentation schématique de l'évolution des processus d'une organisation avec l'augmentation de la maturité

Au niveau 1, le processus logiciel est une entité amorphe et la vue sur les processus du projet est limitée. Le découpage des activités étant à l'état d'ébauche, il est très difficile pour les gestionnaires d'évaluer les activités et l'état du projet. Les exigences sont prises en compte par le processus de façon anarchique et quelques produits en résultent.

Au niveau 2, les exigences du client et les produits du travail logiciel sont contrôlés et des pratiques élémentaires de gestion de projet ont été établies. Ces mécanismes de contrôle donnent, à des moments précis, une meilleure vue sur le projet. Le processus de réalisation de logiciels peut être illustré par une série de boîtes noires qui donne à la direction une vue sur les points de transition quand le déroulement de l'activité passe d'une boîte à l'autre. Même si la direction n'a pas en main tout le détail des opérations se déroulant à l'intérieur de chaque boîte, les produits du processus ainsi que les points de contrôle permettant de confirmer le bon déroulement du processus sont identifiés et connus. La direction réagit alors aux difficultés au fur et à mesure qu'elles se manifestent.

Au niveau 3, la structure interne de composants (c'est à dire les tâches constituant le processus logiciel défini du projet) est visible de l'extérieur. La structure interne illustre la façon dont les processus logiciels standard de l'organisation ont été appliqués dans le cadre de projets précis. La direction se prépare de façon proactive à gérer les risques potentiels. Les personnes extérieures au projet peuvent obtenir rapidement des renseignements précis sur l'état du projet puisque les processus définis assurent une grande visibilité des activités du projet.

Au niveau 4, les processus logiciel définis sont instrumentés et contrôlés de façon quantitative. Les gestionnaires peuvent mesurer l'avancement du projet et jauger les difficultés rencontrées. Ils disposent de fondements objectifs et quantitatifs leur permettant de prendre des décisions. Leur capacité de prévoir les résultats s'améliore régulièrement au fur et à mesure de la diminution de variabilité du processus logiciel.

Au niveau 5, de nouvelles méthodes améliorées de production de logiciels sont continuellement mises à l'essai dans un environnement contrôlé en vue d'améliorer la productivité et la qualité. Une discipline du changement est devenue chose courante et les activités inefficaces ou à l'origine de défauts sont repérées et ensuite remplacées ou révisées. La perspective dépasse le cadre des processus existants pour englober désormais l'impact des changements éventuels apportés au processus. Les gestionnaires sont en mesure d'évaluer l'impact et l'efficacité des changements et d'en effectuer le suivi quantitatif.

Comme nous l'avons vu, les niveaux de maturité sont organisés en secteurs clés, eux mêmes décomposés en pratiques clés. Passer d'un niveau n au niveau $n+1$ implique donc la mise en œuvre de toutes les bonnes pratiques de tous les secteurs clés du niveau n .

La Figure A-2 liste les secteurs clés identifiés dans le CMM par niveau de maturité.

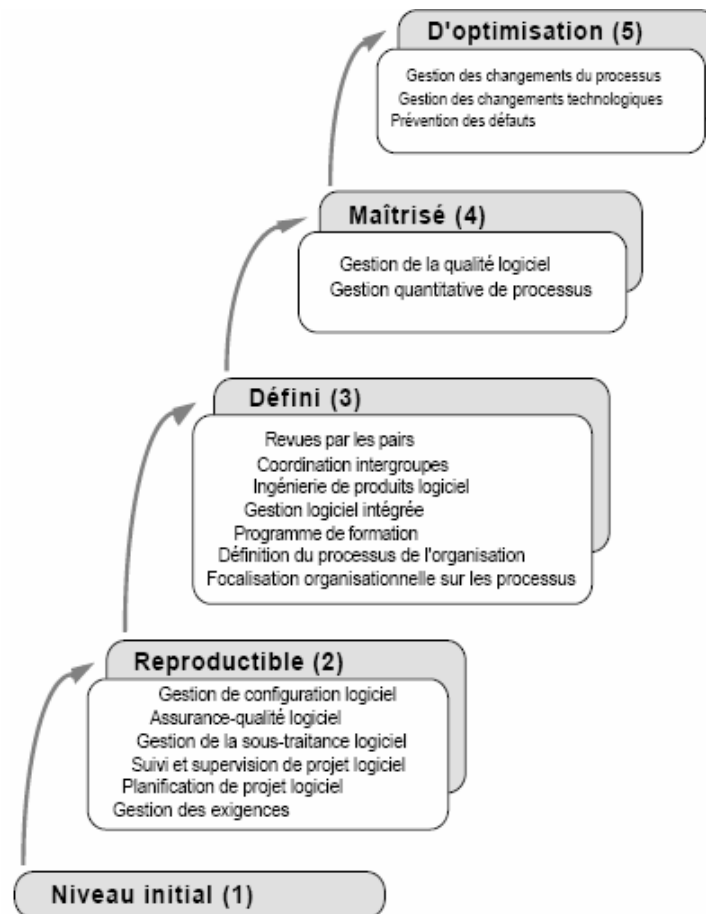


Figure A-2 : les secteurs clés couverts par chacun des 5 niveaux du CMM [CRAM, 1993]

La Figure A-2 illustre ce qui a été présenté précédemment. Les secteurs clés du niveau 2 portent sur la gestion de projet alors que ceux du niveau 3 sont plus axés sur l'ingénierie (réalisation proprement dite).

A titre d'exemple, nous proposons d'instancier la Figure A-2, pour une pratique du secteur clé « Planification de projet logiciel ».

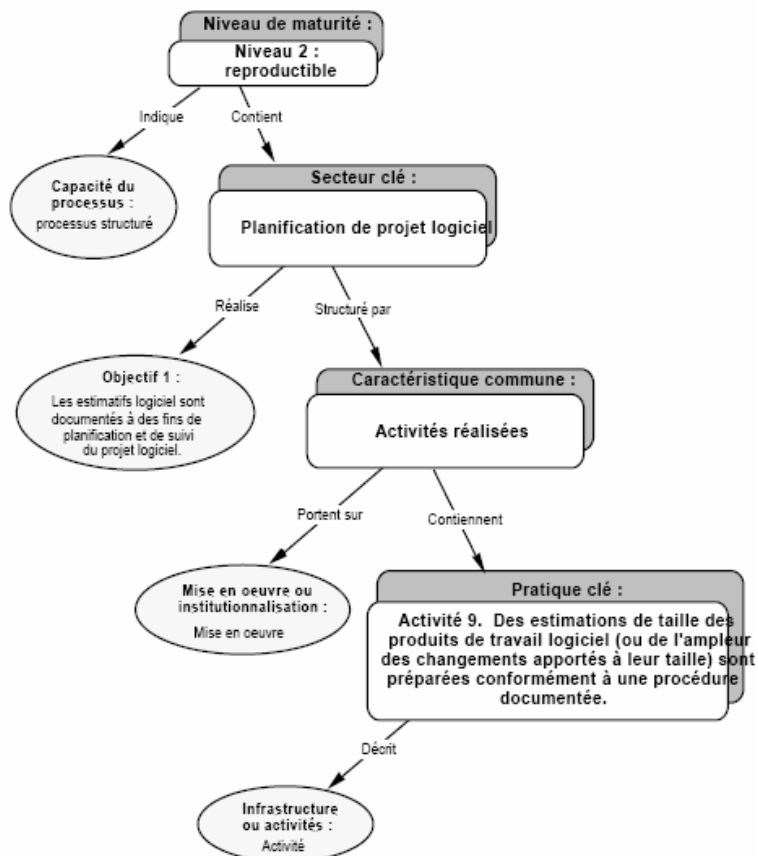


Figure A-3 : instanciation pour une pratique clé [CRAM,1993]

Les plans des deux livres de la norme ISO 20000

Livre 1	Livre 2
1. Périmètre	1. Périmètre
2. Termes et définitions	2. Termes et définitions
3. Exigences pour un système de management	3. Le système de management
3.1 Responsabilités	3.1 Responsabilités
3.2 Documentation des exigences	3.2 Documentation des exigences
3.3 Compétences, sensibilisation et formation	3.3 Compétences, sensibilisation et formation
4. Planification et implémentation de la gestion des services	4. Planification et implémentation de la gestion des services
4.1 Planifier le gestion de services (Plan)	4.1 Planifier le gestion de services (Plan)
4.2 Implémentation de la gestion de services et fourniture des services (Do)	4.2 Implémentation de la gestion de services et fourniture des services (Do)
4.3 Superviser, mesurer et revoir (Check)	4.3 Superviser, mesurer et revoir (Check)
4.4 Amélioration continue (Act)	4.4 Amélioration continue (Act)
4.4.1 Politique	4.4.1 Politique
4.4.2 Gestion des améliorations	4.4.2 Planifier les améliorations de services
4.4.3 Activités	
5 Planification et implémentation d'un nouveau service (ou modification d'un service existant)	5 Planification et implémentation d'un nouveau service (ou modification d'un service existant)
6 Processus de fourniture de services	6 Processus de fourniture de services
6.1 Accord sur le niveau de services (SLA)	6.1 Gestion du niveau de services
6.2 Reporting	6.2 Reporting
6.3 Continuité de service et gestion de le disponibilité	6.3 Continuité de service et gestion de le disponibilité
6.4 Budgetiser les services IT	6.4 Budgetiser les services IT
6.5 Gestion du capacitif	6.5 Gestion du capacitif
6.6 Gestion de la sécurité de l'information	6.6 Gestion de la sécurité de l'information
7. Processus de gestion des relations	7. Processus de gestion des relations
7.1 Généralités	7.1 Généralités
7.2 Gestion des relations métiers	7.2 Gestion des relations métiers
7.3 Gestion des fournisseurs	7.3 Gestion des fournisseurs
8. Processus de résolution	8. Processus de résolution
8.1 Background	8.1 Background
8.2 Gestion des incidents	8.2 Gestion des incidents
8.3 Gestion des problèmes	8.3 Gestion des problèmes
9. Processus de control	9. Processus de control
9.1 Gestion des configurations	9.1 Gestion des configurations
9.2 Gestion des changements	9.2 Gestion des changements
10. Processus de mise en production	10. Processus de mise en production
10.1 Gestion des mises en production	10.1 Gestion des mises en production

Tableau A-2 : structure des deux livres de la norme ISO 20000 (traduction personnelle) [ISO, 2005a] et [ISO, 2005b]¹

¹ Nous n'avons détaillé que les parties qui nous ont semblées pertinentes. Ainsi certaine sous-partie et sous-sous-parties du livre 2 n'apparaissent pas dans le tableau.

La validation pharmaceutique

Le projet de validation d'un système informatisé permet « d'établir la preuve documentée avec un haut degré d'assurance que le système mis en place produira régulièrement les résultats escomptés en termes de qualité et de spécifications » [ISPE, 2001a]. Pour ce faire, l'ISPE propose une démarche structurée permettant d'établir la preuve documentée que le système est validé.

Cette démarche comprend plusieurs étapes [Cabanne, 2001]:

- plan de validation : il permet d'organiser et de planifier les différentes phases de qualification du système informatisé.
- phases de qualification : ces phases permettent de tester le système selon des schémas approuvés : qualification de conception (QC), d'installation (QI), opérationnelle (QO), de performance (QP).
- rapports de qualification et de validation : ils permettent d'effectuer une synthèse des tests et résultats obtenus de manière à juger son adéquation avec les besoins de l'entreprise.

Les phases de QC, QI, QO et QP concernent tous les composants des systèmes informatisés et leurs contextes d'utilisation.

Qualification de Conception (QC)

La qualification de conception (également appelé « revue de conception avancée ») couvre les premières étapes du cycle de vie des applications informatiques [Cabanne, 2001] :

- expression des besoins,
- spécifications fonctionnelles,
- spécifications détaillées et spécifications techniques,
- réalisation.

Il s'agit de la « vérification documentée établissant que la conception proposée des installations, systèmes et équipements est adaptée à l'objectif fixé » [ISPE, 2001a].

Pratiquement, il peut s'agir par exemple :

- d'un audit du système qualité de la documentation de conception, de réalisation et de test du fournisseur, avec un résultat satisfaisant,
- de la vérification documentée de l'adéquation du produit aux exigences et besoins exprimés dans le cahier des charges,
- de la mise en place des plans de validations (PV) et des plans de tests (PT),

- de la revue formelle commune du cahier des charges (CDC) pour s'assurer de l'exhaustivité des besoins, exigences et contraintes exprimés par les utilisateurs et de leur bonne compréhension par le fournisseur.
- de la revue et signature pour acceptation des spécifications fonctionnelles.

Qualification d'Installation (QI)

La qualification d'installation consiste en la « vérification documentée établissant qu'un système est installé conformément à des spécifications écrites et préalablement approuvées » [ISPE, 2001a]. L'analyse de la définition précédente permet d'établir les différentes phase d'une QI :

- vérifier la présence et l'approbation effective (cohérence des dates, vérification des rôles...) des documents relatifs à l'installation (dont sécurité, robustesse ...) *théorique* du système à qualifier,
- vérifier que le système est effectivement installé conformément aux documents qui décrivent son installation prévisionnelle,
- vérifier que le système de management de la qualité permet de garantir que les évolutions futures du système seront contrôlées,
- documenter les activités précédentes de vérification.

Les procédures d'installation et les protocoles de qualification d'installation sont de la responsabilité du fournisseur, qui doit s'appuyer sur les spécifications techniques pour les élaborer. Ces documents doivent être approuvés avant d'être applicables.

Qualification Opérationnelle (QO)

Une qualification opérationnelle est une « vérification documentée établissant qu'un système fonctionne conformément à des spécifications écrites et préalablement approuvées, dans l'ensemble du domaine d'exploitation » [ISPE, 2001a]. Cela consiste donc à :

- vérifier la présence et l'approbation effective (cohérence des dates, vérification des rôles...) des documents relatifs aux fonctionnalités (dont sécurité, robustesse ...) attendus du système à qualifier,
- vérifier que le système fonctionne effectivement conformément aux documents qui décrivent ses fonctionnalités attendus,
- vérifier que le système de management de la qualité permet de garantir que les évolutions futures du système seront contrôlées,
- documenter les activités précédentes de vérification.

Qualification de Performance (QP)

Elle permet d'apporter la preuve que le système « tel que construit », dont l'installation et les fonctionnalités ont été qualifiées, fonctionne dans son environnement réel d'utilisation, en conformité avec les critères d'acceptation présents dans le CDC, et produit les résultats conformes escomptés.

Classification GAMP

L'ISPE propose de classer les systèmes informatisés en 5 catégories. Chaque catégorie va correspondre à une stratégie de validation différente, et donc un effort de validation différent. Elles sont fondées sur le risque croissant de défaillance de systèmes selon la progression allant des logiciels et matériels standards jusqu'aux logiciels et matériels sur mesure. Les 5 catégories sont décrites dans le tableau suivant (Tableau A-3).

Catégorie	Type de logiciel	Approche de la validation
1	Système d'exploitation	Gestion des versions, gestion des changements pour les mises à jour, analyse d'impact des nouvelles fonctionnalités
2	Microprogrammes (Firmware)	Gestion des versions, étalonnage, vérification du fonctionnement par rapport aux exigences utilisateurs, gestion des configurations, formation des utilisateurs, procédures d'utilisation
3	Logiciels standards	Vérification des spécifications (documentation, revue, tests), de la documentation fournisseur, audits fournisseurs, formation, procédures d'utilisation
4	Logiciels configurables	Vérification des spécifications (documentation, revue, tests), de la documentation fournisseur, audits fournisseurs, formation, procédures d'utilisation, gestion des configurations...
5	Logiciels personnalisés	Audits fournisseurs complets (systèmes de production, systèmes qualité...), Vérification des spécifications (documentation, revue, tests), de la documentation fournisseur, formation, procédures d'utilisation, gestion des configurations, audits des processus de l'environnement ...

Tableau A-3 : classification GAMP [ISPE, 2001a]

La réglementation pharmaceutique autorise donc de varier l'effort de validation en fonction du type du système considéré. La validation d'un système d'exploitation ne sera pas la même que celle d'un logiciel entièrement développé sur mesure.

Dans tous les cas, une analyse de risque doit être réalisée afin de [Féraud *et al.*, 2001] :

- *prévoir* les composants sur lesquels l'effort de validation devra se concentrer,
- *justifier* les choix de validation effectués (par exemple, absence de QP...)
- *prouver* que les risques sont maîtrisés, grâce à une démarche de validation adaptée.

Cycle en V et Validation

La processus de validation doit être considéré comme un processus « métier », et à ce titre, il doit être partie intégrante du cycle de vie du système. Ainsi, les phases de qualification et les systèmes de maîtrise nécessaires dans l'environnement réglementaire applicable à la pharmacie s'intègrent facilement dans le cycle en V traditionnel. La Figure A-4 illustre ceci.

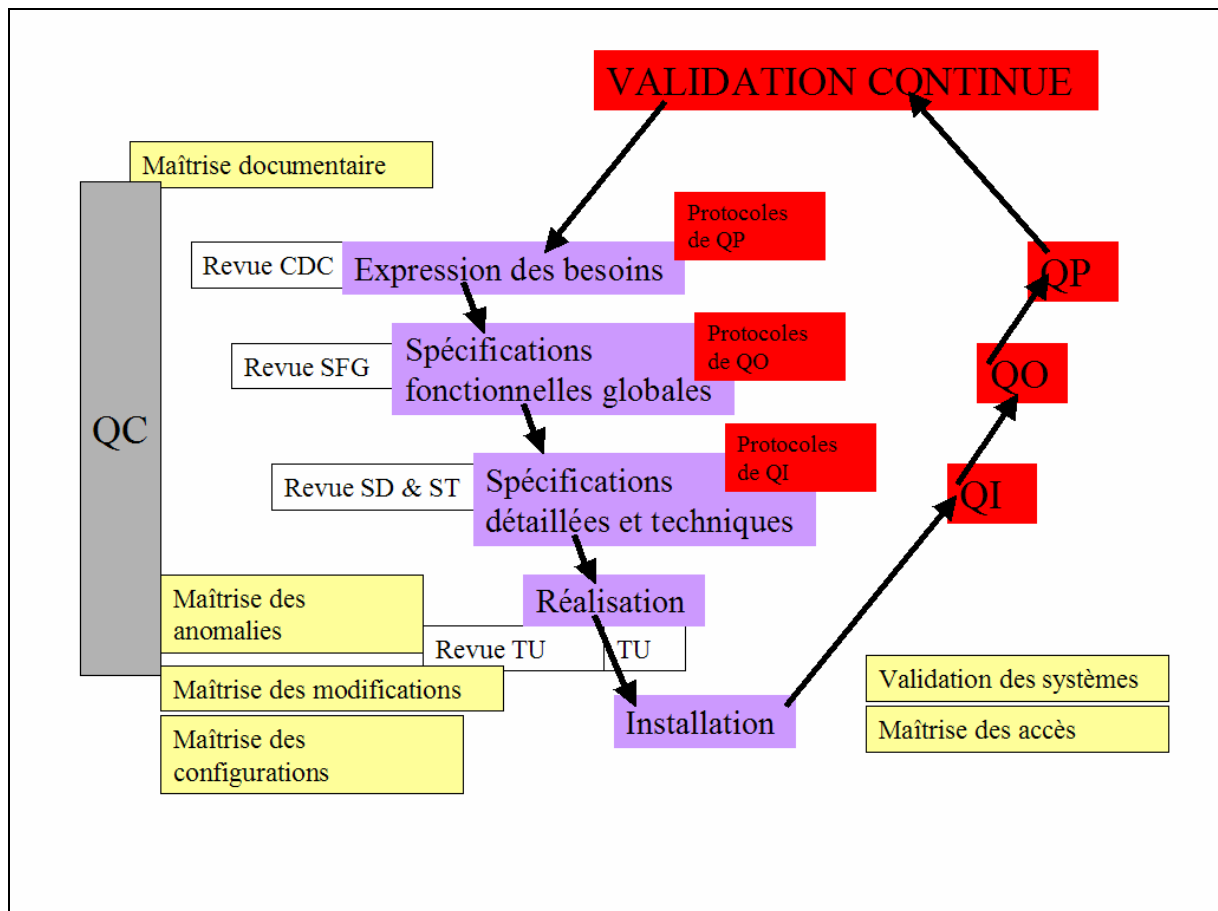


Figure A-4 : le cycle en V et la validation [Cabanne, 2001]

Diagrammes d'Ishikawa

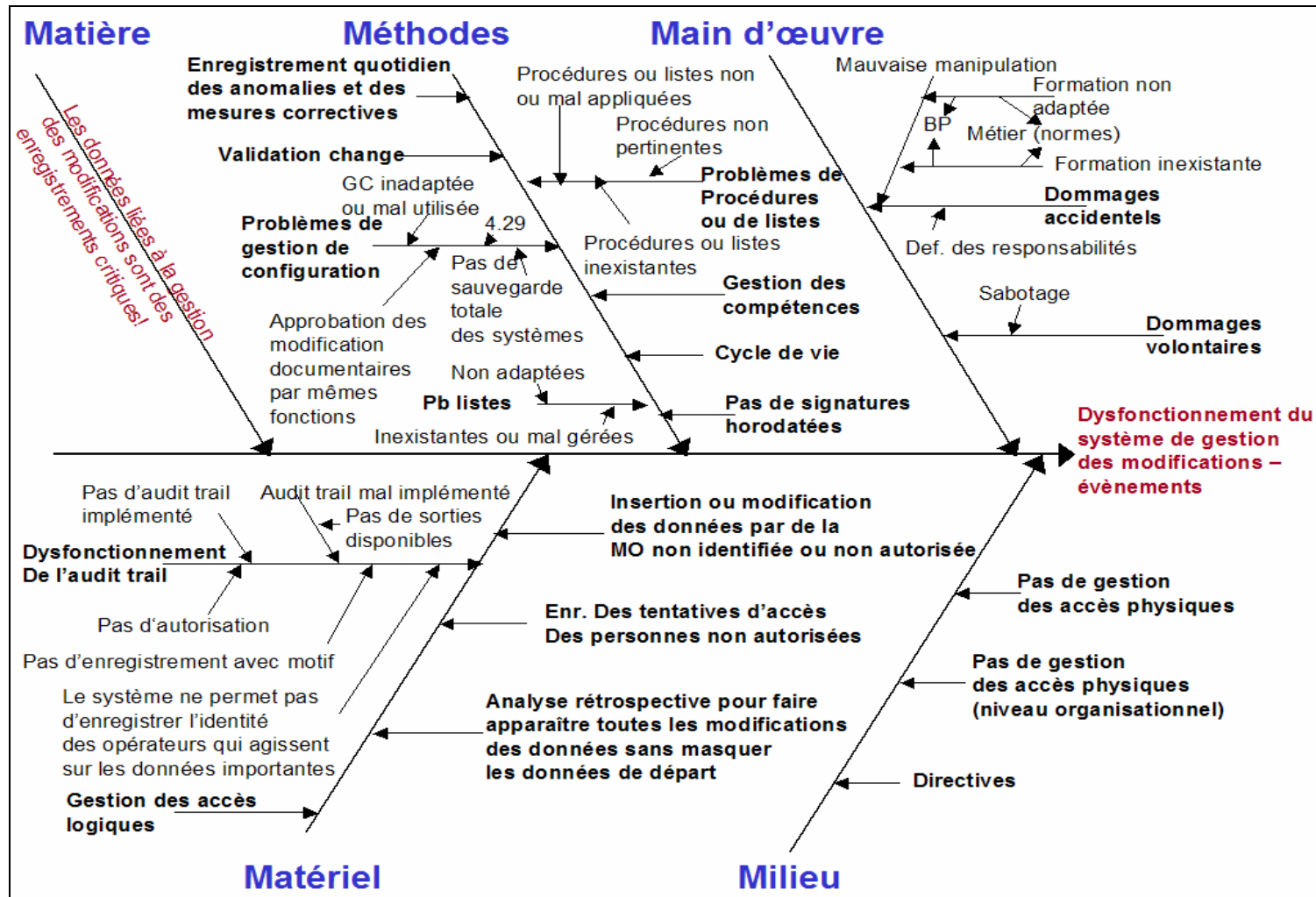


Figure A-5 : diagramme d'Ishikawa pour relatif au « groupe de risque » 2

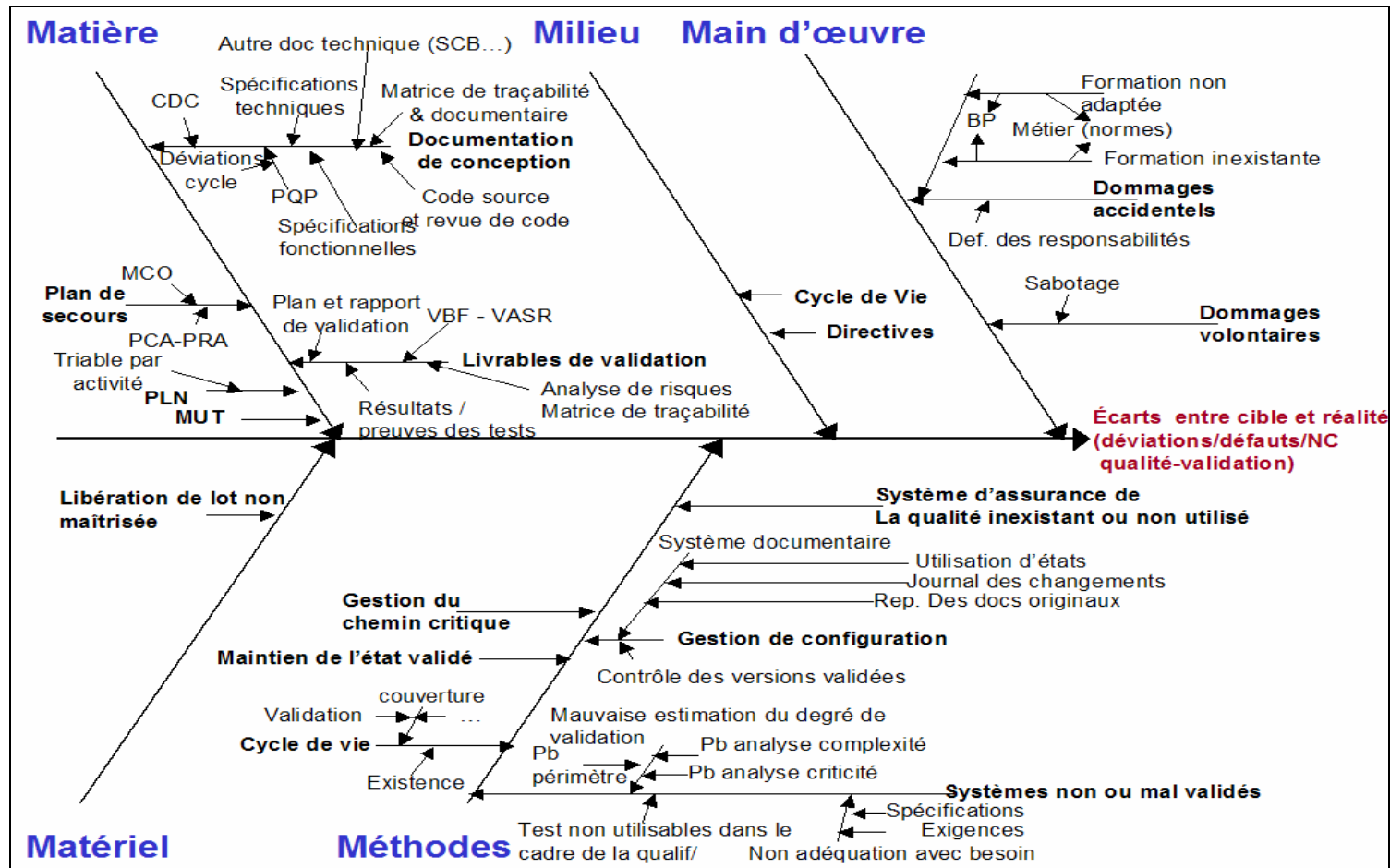


Figure A-6 : diagramme d'Ishikawa pour relatif au « groupe de risque » 3

Détail : Système d'assurance de la qualité

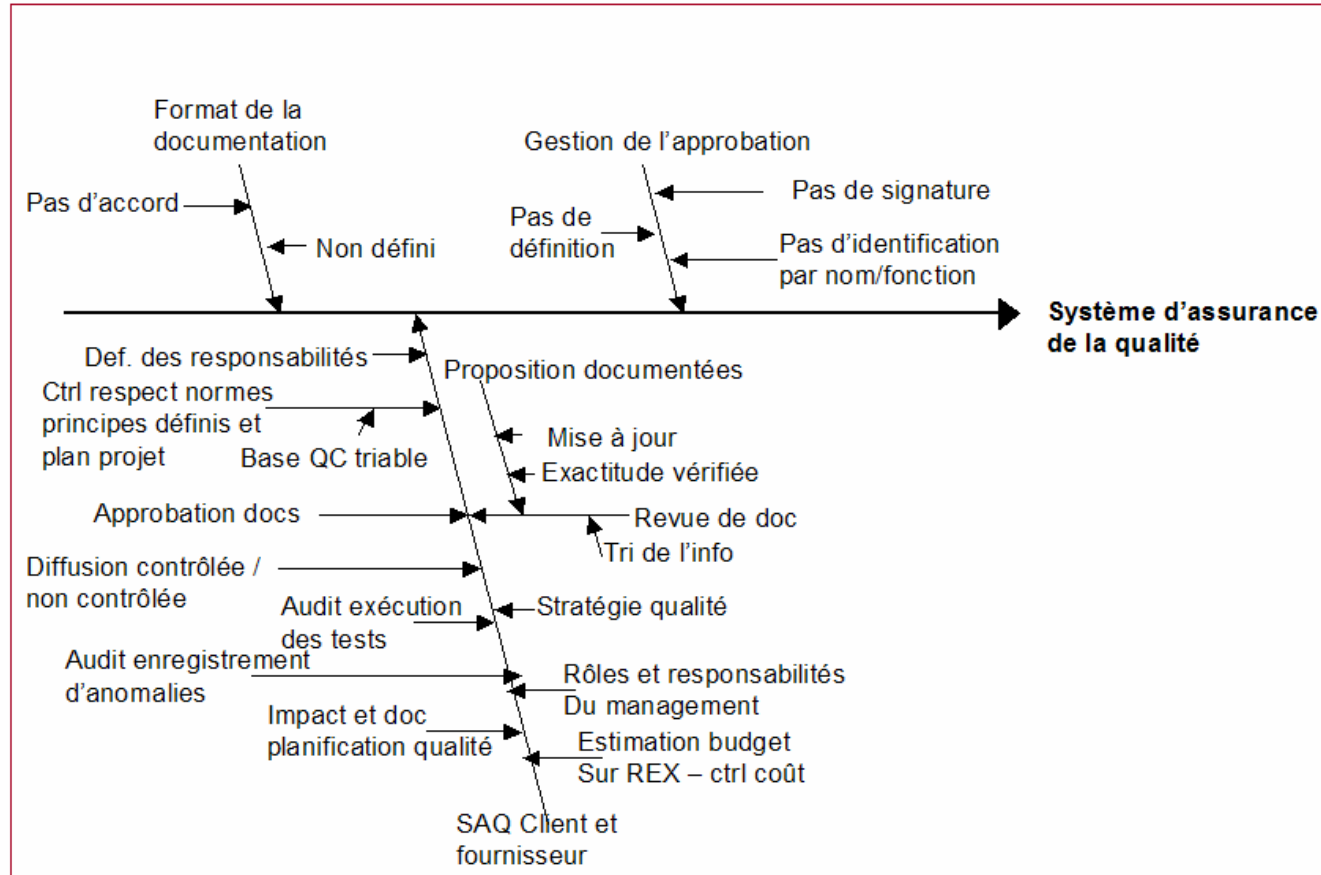


Figure A-7 : diagramme d'Ishikawa – zoom sur le système d'assurance de la qualité

Détail : Bonnes Pratiques de Test

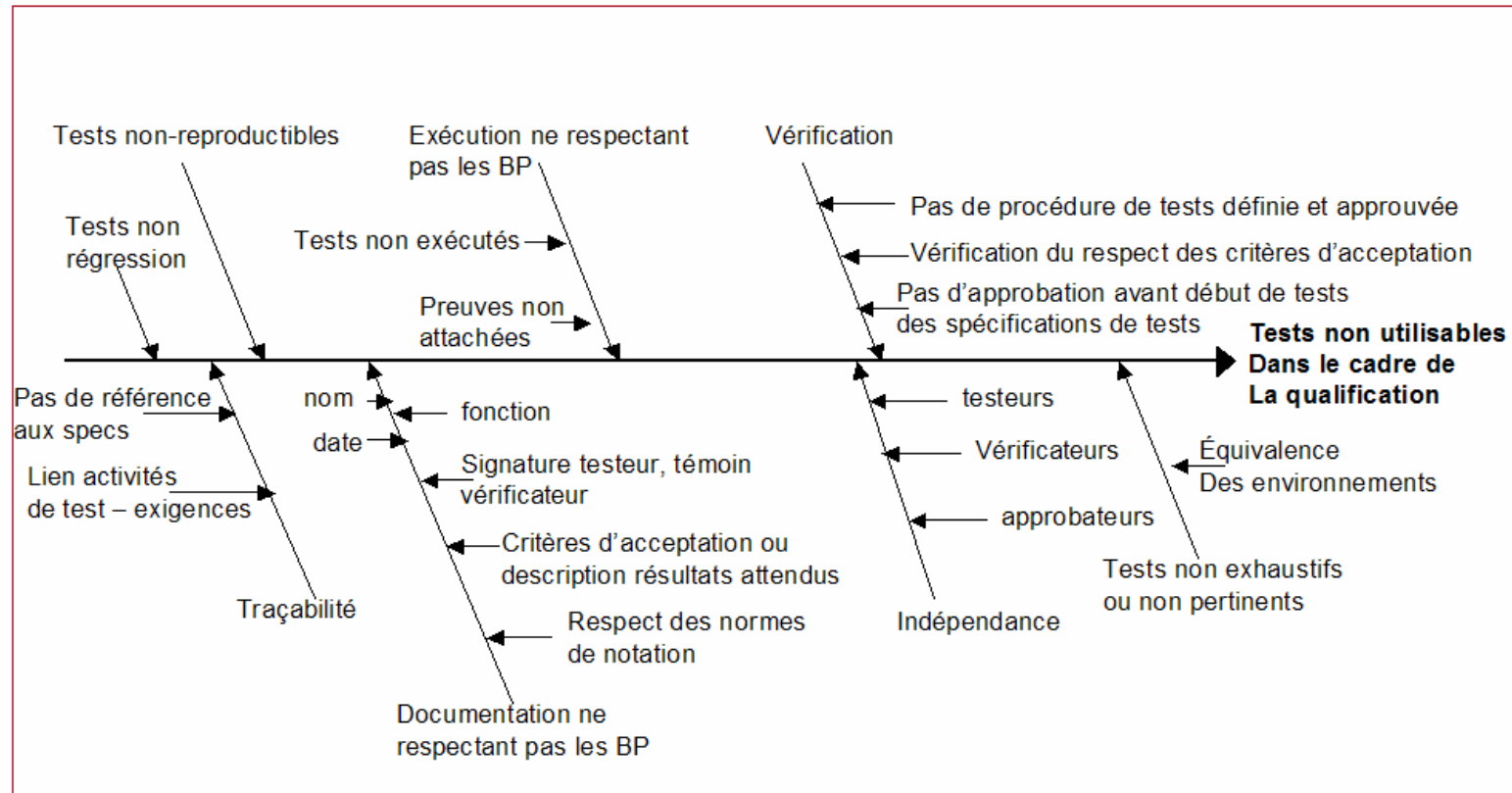
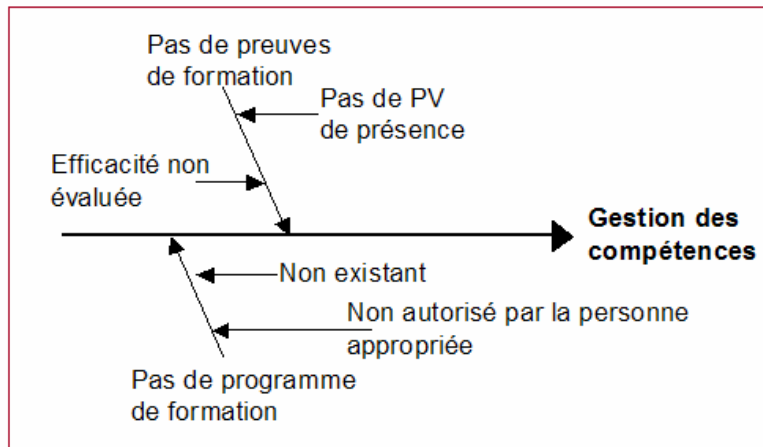


Figure A-8 : diagramme d'Ishikawa – zoom sur les bonnes pratiques de tests

Détail : Gestion des compétences



Détail : Branche pour chaque document

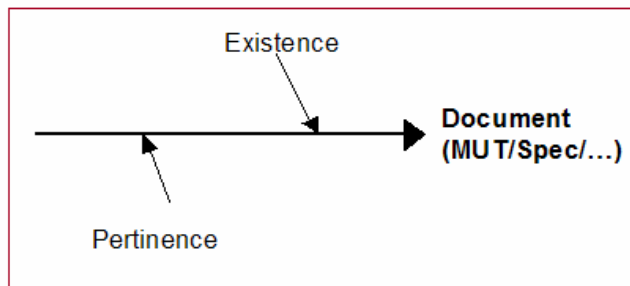


Figure A-9 : diagrammes d'Ishikawa – zoom sur Gestion des compétences et documents

Résumé :

Les systèmes d'information de l'industrie pharmaceutique sont maintenant tenus de respecter des exigences réglementaires imposées par le secteur d'activité. Ces exigences sont listées dans des documents, comme par exemple les GxP pour l'Europe, auxquels les autorités réglementaires se réfèrent.

De manière concomitante, l'entreprise peut avoir mis en œuvre des démarches d'amélioration, basées sur des référentiels de bonnes pratiques, comme le sont les modèles de maturité. Le Capability Maturity Model Integration (CMMI) est l'un d'entre eux et est souvent utilisé dans le domaine des systèmes d'information.

Ces deux catégories de documents, provenant d'expertise sectorielle pour l'un et d'expertise métier pour l'autre, se recouvrent. Dans un souci d'efficacité, leur déploiement ne peut s'imaginer sans une analyse de ce recouvrement, et nécessite la production d'une directive issue des deux sources documentaires pour l'assurance qualité. L'objectif de nos travaux est de mesurer le niveau de correspondance entre ces deux types de référentiels qualité.

Pour apporter une réponse à cette question, nous appliquons une démarche d'ingénierie dirigée par les modèles. Un mapping est réalisé entre des modèles de ces deux types de documents. Il a pour but de mettre en relation des connaissances comparables. Nous proposons ensuite une méthode de génération du référentiel cible basée sur des opérateurs logiques, et ce afin de retenir un ensemble minimal de pratiques à déployer assurant le respect des exigences imposées.

L'approche est déployée sur un cas d'application industrielle mobilisant 175 exigences des GxP et plus de 800 pratiques issues du CMMI, mais aussi d'autres sources de bonnes pratiques (ISO 9000, ITIL, ...).

Mots-clefs : *maturité, pratique, qualité, BPx, CMMI, industrie pharmaceutique*

Abstract :

Information systems for pharmaceutical industry are now obliged to respect regulatory requirements imposed by this branch of industry. These requirements are listed in documents, as for example the GxP for Europe.

In a concomitant way, organization can have deployed software process improvement methods, based on good practices repositories. The Capability Maturity Model Integration (CMMI) is one of them. It is often used in the area of information systems.

These two types of documents, coming from sectorial expertise for the first one and from business expertise for the second one, are overlapping. To be efficient, their deployment can not be done without analysing this overlapping and needs the realization of a policy stemming from the two documentary sources for quality assurance. The aim of our researches is to measure the degree of correspondences between these two quality repositories.

To bring an answer to this question, we apply a model driven engineering method. We realize a mapping between models of those two categories of documents. Its aim is to put into relation comparable knowledge. Then, we propose a method to build the target repository, based on logical operators use in order to identify a minimal set of good practices required by pharmaceutical regulation.

The approach is deployed on an industrial study case composed by 175 regulatory requirements and more than 800 practices, coming from CMMI but also from other repositories (ISO 9000, ITIL, ...).

Keywords : *maturity, practice, quality, GxP, CMMI, pharmaceutical industry*