



HAL
open science

CONTRIBUTION À L'INTEGRATION DES PROCESSUS METIER : APPLICATION A LA MISE EN PLACE D'UN REFERENTIEL QUALITE MULTI-VUES

Anis Ferchichi, Jean Pierre Bourey, Michel Bigand

► **To cite this version:**

Anis Ferchichi, Jean Pierre Bourey, Michel Bigand. CONTRIBUTION À L'INTEGRATION DES PROCESSUS METIER : APPLICATION A LA MISE EN PLACE D'UN REFERENTIEL QUALITE MULTI-VUES. Sciences de l'ingénieur [physics]. Ecole Centrale de Lille; Ecole Centrale Paris, 2008. Français. NNT: . tel-00295306

HAL Id: tel-00295306

<https://theses.hal.science/tel-00295306>

Submitted on 11 Jul 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

N° d'ordre : 74

ECOLE CENTRALE DE LILLE
ECOLE CENTRALE DE PARIS

THESE

présentée en vue
d'obtenir le grade de

DOCTEUR

Spécialité : Génie Industriel

par

Anis FERCHICHI

DOCTORAT DELIVRE CONJOINTEMENT
PAR L'ECOLE CENTRALE DE LILLE ET L'ECOLE CENTRALE DE PARIS

Titre de la thèse :

CONTRIBUTION À L'INTEGRATION DES PROCESSUS MÉTIER : APPLICATION À LA MISE EN PLACE D'UN RÉFÉRENTIEL QUALITÉ MULTI-VUES

Soutenue le 1er Juillet 2008 devant le jury d'examen :

Président	<i>David CHEN, Professeur à l'Université Bordeaux 1</i>
Rapporteur	<i>Pascal LHOSTE, Professeur des universités à l'Ecole Nationale Supérieure en Génie des Systèmes Industriels</i>
Rapporteur	<i>Didier GOURC, Maître-assistant HDR à l'Ecole Nationale Supérieure des Techniques, de l'Industrie et des Mines d'Albi-Carmaux</i>
Membre	<i>Hervé LFEBVRE, Directeur Produits à Syllis</i>
Directeur de thèse	<i>Jean-Pierre BOUREY, Professeur des Universités à l'Ecole Centrale de Lille</i>
Directeur de thèse	<i>Michel BIGAND, Maître de conférences HDR à l'Ecole Centrale de Lille</i>

Thèse préparée dans le Laboratoire de Génie Industriel de Lille – Ecole Centrale de Lille
et Recherche Opérationnelle Innovation - Syllis

Ecole Doctorale SPI 287 (EC Paris, EC Lille)

Remerciements

Je tiens à remercier très vivement les personnes qui ont su me motiver à entreprendre cette thèse et m'aider de leurs précieux conseils ; il m'importe en effet d'exprimer toute ma gratitude et ma reconnaissance à Mrs Jean-Pierre BOUREY et Michel BIGAND, mes directeurs de thèse à l'École Centrale de Lille. Leurs directives, leur ouverture d'esprit, leur soutien scientifique, leur disponibilité et leur enthousiasme pour mon projet de recherche m'ont énormément aidé pour réaliser ce travail.

*

Je tiens à remercier aussi chaleureusement Mr. Hervé LEFEBVRE, mon encadrant en entreprise et Business Manager de SYLIS. Ses observations et ses remarques m'ont amené souvent à remettre en question la pertinence de mes idées, lesquelles se sont souvent améliorées en intégrant ses suggestions. Je tiens à remercier également Mr. Marc BARRON, mon ex-encadrant en entreprise et ex-directeur de la filiale SDIE Widesourcing de SYLIS. Sa confiance et ses encouragements m'ont permis de débiter cette thèse.

*

Je remercie sincèrement mes rapporteurs : Mrs Pascal LAHOSTE et Didier GOURC qui ont eu l'infinie patience de relire mon mémoire de thèse. Je remercie également Mr. David CHEN d'avoir bien voulu participer au jury de soutenance de ma thèse.

*

Je tiens à remercier tous les professeurs du Laboratoire de Génie Industriel de Lille pour la qualité de leurs réponses et leurs remarques à chaque fois que je me suis adressé à eux. Je tiens à remercier également tous mes supérieurs et collègues à SYLIS et spécialement Mr.

Alain Baheux, directeur qualité de SYLIS, pour l'intérêt qu'ils ont porté à mon travail et pour leurs remarques pertinentes.

*

Je remercie également l'ensemble du personnel de l'École Centrale de Lille et de SYLIS qui m'a permis d'effectuer ma thèse dans les conditions les plus favorables.


*

Une thèse doctorale est une entreprise de longue haleine qui dure trois années, souvent passionnante mais parfois aussi source de moments un peu plus difficiles. C'est dans de tels instants que l'on apprécie d'autant plus le soutien de sa famille et de ses amis. Je voudrais leur dire ma gratitude à tous : mon père Dhifallah, ma mère Salha, mon frère Akram, ma sœur Khaoula, mon amie Louise, toute sa famille et tous mes amis en Tunisie et en France. Merci à tous.

*

Que tous ceux qui n'apparaissent pas dans ces quelques lignes et qui font partie de ma vie soient aussi remerciés : ils me poussent à poursuivre un chemin que j'apprécie davantage de jour en jour.

Table des matières

I	Préambule	1
1	Introduction générale	3
1.1	Cadre de la thèse	3
1.2	Organisation du document	5
		
II	Problématique et positionnement	9
2	L'organisation d'entreprise	11
2.1	Contexte scientifique	11
2.2	Contexte industriel	12
2.3	La gestion des processus	13
2.3.1	La collaboration difficile entre les acteurs définissant les processus . .	13
2.3.2	Un "Time to Market" trop important	15
2.4	Les limitations des solutions actuelles	16
2.4.1	Le Workflow	16

2.4.2	L'EAI - Enterprise Application Integration	17
2.4.3	Le B2Bi – Business to Business integration	17
2.4.4	Les progiciels intégrés	18
2.4.5	Les moteurs de règles métiers	18
2.4.6	Conclusion	18
2.5	La problématique et les objectifs	19
2.5.1	La problématique	19
2.5.2	Les objectifs opérationnels de l'entreprise	19
2.6	Conclusion	21



III Cadre méthodologique **23**

3 La modélisation d'entreprise **25**


3.1	Introduction	25
3.2	Système d'information : Approche systémique de l'entreprise	26
3.2.1	Définitions	26
3.2.2	Le modèle d'entreprise	27
3.2.3	Description du système d'information	27
3.3	Système d'information : Approche fonctionnelle de l'entreprise	29
3.4	Système d'information : Approche structurelle de l'entreprise	30
3.5	La modélisation d'entreprise	31
3.6	Les "framework" de modélisation d'entreprise	33
3.6.1	Qu'est ce qu'un framework de modélisation d'entreprise ?	33
3.6.2	Le Framework de Zachman - Les principes	34
3.6.3	Vue d'ensemble des principaux framework	34


3.6.3.1	Federal Enterprise Architecture Framework (FEAF)	34
3.6.3.2	DoD Architecture Framework (DODAF)	37
3.6.3.3	Autres Frameworks	37
3.7	Conclusion	38
4	Les principales méthodes de modélisation d'entreprise	39
4.1	Introduction	39
4.2	Les processus métier	40
4.2.1	Qu'est ce qu'un processus métier ?	40
4.2.2	Pourquoi parler des processus métier ?	41
4.2.3	Une ontologie d'entreprise	42
4.3	Une classification des langages de modélisation des processus	44
4.3.1	Les types de langages de modélisation	44
4.3.2	Une vue d'ensemble des langages de modélisation des processus métier	45
4.4	Les langages traditionnels de modélisation des processus	46
4.4.1	IDEF - ICAM DEFinition Language	46
4.4.1.1	IDEF0	47
4.4.1.2	IDEF1	48
4.4.1.3	IDEF2	48
4.4.1.4	IDEF3	48
4.4.2	Les réseaux de Petri	49
4.4.3	La méthodologie GRAI	49
4.4.4	Les autres langages traditionnels de modélisation des processus	51
4.5	Les langages d'intégration de processus	52
4.5.1	Langage de modélisation de Workflow	52
4.5.2	RosettaNet	53

4.5.3	BPEL4WS	53
4.6	Les langages orientés objets	53
4.6.1	Unified Modeling Language (UML)	54
4.7	Conclusion	54
5	Le Business Process Management	57
5.1	Introduction	57
5.2	Le concept de Business Process Management (BPM)	57
5.3	Les standards et BPM	58
5.3.1	BPML - Business Process Modeling Language	59
5.3.2	BPMN - Business Process Modeling Notation	59
5.3.3	BPEL - Business Process Execution Language	60
5.3.4	BPQL - Business Process Query Language	61
5.3.5	BAML - Business Activity Monitoring Language	61
5.3.6	BPATS - Business Process Audit Trail Schema	61
5.3.7	Les autres standards de BPM	61
5.4	Les apports du BPM	62
5.5	Le BPMS : Business Process Management System	63
5.5.1	Les principales fonctionnalités d'un BPMS	63
5.5.1.1	Définition des processus	63
5.5.1.2	Exécution des processus	64
5.5.1.3	Surveillance des processus	65
5.5.1.4	Analyse et création de rapport sur les processus	65
5.5.1.5	Autres composants	65
5.5.2	Les principaux BPMS du marché	66
5.5.2.1	ARIS Platform	66

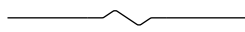
5.5.2.2	Casewise Corporate Modeler	67
5.5.2.3	MEGA Process	68
5.5.2.4	ADONIS	69
5.5.2.5	Microsoft Biztalk server	69
5.5.2.6	Intalio	69
5.5.2.7	GraiTools	70
5.6	Conclusion	70
6	La notion de qualité en entreprise	71
6.1	Introduction	71
6.2	la Qualité	71
6.3	La Qualité Produit	73
6.3.1	La norme ISO 9126, Technologies de l'Information : "Qualité des produits logiciels"	73
6.4	Le management de la qualité par les processus	75
6.4.1	Qu'est ce qu'un processus?	75
6.4.2	L'amélioration des processus	75
6.4.3	Pourquoi une approche par les processus?	78
6.4.4	Six Sigma	79
6.4.4.1	Présentation de Six Sigma	79
6.4.4.2	Les apports de Six Sigma	80
6.4.4.3	Six Sigma, la culture de la mesure au service de l'amélioration continue	81
6.4.4.4	DMAIC : Define, Measure, Analyse, Improve, Control	82
6.4.4.5	DMADV : Define, Measure, Analyze, Design, Verify	82
6.4.5	COBIT, Control Objectives for Business & Related Technology	84
6.4.5.1	Présentation de COBIT	84

6.4.5.2	Les 5 parties de la méthode COBIT	84
6.4.6	ITIL, Information Technology Infrastructure Library	87
6.4.6.1	Principe d'ITIL	87
6.4.6.2	Description d'ITIL	87
6.4.7	PMBok, Project Management Body of Knowledge	88
6.4.8	SWEBoK, SoftWare Engineering Body of Knowledge	92
6.4.9	ISO 9001v2000 "Systèmes de management de la qualité - Exigences" .	94
6.4.9.1	Présentation ISO 9001v2000	94
6.4.9.2	Contenu d'ISO 9001v2000	97
6.4.9.3	PDCA : Plan, Do, Check, Act, "la roue de Deming"	98
6.4.10	CMMI, Capability Maturity Model Integration	100
6.4.10.1	Présentation de CMMI	100
6.4.10.2	CMMI représentation étagée	101
6.4.10.3	CMMI représentation continue	104
6.4.10.4	Apports de CMMI	106
6.4.10.5	IDEAL SM : Initiating, Diagnosing, Establishing, Acting, Learning	107
6.4.10.6	SCAMPI : "Standard CMMI Appraisal Method for Process Improvement"	109
6.5	Conclusion	111
7	L'interopérabilité	113
7.1	Introduction	113
7.2	Les principaux travaux en interopérabilité	114
7.2.1	Le groupe de travail TG2 d'INTEROP	114
7.2.2	Le projet ATHENA & UEML	118
7.2.3	OASIS	119

7.2.4	OWL et l'interopérabilité	119
7.3	Conclusion	120
8	Conclusion de la partie	121
		
IV	Apport scientifique	125
9	Référentiel qualité multi-vues	127
9.1	Introduction	127
9.2	Le modèle proposé	129
9.2.1	La description du modèle	129
9.2.2	Les étapes de mise en place d'un référentiel intégré	131
9.2.2.1	L'étape 1 : Choix des modèles	131
9.2.2.2	L'étape 2 : Analyse d'écart et rapprochements	131
9.2.2.3	L'étape 3 : Construction d'un modèle intégré	132
9.2.2.4	L'étape 4 : Adapter ce modèle au fonctionnement de l'entreprise : le rendre spécifique	133
9.2.3	Cas d'application ISO 9001v2000 - CMMI	134
9.2.3.1	Les synergies entre ISO 9001v2000 et CMMI	134
9.2.3.2	La possibilité de combiner ISO 9001v2000 et CMMI	137
9.2.3.3	Traduction des terminologies	138
9.2.3.4	Implémentation de l'ontologie pour la réalisation du mapping ISO 9001v2000 - CMMI	139
9.2.3.5	Le mapping ISO 9001v2000 - CMMI	139
9.2.3.6	La validation du mapping ISO 9001v2000 - CMMI	140
9.2.4	L'interopérabilité des processus	140

9.2.4.1	La solution d'interopérabilité orientée clients	141
9.2.4.2	La solution d'interopérabilité orientée partenaires	145
9.3	Conclusion	149
10	Focalisation sur un processus critique : estimation des délais et efforts des projets	153
10.1	Introduction	153
10.2	Modèle proposé	154
10.3	Présentation de la ligne de code	156
10.4	Présentation des Points de Fonction (FP)	156
10.4.1	Des Points de Fonction vers les Lignes de Code	157
10.5	Recommandations	158
10.6	Choix du modèle d'estimation	159
10.7	Calcul de l'effort et du temps de développement	159
10.8	La répartition par phase	160
10.9	Le calibrage du modèle	160
10.10	Conclusion	162
		
V	Cas d'étude	165
11	Application à Sylis	167
11.1	Introduction	167
11.2	Présentation de Sylis	167
11.2.1	Les métiers de Sylis	168
11.2.2	Les axes d'intervention de Sylis	171
11.3	Description de la démarche globale	172

11.4	Référentiel qualité	173
11.4.1	Le mapping opérationnel ISO 9001v2000 - CMMI	176
11.4.2	Une cartographie unique des processus	176
11.4.3	CMMI et les processus implémentés	180
11.4.4	Un exemple de processus mis en place : PROCES-REAL-001 Processus Vendre avec engagement de moyens	181
11.5	Plan d'amélioration du référentiel Qualité	186
11.5.1	Examen du système qualité	187
11.5.1.1	Point sur les décisions prises	187
11.5.1.2	Point de fonctionnement du Système de Management de la Qualité ISO 9001v2000 - CMMI	187
11.5.2	Synthèse d'appréciation	188
11.6	Le référentiel documentaire	188
11.7	Audit projet	192
11.8	Le référentiel des compétences	192
11.9	L'aspect humain	197
11.10	Conclusion	198
VI	Épilogue	201
12	Conclusion générale et Perspectives	203
12.1	Rappel du cadre des travaux	203
12.2	Synthèse sur les apports de ces travaux	204
12.3	Limitations et perspectives	206



Annexes	209
A Explication des Process Areas de CMMI	211
B SCAMPI (Standard CMMI Appraisal Method for Process Improvement)	
class A	215
B.1 Planification et préparation de l'évaluation	215
B.2 Conduite de l'évaluation	219
B.3 Le rapport des résultats	219
B.4 Qualification des Appraisal Team Members (ATM)	220
C Le résultat du mapping ISO 9001v2000 - CMMI	223
C.1 Mapping ISO 9001v2000 vers CMMI	223
C.2 Mapping CMMI vers ISO 9001v2000	231
D Les techniques d'estimation des coûts et des charges	261
D.1 Introduction	261
D.2 La mesure Homme-Mois (h.m)	263
D.3 Techniques d'estimation des coûts	265
D.3.1 Jugement des experts	265
D.3.2 Estimation par analogie	266
D.3.3 Estimation ascendante	267
D.3.4 Estimation descendante	267
D.4 Les modèles d'estimation des coûts	268
D.5 Les modèles paramétriques d'estimation des coûts	270
D.5.1 Les modèles linéaires	270
D.5.2 Les modèles analytiques	274
D.5.3 Les limitations des modèles paramétriques	277

D.6	Les modèles non-paramétriques d'estimation des coûts	279
D.6.1	Les modèles d'estimation utilisant les réseaux de neurones	279
D.6.2	Les modèles d'estimation utilisant les arbres de régression	281
D.6.3	Les modèles d'estimation utilisant les algorithmes génétiques	284
D.7	La validation des modèles d'estimation des coûts	287
D.8	Conclusion	290
E	Les paramètres de la méthode d'estimation proposée	293
E.1	Les composants des Points de Fonction (FP)	293
E.2	Les facteurs d'échelle	298
E.3	Les facteurs de coûts du modèle post-architecture	299
E.3.1	Les attributs du produit	301
E.3.2	Les attributs de la plate-forme	304
E.3.3	Les attributs du personnel	305
E.3.4	Les attributs du projet	306
E.4	Les facteurs de coûts du modèle précoce	307
	Liste des figures	313
	Liste des tableaux	319
	Bibliographie	323

PREMIÈRE PARTIE

PRÉAMBULE

Introduction générale

DANS ce chapitre introductif, nous faisons le point sur le *cadre* de la thèse. Nous proposons également un rapide *survol* des éléments présentés dans ce manuscrit et de la façon dont ils s'enchaînent.

1.1 Cadre de la thèse

Contexte

Afin de préserver son avantage concurrentiel dans une conjoncture où les prix de vente des prestations sont extrêmement serrés, la société Sylis, société de services en ingénierie informatique, souhaite adopter une démarche volontariste en phase avec la réalité de la mondialisation de l'économie. Deux leviers majeurs de sa compétitivité peuvent être actionnés :

- d'une part, la diminution des coûts salariaux, par l'externalisation à l'international d'une partie des travaux de maintenance - développement ;
- d'autre part, l'augmentation de la valeur ajoutée des prestations par la mise en œuvre ou l'amélioration d'approches méthodologiques rigoureuses.

Pour réduire ses coûts, Sylis a déjà collaboré avec une société roumaine, et a tissé des liens avec la Tunisie, l'Inde et le Maroc. Ce premier aspect est lié au second : le travail externalisé demande un pilotage fiable, seul garant de son intérêt économique. Sylis peut d'ores et déjà s'appuyer sur son savoir-faire en matière de réalisations de prestations avec engagement de résultats ou de moyens, mais la complexité de cette nouvelle problématique

d'internationalisation requiert de nouvelles réponses. C'est sur ce point que le Laboratoire de Génie Industriel de Lille (LGIL) a apporté son concours.

Thème de recherche

Cette thèse s'inscrit dans la suite logique du stage de Master Recherche mention Génie Industriel, spécialité Systèmes d'Information et Ingénierie de la Conception réalisé entre mai et septembre 2005. Le sujet de master s'intitulait « Ingénierie de production de logiciels : formalisation d'une démarche d'évaluation des développements et définition d'un environnement de développement collaboratif » ; il s'intéresse donc spécifiquement à l'un des processus vitaux de l'entreprise, l'évaluation des développements en termes de charge de travail (donc de coût) et de délai, dont dépend en grande partie l'équilibre économique de la prestation. Dans ce cadre, une méthode de convergence des estimations à partir de différents modèles de coûts a été proposée ; elle a été déployée sur toutes les agences de Sylis en France.

Le sujet de thèse porte sur la modélisation d'entreprise pour l'amélioration des performances dans le cadre d'activités de conception et réalisation d'applications informatiques sur le mode collaboratif. Il couvre un champ beaucoup plus large que celui abordé dans le cadre du stage de master. Cette problématique prend en compte à la fois des aspects liés aux technologies, des aspects organisationnels et des aspects socio-économiques ; une approche génie industriel s'avère donc pertinente.

Il s'agit de mettre en place une ingénierie de projets logiciels, c'est-à-dire des méthodes, démarches et outils évaluables, reproductibles et contrôlables, conduisant à une « industrialisation » des processus.

La méthodologie proposée est générique et peut être généralisée à n'importe quelle entreprise souhaitant organiser sa façon de travailler à travers ses processus. Dans notre travail, Sylis constitue notre cas d'application.

Les processus concernent la conception – réalisation de projet neuf ainsi que la tierce maintenance applicative. Ils doivent intégrer le système qualité ISO 9001 version 2000 déjà en place dans la société Sylis ainsi que le référentiel CMMI (Capability Maturity Model Integration) propre aux développements informatiques. Plus généralement, la démarche proposée est générique et prévoit l'intégration future d'autres référentiels qualité (ITIL...).

La méthodologie permet le pilotage en temps réel de l'ensemble des développements, et prend en compte le système d'échanges à mettre en place à tous les niveaux (en interne avec la direction et les collaborateurs, en externe avec les partenaires internationaux et les clients). Les différences culturelles avec les pays étrangers et l'évolution des missions des collaborateurs français ont fait l'objet d'une attention particulière.

1.2 Organisation du document

Ce mémoire de thèse est organisé comme suit :

Partie I : Introduction Générale présente le cadre général de la thèse, le thème de recherche et l'organisation du manuscrit de thèse.

Partie II : Problématique et positionnement présente la problématique de nos travaux et notre positionnement. Nous exposons, dans un premier temps, la situation difficile actuelle pour organiser et gérer les processus d'entreprise. Dans un deuxième temps, nous énonçons notre problématique d'organisation des processus métier en intégrant différents standards qualité afin de mettre en place un référentiel qualité multi-vues. Enfin, ce chapitre se termine par la présentation des objectifs opérationnels assurant l'intégration du référentiel qualité au sein d'une entreprise.

Partie III : Cadre méthodologique présente une exploration de l'espace de nos travaux de thèse. Elle comporte cinq chapitres. Le premier chapitre traite la modélisation d'entreprise. Nous présentons les différentes approches de l'entreprise et nous concluons par les principaux "frameworks" de modélisation d'entreprise. Le deuxième chapitre présente un panorama des langages et standards de modélisation d'entreprise. Le troisième chapitre s'intéresse au Business Process Management (BPM) véritable standard d'organisation d'entreprise et de gestion des processus métier en exposant les principes et les différents standards et outils relatifs au BPM. Le quatrième chapitre présente la notion de qualité en entreprise, coeur de nos travaux, et expose les principaux standards et normes qualité. Enfin le cinquième chapitre, présente un panorama des principaux travaux, solutions et standards d'interopérabilité.

Partie IV : Apport scientifique présente notre proposition d'une démarche générique permettant d'implémenter un référentiel qualité multi-vues. Ensuite, nous appliquons notre démarche afin d'intégrer les standards qualité ISO 9001v2000 et Capability Maturity Model Integration (CMMI) dans un référentiel qualité unique et d'assurer l'interopérabilité des processus définis avec ceux des clients et/ou partenaires. Dans le deuxième chapitre de cette partie, nous nous focalisons sur un processus critique, l'estimation des délais et des charges des projets, et nous présentons la méthodologie définie et mise en place.

Partie V : Cas d'étude et implémentation à Sylis présente l'application de la démarche générale permettant d'intégrer deux standards qualité afin de créer un référentiel qualité multi-vues sur un cas concret au sein de Sylis. La mise en place du référentiel qualité est accompagné de la mise en place d'un référentiel documentaire, une méthodologie

d'audit projet et d'un référentiel de compétences garants de la bonne implémentation du référentiel qualité au sein d'une entreprise.

Partie VI : Conclusion et perspectives clot cette thèse en donnant les conclusions et les perspectives de ce travail. Nous proposons un bilan du travail effectué durant cette thèse et un ensemble de perspectives.

DEUXIÈME PARTIE

PROBLÉMATIQUE ET POSITIONNEMENT

L'organisation d'entreprise

DANS ce deuxième chapitre, nous présentons un état des lieux de la situation actuelle des projets informatiques et des solutions mises en place pour remédier aux problèmes récurrents de ces mêmes projets. Nous proposons également notre *problématique de recherche* et les *objectifs* que nous nous sommes fixés.

2.1 Contexte scientifique

Les problématiques d'organisation d'entreprise, de modélisation de processus et d'implémentation des bonnes pratiques sont traitées par beaucoup de communauté de recherche :

- À l'échelle nationale, on peut citer l'**AFIS (Association Française en Ingénierie Système)** et surtout ses groupes de travail : Modèle de Maturité et Indicateurs (MMI), Management de Projet, Méthodes et Outils ;
- À l'échelle internationale, on peut citer l'**INCOSE (International Council On Systems Engineering)**.

Ces deux communautés travaillent sur la promotion de la pratique de l'Ingénierie Système auprès des entreprises pour la modélisation et l'implémentation des bonnes pratiques.

- On peut citer également le **SEI (Software Engineering Institute)** de l'université Carnegie Mellon. Cette communauté travaille sur l'amélioration de l'ingénierie informatique en définissant les bonnes pratiques et des standards qualité.

Parmi les travaux de recherche, en cours ou déjà finis, on peut citer :

- La thèse de Messieurs **Romain DEGUIL, Hervé PINGAUD et Didier GOURC,**

de l'Ecole des Mines d'Albi-Carmaux, intitulée "**Détermination d'un ensemble minimal de bonnes pratiques informatiques nécessaires pour couvrir un référentiel réglementaire de l'industrie pharmaceutique**". Cette thèse traite la détermination d'un ensemble de bonnes pratiques tirés d'un standards qualité CMMI (Capability Maturity Model Integration) pour couvrir un référentiel réglementaire de l'industrie pharmaceutique. La définition de cet ensemble passe par la réalisation d'un mapping entre le standard des bonnes pratiques informatiques et le référentiel réglementaire. C'est la démarche que nous adopterons dans notre travail à savoir un mapping entre deux standards qualité. En plus de ça, nous nous intéressons au même standard qualité : CMMI ;

- La thèse de Messieurs **Jean-Christophe BLAISE, Pascal LHOSTE et Joseph CICCOTELLI**, de l'Université Henri Poincaré Nancy I, intitulée "**Apport d'une modélisation de l'information normative à l'intégration des règles de sécurité des machines en conception**". Cette thèse traite la modélisation d'entreprise en intégrant des règles de sécurité. La démarche d'intégration proposée converge avec nos travaux et notre démarche d'intégration de deux standards qualité ;
- Les travaux de recherche de **Boris MUTAFELIJA et Harvey STROMBERG**, de l'Université de Boston, intitulé "**Exploring CMMI-ISO 9001v2000 synergy when developing a process improvement strategy**". Ce travail étudie la possibilité d'intégrer CMMI et ISO 9001v2000 afin de réaliser une stratégie d'amélioration des processus. L'intégration proposée passe par la réalisation d'un mapping entre ces deux standards qualité. Notre démarche s'appuie également sur un mapping CMMI et ISO 9001v2000 afin de mettre en place un référentiel qualité. Ainsi ce travail de Boris MUTAFELIJA et Harvey STROMBERG nous a permis de valider nos résultats en confrontant les résultats des deux mappings.

2.2 Contexte industriel

Un rapport de 1994 du Standish Group [62] dressait un état accablant des projets informatiques. Cette étude réalisée à partir d'un échantillon de 365 sociétés et/ou organisations totalise 8380 applications. Celle-ci établit que :

- 16.2 % des projets sont conformes aux prévisions initiales ;
- 52.7 % subissent des dépassements en coût et en délai d'un facteur 2 ou 3 avec diminution du nombre des fonctions offertes ;
- 31.1 % sont tout simplement abandonnés pendant le développement.

Le deuxième rapport publié en 2006 [63] a été réalisé sur la base de 50000 projets informatiques et a duré 12 ans. Cette étude mentionne une légère amélioration de la situation, 75 % subissent un dépassement de coût ou de délai ou sont abandonnés. Cette constatation reste préoccupante et alarmante.

Ces statistiques peuvent être interprétées de différentes façons suivant le maître d'ouvrage et le maître d'œuvre, cependant elles traduisent sans contestation possible l'immatunité actuelle des diverses activités informatiques au sein des entreprises. Cette immaturité est la conséquence directe de la jeunesse de l'informatique par rapport à l'industrie qui depuis près de 150 ans développe méthodes, techniques et outils; mais elle est aussi liée intrinsèquement au caractère dynamique du monde informatique et des évolutions technologiques que connaît notre société actuelle (téléphone portable, satellite de communication, Internet, progiciel de gestion intégrée...).

Une prise de conscience collective des dirigeants des grandes compagnies a été amorcée depuis les années 80 pour faire face aux dérives et aux coûts faramineux des projets informatiques. Cette prise de conscience a permis de rejoindre les initiatives des communautés de recherche. En effet, afin de faire face aux progrès nécessaires, de nombreuses communautés scientifiques et de professionnels de l'informatique ont vu le jour dans les années 80 s'intéressant alors aux concepts des qualitiens Deming [109], Crosby et Juran. Pendant les années 60, le développement du système d'opération 360 d'IBM courait à sa perte. Watts Humphrey [64] fut appelé à la rescousse. Prenant la direction du projet, il s'inspire de la qualité totale développée pour l'industrie, applique ses concepts et finalement réussit à sauver le projet faisant alors entrer par la même occasion le management de la qualité par les processus dans le monde de l'informatique.

2.3 La gestion des processus

2.3.1 La collaboration difficile entre les acteurs définissant les processus

La définition et l'exécution des processus métiers de l'entreprise nécessite la collaboration d'utilisateurs ayant des profils différents, et peu habitués à collaborer. Comme le montre la figure 2.1, ces utilisateurs manipulent des concepts et utilisent des outils de natures complètement différentes :

- Les équipes méthodes définissent les recommandations pour la modélisation des processus métier. Il peut s'agir d'équipes internes à l'entreprise, ou d'organismes de normalisation. Les équipes méthodes définissent un formalisme pour la modélisation des processus

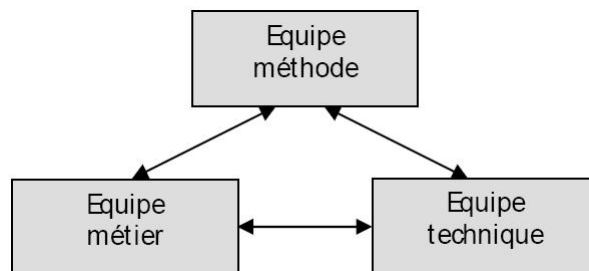


Figure 2.1 – Les différentes équipes intervenant dans la gestion des processus

métiers, et peuvent définir des processus normalisés de manière verticale (spécifique à un corps de métier) ou horizontale (indépendante du corps de métier de l'entreprise).

- les décideurs et analystes métiers définissent les processus métiers de haut niveau, les cas d'utilisation et les scénarios détaillés, en s'appuyant sur les recommandations des équipes méthode. Ils doivent avoir la possibilité de suivre le déroulement des processus via des points de contrôle métier. Des outils permettent de représenter ces processus métiers de manière très complète, mais ces outils sont encore peu utilisés. Des supports hétéroclites sont utilisés pour la modélisation : documents Word, Powerpoint ...
- les équipes techniques transposent ces processus en termes d'applications, services et intégration de l'existant, en essayant de capitaliser sur le Système d'Information (SI) de l'entreprise. Le langage UML et ses différents diagrammes sont utilisés pour la modélisation : diagrammes Use Case pour représenter les cas d'utilisation, diagrammes de classes pour les détails d'implémentation... [29]

La collaboration entre ces différents acteurs est actuellement difficile. Tout d'abord, ces acteurs manipulent des concepts différents : un diagramme UML a par exemple peu de signification pour un analyste métier. De plus, les outils ne fournissent pas toujours les moyens pour cette collaboration.

Actuellement, que ce soit dans les phases de modélisation métier et technique, ou d'implémentation, des outils de conception sont rarement utilisés de bout en bout, de la modélisation à l'exécution. On retrouve un écart entre les phases de spécification et d'implémentation, même dans la partie technique.

De ce fait, il est difficile pour les décideurs d'avoir une vue claire sur les besoins auxquels répondent les applications du système d'information, et donc de justifier les nouveaux investissements sur les technologies de l'information (IT). La maîtrise des processus est actuellement dans les mains des équipes techniques, pas des personnes qui les définissent [29].

2.3.2 Un "Time to Market" trop important

Comme le montre la figure 2.2, le cycle de vie d'un processus est composé de quatre étapes : modélisation du processus (métier, fonctionnelle et technique), déploiement, exécution se basant sur les applications existantes de l'entreprise et interaction des utilisateurs avec ce processus (contrôle d'exécution, optimisation) [29].

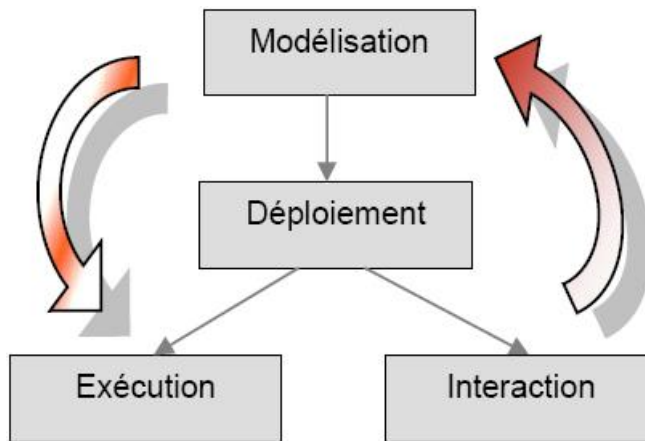


Figure 2.2 – Cycle de vie d'un processus

Le "Time to Market" est le temps qui sépare la décision de conception d'un produit nouveau de sa mise à disposition sur le marché. La réduction du "time to market" est à l'origine de la conception simultanée du produit et du process. Actuellement, le "Time to Market" est beaucoup trop important pour passer de la modélisation à l'exécution des processus, puis pour modifier les processus existants et rendre ces modifications effectives.

Contrairement au cas idéal représenté dans la figure 2.2, pour réaliser un processus exécutable, il y a actuellement cinq étapes :

- 1) modélisation par les utilisateurs métiers dans un outil ;
- 2) nouvelle modélisation par l'équipe technique – sans pouvoir capitaliser sur les processus modélisés dans l'étape précédente ;
- 3) phase d'implémentation (où la majorité du code est réalisée à la main) ;
- 4) déploiement du code réalisé ;
- 5) recette technique et fonctionnelle.

Les étapes 2 et 3 devraient être améliorées : premièrement, il devrait être possible d'importer les définitions des processus métiers dans les outils techniques pour ajouter le lien avec les applications du Système d'Information, et deuxièmement, les capacités de modélisation métier et technique des outils devraient permettre de déployer directement les processus

modélisés sans passer par une phase d'implémentation. De plus, la rupture existant entre la modélisation métier et la modélisation technique entraîne généralement des décalages entre les besoins exprimés au départ et les applications effectivement réalisées.

Une fois le processus déployé, la phase d'interaction permet de l'optimiser. Cette phase permet, au niveau technique et métier, d'identifier les modifications à effectuer. Pour les réaliser, on entre alors dans une autre phase de modélisation, et ce cycle se répète. Comme il existe une rupture entre la modélisation fonctionnelle et la modélisation technique, il est complexe de maintenir les deux spécifications cohérentes et de les rendre évolutives. Généralement, après trois itérations, il est plus simple de re-développer l'application hébergeant le processus que d'en modifier l'implémentation.

2.4 Les limitations des solutions actuelles

La section suivante présente les principales solutions actuelles pour la gestion des processus d'entreprise.

2.4.1 Le Workflow

Le Workflow [106][61][66] peut être défini comme "l'automatisation de processus métiers par échange de documents, informations et tâches entre acteurs pour action". Le workflow a pour objectif la coordination automatisée de tâches réalisées par des intervenants humains. Le moteur de workflow transfère des documents entre les participants à un processus en leur assignant des tâches (valider le document, effectuer une modification ...). Cette approche pragmatique a l'avantage de l'efficacité : les concepts sont clairs, les outils relativement aisés à mettre en place. Par contre :

- Les participants au processus sont par définition des utilisateurs humains, on ne tient pas compte des applications du système d'information. L'intégration du workflow aux systèmes est une tâche difficile qui nécessite beaucoup de code propriétaire ;
- Les documents et les tâches ne sont pas suffisants pour l'automatisation des processus métiers. Il est nécessaire d'avoir un niveau d'abstraction supplémentaire, où l'on parle plus généralement de services et d'informations.

2.4.2 L'EAI - Enterprise Application Integration

EAI [5][72] signifie "intégration des applications d'entreprise". A l'origine, les progiciels EAI avait pour vocation la collaboration des applications d'une entreprise pour accomplir des objectifs métiers, mais dans les faits leur utilisation est beaucoup plus technique. Les fonctions de l'EAI sont :

- Connectivité : fournir les interfaces d'accès aux applications, généralement par l'utilisation de connecteurs propriétaires difficilement maintenables ;
- Transformation : fournir les services de transformation de données permettant de créer un niveau d'abstraction au dessus des applications du SI – un format pivot pour représenter les données du SI (factures, bons de commande ...), et des transformations pour les mapper vers les formats propriétaires attendus par les applications ;
- Routage : fournir les services permettant de localiser dynamiquement le destinataire d'un message en fonction de son contexte.

L'EAI a eu l'avantage d'apporter une réponse au souci de réutilisabilité des entreprises, en leur permettant de capitaliser sur les applications existantes et de les faire communiquer. C'est une réponse au souci technique d'intégration mais cela ne suffit pas :

- Les fonctionnalités de gestion des processus métiers des outils d'EAI sont généralement complexes à utiliser et dissociées de l'offre technique d'intégration ;
- Les processus sont définis à un niveau technique, et les décideurs n'ont aucune visibilité sur leur SI ;
- Il est impossible de réconcilier cette vision avec la vision Workflow : comment faire intervenir les utilisateurs dans les processus ?

2.4.3 Le B2Bi – Business to Business integration

Les outils de B2Bi [72] visent à définir les processus de collaboration entre entreprises partenaires, partant du principe que les processus intra-entreprise, donc d'EAI, n'ont pas les mêmes contraintes que les processus externalisés. Le résultat est que ces outils fournissent surtout un moyen technique de surveiller une collaboration avec un partenaire : gestion de la sécurité des échanges, fiabilité du transport (le bon de commande a bien été envoyé, et reçu par le partenaire), support des protocoles EDI ... Cela pose un certain nombre de problèmes :

- Ces outils doivent pouvoir collaborer avec les outils d'EAI – les processus B2B se déclinant en un processus interne par participant de la collaboration ;
- Pourquoi avoir deux outils différents pour les processus internes et externalisés ? Ces deux types de processus participent en réalité au même processus métier !

2.4.4 Les progiciels intégrés

Les progiciels intégrés [102] sont une solution "clé en main" : des progiciels tels que ceux de SAP fournissent une solution complète de comptabilité, de facturation ... Le problème qui se pose ici est celui de l'adaptabilité :

- la mise en place d'un tel progiciel demande un long projet de paramétrage et d'adaptation. L'entreprise s'adapte aux processus définis par le progiciel et non l'inverse ;
- pour modifier un processus, il est nécessaire de customiser la plateforme. Cela peut s'avérer coûteux et problématique. De manière plus générale, il n'est pas aisé d'adapter un processus pour répondre à un besoin métier urgent et critique.
- les progiciels changent régulièrement de versions, en stoppant le support des versions précédentes, ce qui force la mise à jour. Le principal problème est que les outils de migration ne prennent pas en compte les customisations du progiciel, qui sont inévitables. Il est donc nécessaire de refaire les customisations sur la nouvelle version après migration.

2.4.5 Les moteurs de règles métiers

Les moteurs de règles métiers (Business Rules Engine) [54] permettent de modéliser les règles métiers de l'entreprise, par exemple "un bon de commande reçu de la part de tel client doit être traité par telle division du service commercial".

Ces outils sont adéquats pour automatiser les processus métiers de l'entreprise. Ils permettent en particulier de formaliser et automatiser les prises de décisions sur la branche du processus à choisir, en fonction du contexte du processus. Par contre, ces solutions doivent obligatoirement être couplées à d'autres outils, permettant de gérer les processus.

2.4.6 Conclusion

Cette présentation des principales solutions actuelles pour la gestion des processus d'entreprise montre, d'un côté, l'inexistence d'une offre globale d'outils adaptés aux entreprises. D'un autre côté, l'offre actuelle est guidée par les fournisseurs et non par le besoin des entreprises. On remarque également le manque de standard, ou de norme, régissant la définition de ces outils pour la gestion des processus d'entreprise.

2.5 La problématique et les objectifs

2.5.1 La problématique

Les sections précédentes (sections 2.2, 2.3 et 2.4) montrent le besoin des entreprises d'organiser leur façon de travailler. Les solutions actuelles semblent ne pas répondre complètement à ce besoin. En effet, ces solutions se focalisent plus sur l'implémentation des processus que sur leur définition et modélisation qui sont deux étapes essentielles pour organiser la façon de travailler d'une entreprise.

La création et la maîtrise d'informations de toute nature dans une entreprise et leur intégration dans un système d'information qui prenne en compte à la fois les aspects stratégiques, techniques, commerciaux, économiques et humains ne sauraient y échapper : il faut être capable de modéliser l'existant, les objectifs et les moyens d'y parvenir. Le développement de la productivité passe par la modélisation.

Il y a aujourd'hui, et pour longtemps encore, matière à réflexion sur la modélisation d'entreprise : depuis plusieurs années, des chercheurs et des industriels travaillent dans cette direction, appartenant à diverses communautés (Mathématiques, Sciences de l'Ingénieur, Sciences Humaines et Sociales...). Les pouvoirs publics ont financé une partie de ces recherches, tant à l'échelle nationale que dans le cadre de projets européens et internationaux.

Notre présent travail est motivé par la recherche d'une façon d'organiser l'entreprise basée sur ses processus métier et de l'adapter en considérant des normes et standards qualité différents. La facilité d'intégration, ou d'utilisation au sein de l'entreprise et la capacité de considérer la réalité de l'entreprise doivent être prises en compte. Ce modèle ne doit pas négliger les différents facteurs pouvant influencer son implémentation à savoir les facteurs humains internes à l'entreprise ou la problématique d'interopérabilité de l'entreprise avec ses clients et/ou partenaires et sous-traitants. Enfin la solution choisie doit être généralisable au sein de n'importe quelle entreprise voulant l'implémenter.

2.5.2 Les objectifs opérationnels de l'entreprise

Comme application de notre méthodologie proposée, la figure 2.3 montre les objectifs opérationnels que notre modèle doit permettre d'implémenter :

la mise en place d'un référentiel qualité : L'objectif du référentiel qualité est de formaliser les bonnes pratiques en se basant sur des normes et des standards qualité. Dans une vision à moyen et à long terme, le référentiel engage tous les intervenants dans la

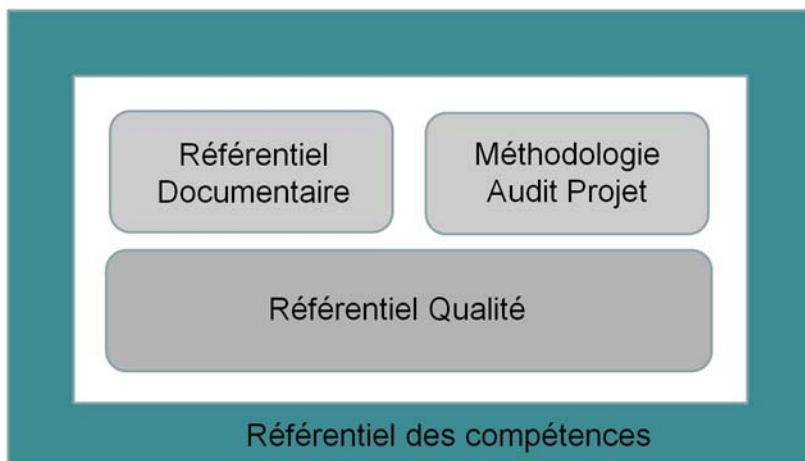


Figure 2.3 – Les objectifs opérationnels de l'entreprise

réalisation des projets, sur des objectifs qui conduisent à assurer le respects des engagements et la réussite des projets. Ainsi, le référentiel qualité fait partie des instruments visant à organiser l'entreprise.

la définition d'une méthodologie d'audit projet : La méthodologie d'audit projet assure la mise en place des bonnes pratiques, mentionnées dans le référentiel qualité, au niveau des projets. L'audit projet permet d'évaluer les écarts du projet en cours par rapport aux bonnes pratiques définies dans le référentiel qualité et de remédier à ces écarts, le cas échéant.

La mise en place d'un référentiel documentaire : Le référentiel projet a pour but de créer une structure type de répertoire des projets. Cette structure contient tous les modèles de documents utilisables tout le long du projet. Ce répertoire type est dupliqué pour chaque projet. Les objectifs du "référentiel documentaire" sont :

- créer une structure unique des répertoires projet ;
- créer un ensemble de modèles de documents utilisable dans tous les projets ;
- standardiser les documents et les méthodes utilisés ;
- accélérer la phase de production des documents ;
- partager les connaissances et les références ;
- s'assurer du respect du référentiel qualité ;
- préparer et faciliter la localisation des preuves d'évaluation et de certification des normes qualité prises en compte dans le référentiel qualité.

la définition d'un référentiel des compétences : Le référentiel de compétences répond à un besoin de cohérence sur les qualificatifs, la définition des postes et le rôle de chaque

intervenant au niveau d'un projet. Il permet :

- d'évaluer objectivement les collaborateurs et leur évolution ;
- de faire correspondre les compétences des collaborateurs avec leurs rôles sur les projets.

2.6 Conclusion

Dans ce chapitre, nous avons présenté deux points principaux : le premier concerne la situation actuelle chaotique de la plupart des projets informatiques et les limitations des solutions actuelles. Le deuxième concerne un exposé de notre problématique qui consiste en la recherche d'une façon d'organiser l'entreprise basée sur les processus et de l'adapter en considérant la facilité d'intégration, ou d'utilisation au sein de l'entreprise et la capacité de prendre en compte la réalité de l'entreprise. Ce modèle ne doit pas négliger les différents facteurs pouvant influencer son implémentation à savoir les facteurs humains internes à l'entreprise ou la problématique d'interopérabilité de l'entreprise avec ses clients et/ou partenaires et sous-traitants. Enfin la solution choisie doit être généralisable au sein de n'importe quelle entreprise voulant l'implémenter.

TROISIÈME PARTIE

CADRE MÉTHODOLOGIQUE

La modélisation d'entreprise

3.1 Introduction

L'objectif global recherché de la modélisation d'entreprise est de construire un système d'information en relation directe avec les besoins propres de la société et répondant par là même aux exigences du marché et de la clientèle. La mise en place de tel projet est généralement réalisée dans le même esprit que les démarches Qualité dans l'entreprise, afin de pouvoir représenter les processus de l'entreprise, les analyser, les améliorer et capitaliser le métier de l'entreprise.

Le système d'information est composé :

- du système informatique : ensemble des logiciels, fonctions et services utilisés dans l'entreprise.
- de tous les participants aux activités de l'entreprise.

Ainsi, le système d'information est le support de la performance de l'entreprise, traduction de sa stratégie et de son organisation. L'étude du système d'information se place dans une démarche d'amélioration globale et durable des performances de l'entreprise. Dans ce contexte, le schéma directeur doit suivre une démarche rigoureuse et structurée afin de définir une architecture informatique cible et une trajectoire existant-cible.

La première partie d'une telle démarche consiste à établir un **Modèle de Référence** qui offre la garantie de couvrir l'ensemble des domaines de l'entreprise. Ce référentiel a pour but de déterminer les activités nécessaires à la réalisation des finalités de l'entreprise et d'identifier les processus qui permettent de satisfaire les attentes du marché et des clients. Ce modèle représente le niveau conceptuel du système d'information cible décrit au niveau logique, par son architecture informationnelle et applicative. La seconde partie permet de "concrétiser" le

chemin à parcourir entre le système existant et le futur système cible. Un état des lieux des structures, des procédures et du système d'information, réalisé par une évaluation de l'existant, permet d'examiner la couverture du modèle à la fois par l'organisation et les moyens informatiques. Cette démarche met en évidence les axes prioritaires d'action dans l'entreprise, notamment aux niveaux organisationnels et structurels, qui constituent un pré-requis à la mise en place de l'outil informatique.

3.2 Système d'information : Approche systémique de l'entreprise

Une définition préalable des termes employés (système, analyse systémique, modélisation...) est nécessaire à la bonne compréhension du sujet.

Cette première étape permet également de fixer l'esprit et le contexte dans lequel se situe la démarche à adopter dans l'étude d'un projet de modélisation d'entreprise. Une réflexion sur la méthodologie permet d'aborder les besoins en modélisation de l'entreprise et les moyens d'y parvenir.

3.2.1 Définitions

Un système est un élément identifiable qui réalise une activité, une fonction, qui est doté d'une structure, évoluant dans le temps et dans un environnement pour finaliser un besoin formulé. Un système est donc un ensemble d'éléments en interaction dynamique, organisés en fonction d'un but [49].

L'analyse systémique consiste à définir les limites du système à modéliser, à en identifier les éléments importants et les types d'interaction entre ces éléments, puis à déterminer les liaisons qui les intègrent en un tout organisé. Éléments et types de liaisons sont classés et hiérarchisés [69].

La modélisation consiste à construire un schéma complet des relations causales entre les éléments des différents sous-systèmes. La simulation étudie le comportement dans le temps d'un système [108].

Hypothèses dans le cas des entreprises :

- L'objet à modéliser est supposé doté d'au moins un projet identifiable et ne se contente pas d'obéir uniquement à d'éventuelles lois causales. Le fonctionnement et l'évolution s'interprètent par des projets qui détermineront eux-mêmes les structures possibles.

- L'objet doit être décrit dans sa totalité, fonctionnant et évoluant. L'objet est ouvert sur l'environnement qui est représenté même si la description n'est pas exhaustive.

3.2.2 Le modèle d'entreprise

L'entreprise est représentée, dans un premier temps, comme une boîte noire assurant une fonction de transformation de ressources (voir figure 3.1), où apparaissent entre autres les flux physiques (matières premières et produits manufacturés) et les flux financiers (paiements fournisseurs, règlements clients). La liste des flux n'est pas exhaustive car on peut avoir différents types de flux comme les flux d'information par exemple [69].

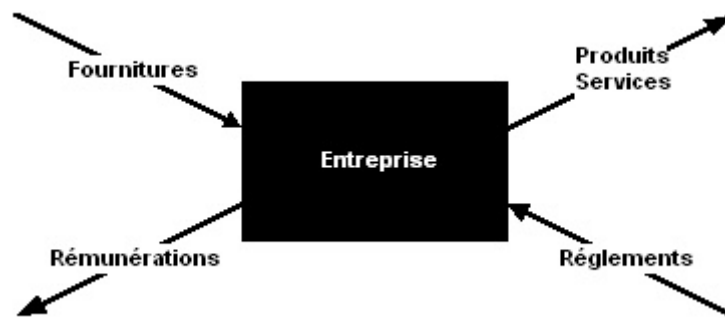


Figure 3.1 – Entreprise "Boîte Noire"

Dans un second temps, le contenu de la boîte est défini par la prise en compte des flux faisant apparaître les fonctions de production, de mémorisation et de pilotage.

La modélisation de l'entreprise (voir figure 3.2) se décline en trois sous-systèmes :

- Le système opérant, où figurent les fonctions de transformation des flux relatifs aux produits et services, spécifiques de l'activité de l'entreprise.
- Le système de pilotage, où figurent les fonctions de pilotage et de suivi des ressources, de définition des règles de transformation et d'adaptation à l'environnement.
- Le système d'information qui regroupe les informations, utilisées tant par les fonctions de transformation que de pilotage.

3.2.3 Description du système d'information

Ce système (voir figure 3.3) traite les principaux flux d'informations avec d'une part le système de décision et le système opérant (règles de fonctionnement, ressources allouées, prio-

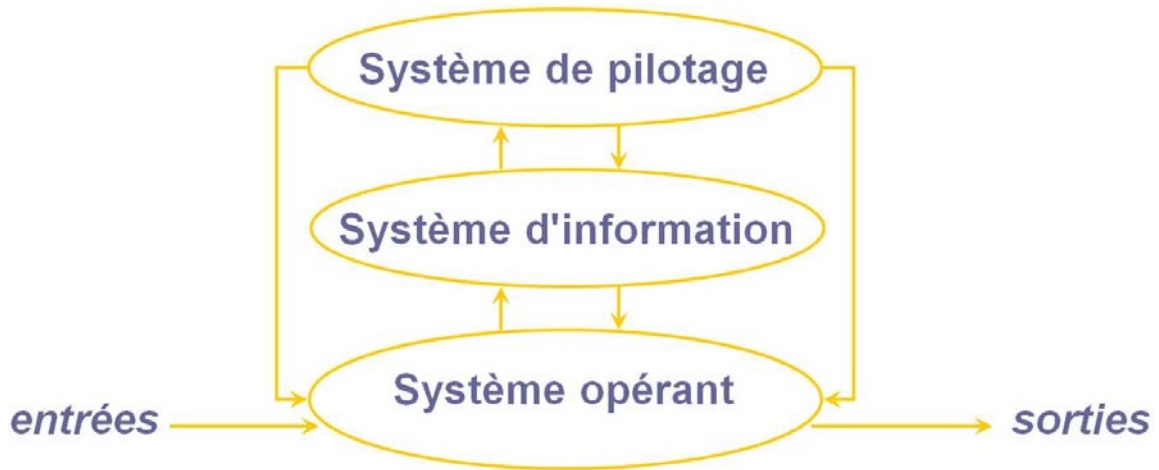


Figure 3.2 – Entreprise "Boîte Blanche"

rités d'exécution, mesure de l'activité, de l'efficacité technique et commerciale...) et d'autre part avec l'environnement.

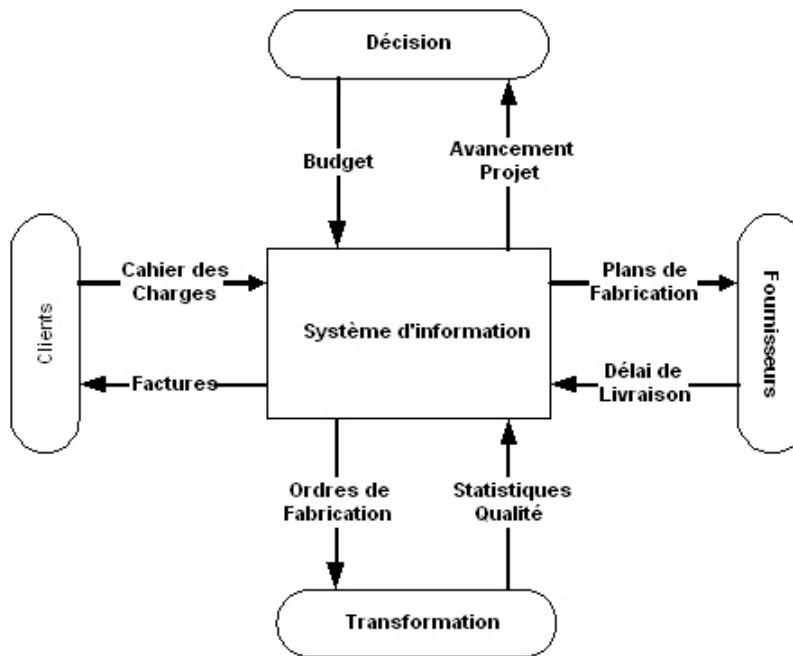


Figure 3.3 – Système d'Information

Les flux d'information avec les pôles clients et fournisseurs sont les représentations (commandes, factures...) des flux physiques et financiers traités par les systèmes opérants et de

décision.

Le concept de système d'information d'une entreprise recouvre donc deux notions :

- la réalité de l'entreprise se transformant, agissant, communiquant et mémorisant des informations, notion qui apparente le système d'information à un objet naturel ;
- le système, objet artificiel, construit par l'homme pour représenter les activités, la communication et la mémorisation dans l'organisation.

Cette opposition dans la définition du système d'information offre l'intérêt de conditionner l'approche méthodologique de conception : partir de l'existant puis définir les objectifs ou bien le sens contraire. Le système d'information, une fois construit, est partie intégrante de l'organisation, support de la performance de l'entreprise.

3.3 Système d'information : Approche fonctionnelle de l'entreprise

Dans cette approche, on s'intéresse aux fonctions qu'un système d'information propose à ses utilisateurs [12]. On considère qu'un « SI doit garantir que la bonne information est disponible au bon endroit et au bon moment ». Le système d'information a naturellement pour objet principal l'information. C'est l'échange d'information qui permet aux utilisateurs de travailler ensemble. Pour que ce travail soit optimal et efficace, la bonne information doit être disponible au bon endroit (bon utilisateur) et au bon moment. C'est la valeur ajoutée d'un système d'information dans l'entreprise.

On peut détailler davantage ce qu'offre un système d'information à un utilisateur : un système d'information est un ensemble organisé de ressources [91] :

- matériels (ordinateurs, papier...);
- personnel (clients, responsables...);
- connaissances (modèles, règles...);
- logiciels et procédures (applications informatiques, méthodes de travail...).

qui permet de :

- acquérir des informations ;
- traiter (organiser) des informations ;
- stocker (mémoriser) des informations ;
- communiquer des informations.

Nous ajoutons à ces fonctions celle d'aider les acteurs de l'organisation dans l'exercice de leur métier, en particulier accompagner la prise de décision au sein de l'organisation. « La raison d'être d'un système d'information est l'accès au bon moment à la bonne information pour

prendre la bonne décision » [93]. Cette dernière définition montre que le système d'information s'inscrit dans un processus d'aide à la décision.

3.4 Système d'information : Approche structurale de l'entreprise

La figure 3.4 permet de voir un SI comme la composition de deux sous-systèmes : le système de traitement de l'information (comprenant les acteurs, les données et les processus) et le système informatique (comprenant les ressources matérielles et logicielles, les bases de données et les fonctions) [79]. L'originalité de cette définition du système d'information est qu'elle met l'accent sur les liens forts entre processus et système d'information. En effet, dans le système de traitement de l'information qui est composé d'acteurs, information et processus « Le processus est un plan d'ensemble indiquant comment les acteurs collaborent au moyen des informations gérées pour accomplir l'objectif de production... » [80].

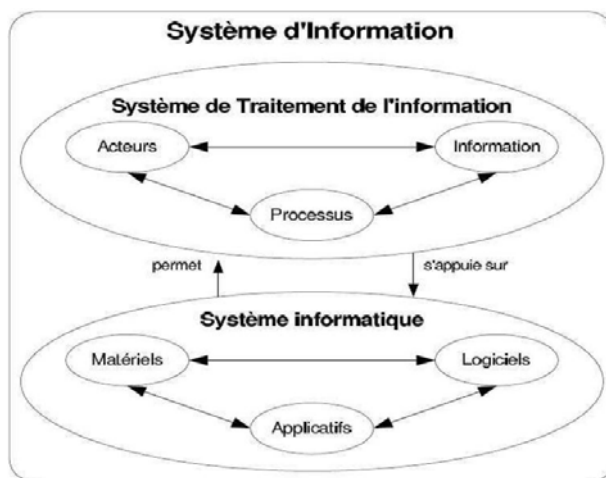


Figure 3.4 – Une approche structurale d'un système d'information

Nous allons préciser les différentes définitions du terme « processus » et son rôle capital dans la définition d'un système d'information dans la suite de ce manuscrit. Cette approche positionne le système informatique comme l'infrastructure du système d'information [79]. L'informatique joue naturellement un rôle important dans le développement des systèmes d'information, mais elle ne présente que la partie physique (applicative) d'un SI. Un système d'information d'entreprise contient aussi une partie non-informatique (archive papier, téléphone...).

3.5 La modélisation d'entreprise

La modélisation d'entreprise est une discipline qui consiste à décrire l'organisation et les processus et activités d'une entreprise soit dans le but de simuler ces processus pour comparer divers scénarios et aider la prise de décision, soit dans le but d'analyser les processus et de les restructurer pour améliorer la performance et le fonctionnement de l'entreprise.

La modélisation d'entreprise s'applique aussi bien aux entreprises de production de biens manufacturés (production continue ou discrète) qu'à celles fournissant des services. Les processus modélisés peuvent correspondre à des procédures administratives, techniques (processus productifs) ou de support de l'entreprise.

Le champ de la modélisation d'entreprise peut dépasser largement le cadre d'une seule entreprise pour recouvrir les relations d'un réseau d'entreprises ou couvrir une chaîne logistique complète (cas de l'entreprise étendue ou virtuelle).

La **modélisation d'entreprise** a pour objet la construction de modèles d'une partie déterminée d'une entreprise pour en expliquer la structure et le fonctionnement ou pour analyser le comportement [108].

Dans la pratique, les principales motivations qui justifient une étude de modélisation en entreprise sont les suivantes :

- comprendre et analyser la structure et le fonctionnement de l'entreprise ;
- prévoir (de manière fiable) le comportement et les performances des processus opérationnels avant leur implantation ;
- choisir la (ou les) meilleure(s) alternative(s) d'implantation ;
- identifier les risques d'implantation à gérer ;
- justifier les choix d'implantation sur des critères liés aux ressources et aux coûts (méthodes de comptabilité par activité, par exemple) ;
- bâtir une vision commune du fonctionnement de l'entreprise et la communiquer facilement au plus grand ensemble possible du personnel.

Si l'on vise l'amélioration de l'intégration dans l'entreprise, le dernier point est de loin le plus important. Cependant, le récent engouement pour les techniques de Business Process Management (BPM) a largement contribué à l'essor des techniques de modélisation et d'analyse de l'entreprise, car elles leur sont sous-jacentes. Le management est perçu comme une remise en cause (pouvant être fondamentale) des processus opérationnels de l'entreprise dans le but d'apporter des gains significatifs en termes d'efficacité et de productivité. Il est donc nécessaire d'établir ce qu'il convient d'appeler la "carte des processus" pour comprendre le fonctionnement de l'entreprise, les interactions entre ses divers composants et la synchronisation des flux.

Pour nous, le terme "entreprise" s'entend au sens large. Il couvre aussi bien les entreprises du secteur industriel (usines de produits manufacturés, raffineries, usines chimiques, aciéries. . .) que les banques, les compagnies d'assurance, les cabinets, les administrations, les sociétés de service. . . **Dans le cas de la modélisation d'entreprise, ce terme est utilisé pour désigner ce que l'utilisateur considère être une entreprise, c'est à dire tout ou partie d'un système socio-économique donné.**

Ainsi, le modèle d'entreprise pourra être une description d'un système industriel, d'un atelier de production, d'une usine, d'un département, d'un service ou d'une chaîne client-fournisseur, suivant les besoins de l'utilisateur.

Un **modèle** est par définition une représentation simplifiée consensuelle d'une partie du monde réel, exprimée dans un langage de représentation dans un but défini. Ce langage peut être :

- formel, c'est à dire ayant une syntaxe et une sémantique bien définies comme la logique du premier ordre ou le langage B ;
- semi-formel, la plupart des formalismes sont semi-formels, exemple UML ;
- informel, description en langage naturel (texte, schéma. . .).

Un **modèle d'entreprise** est un ensemble de modèles décrivant divers aspects d'une entreprise que l'on souhaite analyser. Parmi ces aspects, on peut citer les tâches à réaliser, les informations à échanger ou à stocker, la structure du réseau informatique, la disposition des machines dans les ateliers ou encore la structure organisationnelle de l'entreprise. Le contenu de ces modèles et le degré de détail dépendent de l'utilisation que l'on veut en faire. Il appartient donc à l'utilisateur de définir précisément ce qui doit être contenu dans le modèle et ce qui ne doit pas l'être en fonction de la finalité désirée.

Activité d'entreprise : Une activité est l'accomplissement d'une tâche. Il s'agit en général d'une séquence d'opérations devant être exécutées en totalité par une ou plusieurs ressources et ceci dans un temps donné pour réaliser la tâche spécifiée [49].

Il existe deux grands groupes d'activités :

- les activités d'exécution (ou de transformation) : ce sont des activités qui ont pour objet de transformer des objets d'entrée en objets de sortie. Elles ont en général un caractère procédural voire routinier que l'on peut décrire sous forme d'un algorithme ou d'une procédure et sont pour la plupart automatisables ;
- les activités de prise de décision (ou cognitives) : ce sont des activités de nature cognitive (analyse et résolution de problèmes), souvent réalisées par des opérateurs humains, dont on ne connaît pas toujours le fonctionnement et donc difficiles à modéliser et à automatiser.

Processus opérationnel (ou processus métier, en anglais business process) : C'est une succession de tâches qui contribue à la réalisation des objectifs de l'entreprise. De manière générale, un processus peut être défini comme un enchaînement d'activités à exécuter pour atteindre un but donné. Cet enchaînement forme ce qu'il est convenu d'appeler le flux de contrôle du processus, c'est à dire sa logique d'exécution.

Par exemple, on parlera du processus de fabrication d'un produit comme une suite prédéfinie d'activités d'usinage et d'assemblage. On peut parler aussi du processus de conception d'un produit comme un ensemble d'activités concourant à la définition puis à la spécification technique du produit et de ses composants. Il s'agit dans les deux cas de processus opérationnels.

3.6 Les "framework" de modélisation d'entreprise

3.6.1 Qu'est ce qu'un framework de modélisation d'entreprise ?

Un framework de modélisation d'entreprise :

- identifie les types d'information requis pour modéliser une entreprise ;
- organise les types d'information en structure logique ;
- décrit les rapports entre les types de l'information.

Cette catégorisation des informations sous forme de framework facilite la conception de l'architecture de l'entreprise et l'échange des données. Le framework peut aussi aider à identifier les modèles requis pour décrire l'architecture de l'entreprise et montrer les liaisons entre les différents types d'information circulant au sein de l'entreprise comme les processus métier, les données... L'utilisation d'un framework pré-défini accélère la modélisation des processus. Le framework de modélisation d'entreprise n'est pas un modèle d'entreprise en lui-même mais indique comment organiser et décrire les processus de l'entreprise [55].

Un framework doit, au minimum, organiser les types d'information et décrire les relations entre les processus métier, les données, le système d'information et l'infrastructure utilisée pour réaliser les objectifs futurs de l'entreprise. Un framework doit aussi respecter les standards permettant de relier le modèle à la stratégie et aux objectifs de l'entreprise. Des normes de qualité telles ISO 9000v2000 ou CMMI peuvent être considérées comme des standards que doit respecter un framework, ou du moins faire apparaître les champs couverts par ces normes.

3.6.2 Le Framework de Zachman - Les principes

La majorité des travaux sur les frameworks de modélisation d'entreprise se sont inspirés des travaux de John Zachman publiés en 1987 lorsqu'il travaillait chez IBM [77] et [101]. Ces travaux ont évolué après pour donner naissance au framework de Zachman (voir la figure 3.5). Ce framework est organisé sous forme d'une matrice de six colonnes et cinq lignes. Le cadre conceptuel couvre les différentes problématiques de l'entreprise formulées de la façon suivante sous forme de six interrogations :

- Quoi pour les données : que traite-t-on ?
- Comment pour les fonctions : comment traite-t-on ?
- Où pour le réseau, organisation : où traite-t-on ?
- Qui pour les personnes : qui est impliqué ?
- Quand pour le temps : quand traite-t-on ?, selon quels cycles ?
- Pourquoi pour les objectifs, contraintes : pourquoi veut-on traiter ?

Les cinq lignes représentent les vues des utilisateurs du framework : le planificateur, le propriétaire, le concepteur, le développeur, et le sous-traitant [110].

Dans ce modèle, les perspectives, rattachées aux niveaux stratégique, métier, logique, technologique et technique, ne sont pas étanches mais se complètent sur les niveaux adjacents. Elles décrivent respectivement les objectifs de l'organisation, ses processus métier et leurs relations, les fonctions que le Système d'Information doit supporter, les spécifications techniques, et les composants déployés.

Comme le montre la figure 3.6, dans la pratique, l'application du concept du framework permet donc de se focaliser sur une problématique particulière, la cellule, sans jamais perdre de vue le contexte général (lignes et colonnes adjacentes). La cohérence d'ensemble tient au fait que les choix de construction ne s'effectuent plus de manière dissociés. Chaque niveau (perspective) "pèse" sur les couches adjacentes dans la définition des exigences et la recherche d'une solution globale.

3.6.3 Vue d'ensemble des principaux framework

3.6.3.1 Federal Enterprise Architecture Framework (FEAF)

Le Federal Enterprise Architecture Framework (FEAF), sponsorisé par le Federal CIO Council, définit huit composants pour un framework [95] [1] : drivers de l'architecture, direction stratégique, architecture actuelle, architecture cible, processus transitoires, segments architecturaux, modèles architecturaux et les standards (comme le montre la figure 3.7). La

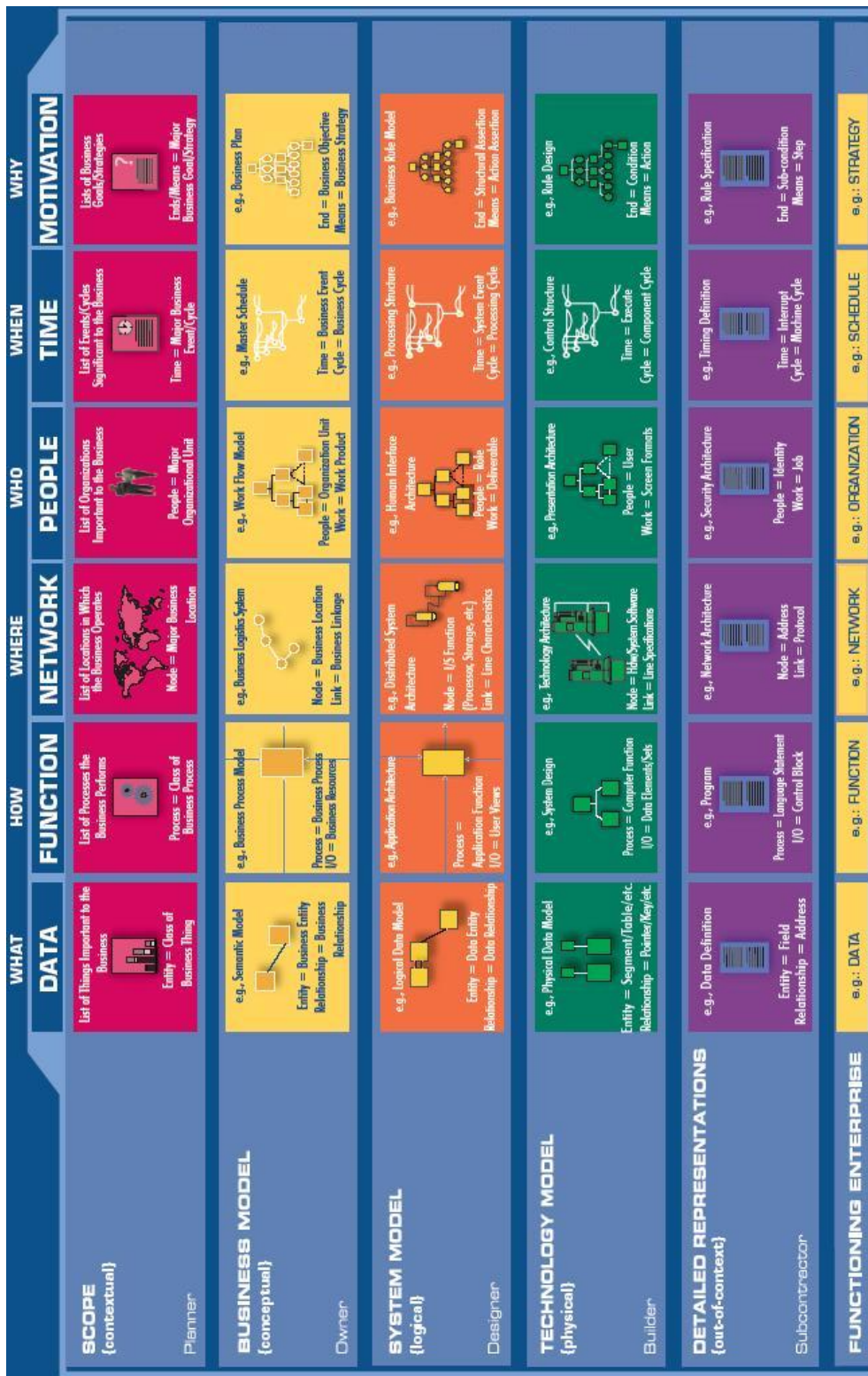


Figure 3.5 – La première partie du Framework de Zachman

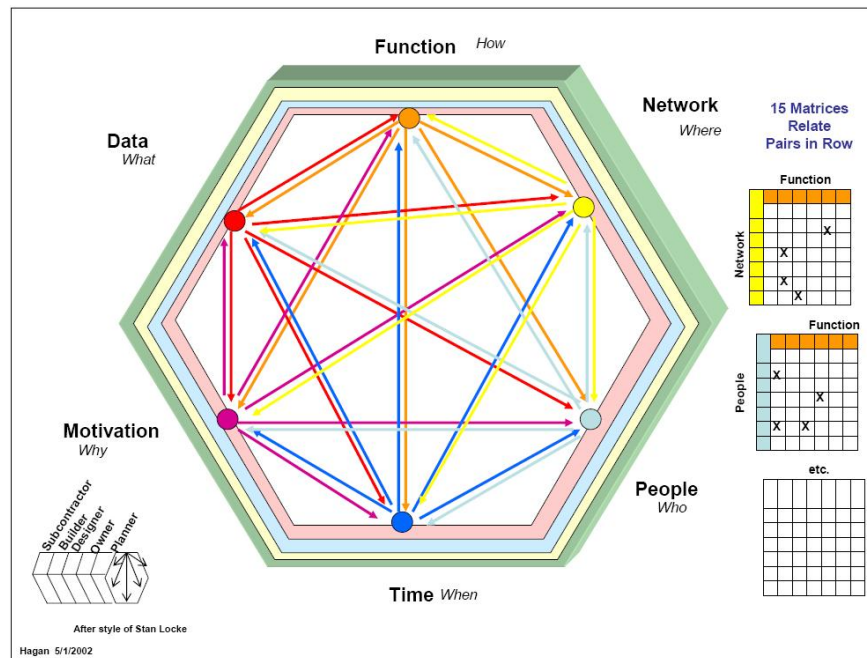


Figure 3.6 – Les relations d’une cellule de Zachman à travers une même ligne

direction stratégique guide le développement de l’architecture cible et inclut les principes, les buts et les objectifs. Les modèles architecturaux définissent le métier, les données, les applications les architectures technologiques. Ce framework s’inspire fortement du framework de Zachman.

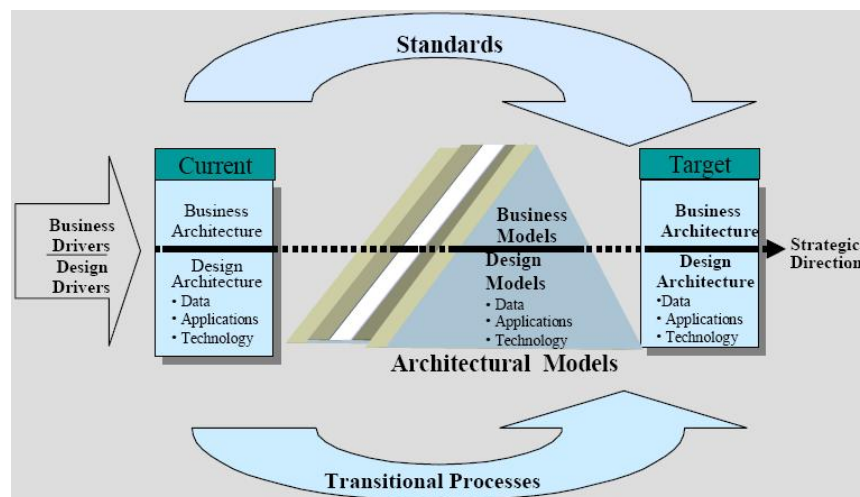


Figure 3.7 – Les huit composants du Federal Enterprise Architecture Framework

3.6.3.2 DoD Architecture Framework (DODAF)

Le DODAF a été développé initialement comme le Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) Architecture Framework (AF) [35]. Le DODAF a été publié en Juin 1996 pour être utilisé par la communauté de C4ISR. Aujourd'hui, son utilisation est exigée à travers tout le DoD. Ce framework est largement utilisé dans les organisations civiles et internationales. Le DODAF a été initialement développé pour permettre de comparer des descriptions architecturales des systèmes de systèmes très complexes et qui doivent être interopérables.

Comme le montre la figure 3.8, le DODAF est organisé autour de trois vues :

Une vue opérationnelle : C'est la description des tâches, activités, éléments opérationnels et l'échange d'information requis pour accomplir une mission ;

Une vue système : C'est une description, incluant des graphiques, des systèmes et les interconnexions pour implémenter une fonction ;

Une vue technique : C'est l'ensemble minimal de règles régissant les interactions et l'interdépendance des éléments du système pour assurer sa conformité aux besoins.

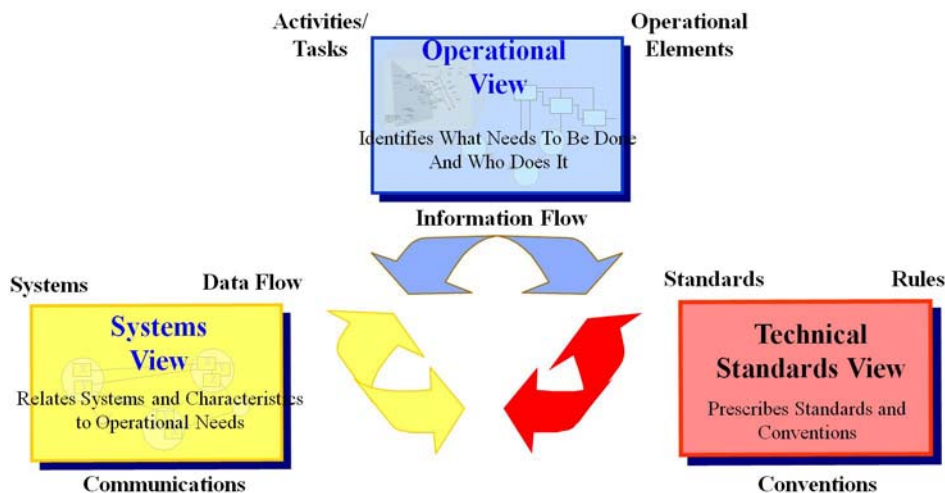


Figure 3.8 – Les vues opérationnelle, système et technique du DODAF

3.6.3.3 Autres Frameworks

Parmi les principaux frameworks, on peut citer :

- Treasury Enterprise Architecture Framework (TEAF) publié en juillet 2000 comme guide au département de trésorerie américain et de ses bureaux [2] ;

- National Association of State Chief Information Officers (NASCIO) [3];
- The Open Group Architectural Framework (TOGAF) [4].

3.7 Conclusion

La modélisation d'entreprise apparaît comme une nécessité qui concerne aussi bien les grandes, moyennes et petites structures. Les raisons d'un tel besoin sont variées :

- L'implémentation d'un système d'information ;
- L'obtention d'une certification ;
- La mise en place d'une nouvelle structure organisationnelle inter et intra entreprises ;
- L'implémentation d'un pont entre les processus et le système d'information ;
- L'interopérabilité entre les différents départements d'une entreprise ;
- La réduction de la complexité organisationnelle ;
- L'illustration et visualisation des flux de données échangés inter et intra entreprises.

Les défis pour la mise en place d'une démarche de modélisation d'entreprise sont nombreux :

- L'existence de différentes méthodologies de modélisations ;
- L'existence de différents langages de modélisations ;
- Les différents environnements : organisations, pays... ;
- La réticence des employés à être observés.

Les principales méthodes de modélisation d'entreprise

4.1 Introduction

Les Méthodes de Modélisation d'Entreprise (MME) ont fait leur apparition sous différentes appellations dès le début des années 80. On peut citer parmi les premiers travaux le programme ICAM (Integrated Computer Aided Manufacturing) de l'U.S. Air Force qui a produit à cette époque la méthode IDEF (ICAM DEFinition). La définition d'une MME diffère d'un ouvrage à un autre mais celle qui peut résumer les différentes approches est la suivante : Une méthode de modélisation d'entreprise consiste à *"décrire l'organisation des processus d'un système soit dans le but de les simuler pour comparer divers scénarios, soit dans le but de les analyser et de les restructurer pour améliorer la performance dudit système"* [108]. Une MME comporte les éléments suivants :

- un modèle conceptuel de référence, définissant les objectifs de l'entreprise, ses fonctionnalités, sa structure, son environnement et son comportement dynamique ;
- des formalismes de modélisation, permettant de représenter les différents concepts du modèle conceptuel de référence, de préférence de façon graphique, pour des raisons de clarté ;
- une démarche, qui structure les étapes d'élaboration des différents modèles (actions à mener, validations à effectuer) [108].

Comme le montre la figure 4.1, il existe des composants méthodologiques supplémentaires qui sont :

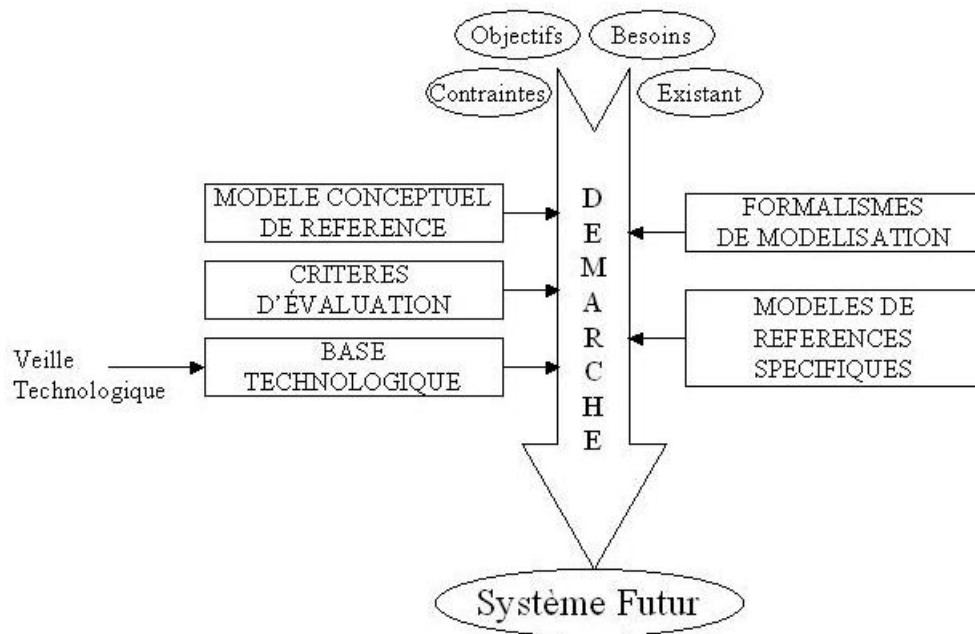


Figure 4.1 – Les éléments d'une Méthode de Modélisation d'Entreprise

- des modèles de références spécifiques, qui décrivent des classes de systèmes de production ;
- des critères d'évaluation permettant d'évaluer les performances de l'entreprise conçue ;
- des bases technologiques qui pourront être utilisées pour construire la solution.

Ce chapitre est consacré à la présentation d'un aperçu des principales méthodes de modélisation d'entreprise. Nous commençons par présenter la notion de processus métier. Ensuite, nous proposons une classification des principaux langages de modélisation.

4.2 Les processus métier

4.2.1 Qu'est ce qu'un processus métier ?

Les processus métier sont les processus représentatifs des activités de l'entreprise indépendamment des moyens humains et techniques. Ces processus interfèrent de manière transversale dans le système d'information et peuvent même traverser les frontières organisationnelles internes de l'entreprise. Les processus métier traversent même les limites de l'entreprise pour collaborer avec les partenaires comme les fournisseurs et les distributeurs [76].

Chaque processus métier est identifié par au moins un objectif et son degré de succès

qualitativement ou quantitativement mesurable.

Différentes définitions ont été énoncées sur le sujet des processus métier. Ce qui émerge de ces définitions est ce qui suit. **Les activités** d'un processus métier sont exécutées par **des acteurs** jouant **des rôles** particuliers, consommant quelques **ressources** et produisant d'autres. Les activités peuvent être déclenchées par **des événements** et peuvent à leur tour produire des événements. **Les activités** d'un processus peuvent être liées par **des dépendances de ressource** (dépendances de producteur-consommateur) ou **des dépendances de commande** (une activité déclenchant une autre). **Les acteurs** opèrent à l'intérieur des frontières **des organisations** qui exécutent **des fonctions** métiers spécifiques. **Les rôles** peuvent soutenir **des fonctions**. La figure 4.2 montre un méta-modèle de processus [76].

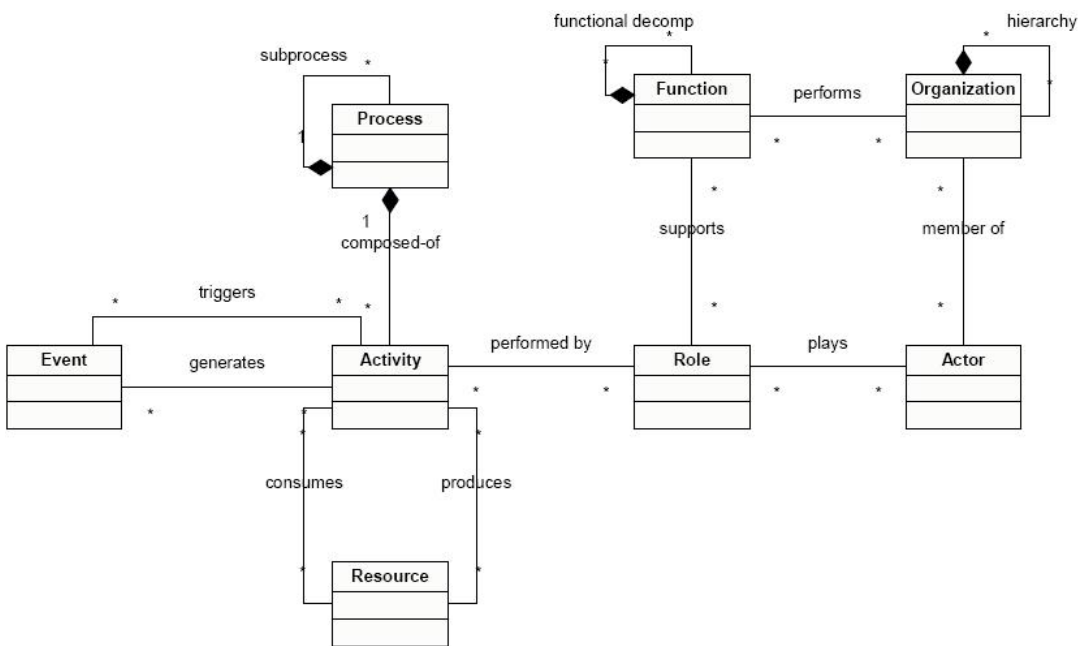


Figure 4.2 – Un méta-modèle de processus [76]

4.2.2 Pourquoi parler des processus métier ?

Habituellement, un métier est considéré comme une organisation hiérarchique qui reflète la décomposition fonctionnelle de l'entreprise et sa chaîne de commande. Les différents départements sont spécialisés dans des fonctions spécifiques (par exemple, vente, production ou comptabilité), et dans chaque département, des sous-départements, équipes ou individus sont spécialisés dans des sous-fonctions. Ainsi par exemple, le traitement d'une commande d'un client traverse les frontières de différents départements : ventes (pour recevoir l'ordre), plani-

fication (pour planifier la fabrication du produit ou la réalisation de l'inventaire), production, expédition, et comptabilité.

Les premières théories de gestion d'entreprise se sont concentrées sur le fonctionnement de l'entreprise et le contrôle de ses départements efficacement (chaîne de la commande, déroulement des opérations, communication. . .) mais en se focalisant sur chaque département à part. Avec l'apparition du BPM (Business Process Management) et le BPR (Business Process Re-engineering), une révolution a eu lieu : au lieu de se concentrer sur chaque fonction de l'entreprise séparément - et de ce fait ne pas remettre en cause la structure globale des processus - les chercheurs ont commencé à préconiser qu'on devrait regarder les processus métier en entier de bout en bout, cherchant ainsi à optimiser le processus en entier. Ce qui veut dire que les processus métier ne traversent pas les mêmes frontières de l'entreprise, ou qu'ils les traversent mais de manières différentes. Pratiquement, cette tendance a été à l'origine de l'apparition de nouvelles structures organisationnelles telles que *la gestion orientée projet* où des équipes de projet sont constituées à partir de différentes unités pour gérer tous les aspects d'un projet ou encore *l'organisation matricielle* où les équipes métier (les colonnes de la matrice) offrent des services aux clients externes en manageant les équipes réalisant les services (les lignes de la matrice) [32].

4.2.3 Une ontologie d'entreprise

Les entreprises ont un grand nombre de processus concourants. Le conseil d'administration a ses propres processus prédéfinis pour la prise de décision, pour nommer le personnel et les membres du conseil. Le directeur aura ses propres processus de décision et de distribution des responsabilités. Chaque département fonctionnel ou division (comptabilité, vente, production, service à la clientèle. . .) aura ses propres processus. . . Chaque niveau de l'organisation aura ses propres objectifs, ses propres processus, ses propres mesures de performances. Ainsi pour parler de "modélisation de processus métier", nous devons identifier quels sont les processus qui nous intéressent, le niveau de détail et les éventuels rapports entre ces processus.

Ould distingue entre *les processus noyau*, *les processus de management* et *les processus support* [86]. En effet, *les processus noyau* sont liés à la fourniture du produit ou du service aux clients et représentent le cœur du métier. Par exemple, pour une société de service, ces processus sont appelés les processus de réalisation. *Les processus de management* sont les processus de direction qui permettent de piloter l'entité (société ou région) et d'en améliorer le fonctionnement. *Les processus support* contribuent au bon fonctionnement de tous les processus, ça concerne par exemple la gestion des collaborateurs. En pratique, nous analysons :

- *les processus noyau* pour améliorer la satisfaction du client ;

- *les processus support* pour améliorer l'efficacité de l'entreprise ;
- *les processus de management* pour améliorer la structure d'entreprise.

Gale et Eldred considèrent l'entreprise comme un système hiérarchique multi-couches, combinant ainsi les caractéristiques des systèmes hiérarchiques et des systèmes multi-couches [50]. Un système hiérarchique est un système global où chaque noeud dans la hiérarchie d'agrégation commande et coordonne ses composants immédiats. La figure 4.3 illustre un exemple d'un système hiérarchique, tel que l'organisation du travail dans une usine :

- un responsable d'équipe commande le travail des ouvriers ;
- un responsable de ligne de production commande le travail des équipes de production ;
- un manager d'opérations commande le travail de diverses chaînes de production.

Dans un système hiérarchique, le travail d'un agrégat est défini comme l'agrégation du travail de ses composants. Dans un système multi-couche, le travail est réalisé à chaque couche du système ; les couches sont séparées d'un point de vue organisationnel, mais chaque couche fournit des services à la couche qui est au-dessus d'elle. La figure 4.4 illustre un exemple d'un système multi-couches [76].

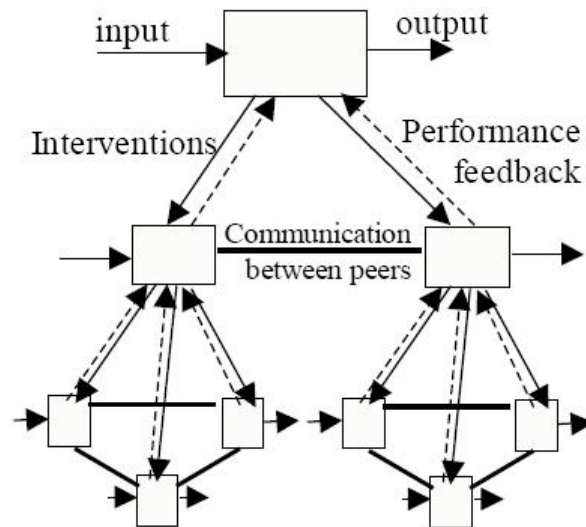


Figure 4.3 – Un système hiérarchique

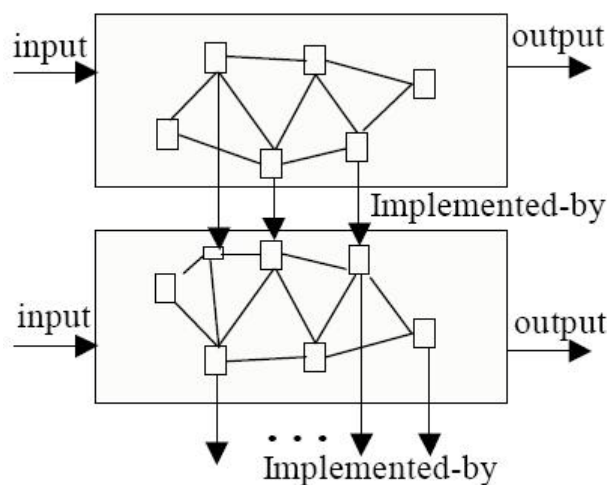


Figure 4.4 – Un système multi-couches

4.3 Une classification des langages de modélisation des processus

4.3.1 Les types de langages de modélisation

Les langages de modélisation d'entreprise proviennent de différentes communautés scientifiques et permettent d'atteindre plusieurs objectifs et de représenter différentes vues de l'entreprise. Cependant représenter un processus métier est utile pour trois raisons :

- 1) décrire un processus : nous modélisons un processus pour pouvoir le décrire et le documenter. Nous pouvons avoir plusieurs cibles pour cette description :
 - des humains : dans ce cas la compréhension est importante ;
 - des machines : dans ce cas le formalisme de modélisation est plus important.
- 2) analyser un processus : l'analyse d'un processus consiste en l'évaluation des propriétés d'un processus. L'amélioration des processus se base sur l'analyse des processus existants pour identifier les étapes redondantes ou non optimales. Si le processus est décrit en utilisant une méthode formelle, nous pouvons vérifier les propriétés structurelles des processus telles le couplage ou la cohésion ou encore les propriétés dynamiques telles l'absence de deadlock ou la viabilité du processus ;
- 3) exécuter des processus : à l'aide d'un outil capable aussi bien d'offrir une interface homme/machine pour l'exécution des tâches par des personnes que de s'intégrer et de faire appel aux applications du système d'information de l'entreprise, et ceci malgré leur hétérogénéité, leur disparité géographique et leur appartenance à des entités différentes.

Puisque les processus métier sont généralement complexes, les concepteurs des langages de modélisation fournissent différentes vues pour la modélisation, chacune se focalise sur un aspect du processus. Curtis identifie quatre vues [30] :

- 1) une vue fonctionnelle : cette vue représente les dépendances fonctionnelles entre les composants d'un processus (activités, sous-processus...). Ces dépendances résultent du fait que certains composants du processus consomment (ou nécessitent) des données (ou des ressources) produites par d'autres composants (ou processus). Les notations typiques utilisées dans la vue fonctionnelle incluent des diagrammes d'échange de données ;
- 2) une vue dynamique (comportementale) : la vue dynamique fournit l'ordonnancement et les paramètres d'exécution d'un processus. Cette vue permet de répondre à deux questions : quand certaines activités sont-elles exécutées (synchronisation, conditions préalables) et comment sont-elles exécutées (par exemple, en décrivant la logique d'exécution) ;
- 3) une vue informationnelle : cette vue inclut la description des éléments qui sont produits, consommés ou manipulés par le processus. Ces éléments incluent des données, des artefacts, des produits... ;
- 4) une vue organisationnelle : cette vue décrit qui exécute chaque tâche ou fonction, et où ça se passe dans l'organisation (fonctionnellement et physiquement).

4.3.2 Une vue d'ensemble des langages de modélisation des processus métier

Dans cette sous-section, nous étudions un certain nombre de langages de modélisation qui ont été développés pour décrire les processus métier avec des objectifs différents. Ces langages représentent différentes vues du processus métier (dynamique, fonctionnelle, informationnelle, organisationnelle) d'une façon plus ou moins formelle. Ces langages peuvent être classés comme suit :

- 1) *les langages traditionnels de modélisation* : Ces langages proviennent la plupart du temps des travaux de la mise en place des systèmes d'information et l'ingénierie des processus métier. Ces langages ne sont typiquement pas formels, mais peuvent se prêter à diverses analyses heuristiques ou informelles. Parmi ces langages, on trouve : IDEF, les réseaux de Petri, Event Process Chains (EPC), Role Activity Diagrams (RAD), Resource-Event-Agent (REA) et les tout récents Business Process Modeling Language (BPML) [23] et Business Process Modeling Notation (BPMN) [24] [84] ;

- 2) *les langages de modélisation des Workflow* : Un Workflow peut être défini comme "l'automatisation de processus métiers par échange de documents, informations et tâches entre acteurs pour action". Le workflow a pour objectif la coordination automatisée de tâches réalisées par des intervenants humains. Le moteur de workflow transfère des documents entre les participants aux processus en leur assignant des tâches (valider le document, effectuer une modification...). En général, un langage de modélisation des Workflow permet de décrire les opérations de façon à être supportées par un système de gestion de Workflow. Ces langages sont, pour la plupart, formels et exécutables. Nous pouvons citer le Workflow Process Description Language (WPDL) et les formats d'échange tels que Process Interchange Format (PIF) et Process Specification Language (PSL);
- 3) *les langages d'intégration de processus* : L'apparition des outils de Business to Business integration (B2Bi) a stimulé l'intérêt pour la modélisation des processus métier dans les buts d'intégrer les processus de deux partenaires ou plus. De tels langages se concentrent sur les mécanismes d'intégration en termes d'abstraction, des interfaces de programmation et des formats d'échange de données. Cette intégration doit être réalisée indépendamment de la technologie. Dans cette catégorie de langages, on peut citer ebXML et Business Process Execution Language for Web Services (BPEL4WS);
- 4) *les langages orientés objets* : En dépit que ces langages soient plus utilisés pour la modélisation des solutions logicielles, ils peuvent aussi être utilisés pour la modélisation d'entreprise grâce à un certain nombre de mécanismes. Le langage le plus utilisé est Unified Modeling Language (UML) qui permet de modéliser des processus métier grâce à un mécanisme d'extensibilité permettant de spécialiser les diagrammes généraux.

4.4 Les langages traditionnels de modélisation des processus

4.4.1 IDEF - ICAM DEFINITION Language

IDEF est une famille de méthodes pour modéliser et analyser l'entreprise qui a été sponsorisée par US Air Force dans le cadre de son programme de fabrication assistée par ordinateur (Integrated Computer-Aided Manufacturing - ICAM).

Le programme, lancé dans les années 70, a cherché à augmenter la productivité de la fabrication par l'application systématique de l'informatique. Le programme a identifié l'analyse de processus comme un outil important et le besoin de mise en place de meilleures techniques pour la description des processus.

4.4.1.1 IDEF0

IDEF0 est un standard constitué d'une carte de haut niveau des processus métier majeurs utilisés par une entreprise. La méthode SADT (Structured Analysis and Design Technique) a été la première à être rendue publique sous le vocable IDEF0 [70].

Selon IDEF0, une activité peut être vue comme une fonction qui transforme des objets d'entrée en objets de sortie (réalisation d'une tâche). Elle adopte une approche systémique en ce sens qu'elle considère que tout système complexe est une structure composée de systèmes plus simples en interaction. Ce modèle se caractérise par son excellence en termes d'appropriation par les utilisateurs, puisque sa notation graphique et sa syntaxe sont simples et naturelles. Une activité (boîte noire) consomme des entrées pour produire des sorties à partir de directives de contrôle (informations) en s'appuyant sur les potentialités des mécanismes (ressources) (voir la figure 4.5).

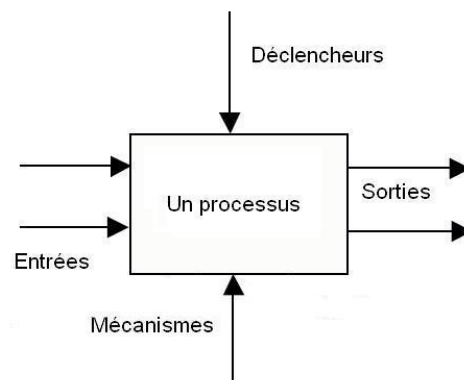


Figure 4.5 – Description d'un processus avec IDEF0

Un diagramme IDEF0 est donc un ensemble de fonctions connectées par des flux. Il existe plusieurs niveaux de détail associés à chaque activité. Il est effectivement possible de décomposer une activité en sous-activités. Ce principe de décomposition fonctionnelle permet de jouer facilement sur le niveau d'abstraction tout en garantissant une cohérence entre les différents degrés de finesse. Cette modélisation est descendante, modulaire, structurée et hiérarchisée. Enfin, le principe de la méthode limite son usage à un point de vue fonctionnel [70] [68].

4.4.1.2 IDEF1

Les concepts de base de ces modèles peuvent être explicités au travers de la sémantique employée. *"Une entité est un ensemble d'objets réels ou abstraits, ayant chacune une existence propre, partageant un ensemble de propriétés communes et présentant un intérêt pour l'entreprise"* [108]. Une relation est une association entre deux entités au moins. Ces principaux éléments (il en existe d'autres : attributs, clé, cardinalité...) permettent de définir graphiquement les besoins d'une part et de spécifier la conception de modèles de données d'autre part (informatique par exemple). Il s'agit donc d'une méthode particulièrement axée sur la vue informationnelle dans son acception conceptuelle et structurelle (même si le niveau "physique" est également intégré à la méthode) [100].

4.4.1.3 IDEF2

IDEF2 est un langage de modélisation du comportement d'un système basé sur le principe des files d'attente. C'est une méthode complémentaire à IDEF0/SADT qui vise à répondre aux lacunes du point de vue analyse des aspects dynamiques d'un système. Cette méthode est basée sur quatre modèles : système physique, flux des entités, contrôle du système et gestion des ressources. IDEF2 aborde donc de façon privilégiée les vues informationnelles et ressources sur des niveaux d'abstraction proches de l'exécution [71].

4.4.1.4 IDEF3

La méthode IDEF3 a été proposée en 1992 pour compenser les limites d'IDEF0 en matière de modélisation du comportement de l'entreprise. C'est une méthode qui se limite à la saisie et à la description des processus en utilisant une notation graphique. IDEF3 modélise un processus sous forme d'étapes, appelés unités de comportement, connectés par des boîtes de jonction et des liens. Une telle représentation forme un diagramme appelé description des flux de contrôle du processus. En d'autres termes, on retrouve une analyse macroscopique (niveau d'abstraction conceptuel) des vues fonctionnelle et organisationnelle. IDEF3 s'appuie pour ce faire sur la décomposition de l'IDEF0 en termes de typologie de système et de niveau d'avancement du projet. IDEF3 s'avère être une méthode intéressante pour la description de flux de processus. Néanmoins, ses composants n'offrent pas de description simple et formelle des conditions sur l'exécution d'un processus, toutes les informations additionnelles sont traitées sous forme de commentaires. IDEF3 ne permet pas, par exemple, de gérer les ressources et les flux de matières [71] [68].

De nouvelles méthodes ont rejoint la famille IDEF, les plus connus sont IDEF4 qui est une méthodologie orientée objet et IDEF5 qui est une méthodologie pour le développement des ontologies. La liste des méthodes IDEF s'étend de IDEF0 jusqu'à IDEF14 regroupant ainsi quinze méthodes différentes.

4.4.2 Les réseaux de Petri

Un réseau de Petri permet de représenter le comportement des systèmes dynamiques. Ce formalisme a été utilisé pour modéliser le comportement des systèmes concourants, asynchrones, distribués, parallèles, non déterministes et même stochastiques. La puissance des réseaux de Petri réside dans le fait qu'un nombre restreint de constructs (marque, place et transitions) peuvent représenter des comportements assez complexes d'une manière assez concise. En outre, les modèles réalisés en utilisant des réseaux de Petri se prêtent bien à l'analyse et à la validation formelle [22] [33].

4.4.3 La méthodologie GRAI

La méthodologie GRAI a été développée à l'origine par le groupe de recherche GRAI du Laboratoire d'Automatique et de Productique de l'Université Bordeaux 1 depuis le début des années 80, à partir :

- d'un ensemble de concepts de base rassemblés dans le modèle conceptuel de référence qui décrit la structure d'un système de production (ou d'une entreprise) et celle de son pilotage, de façon générique ;
- de formalismes graphiques de représentation qui décrivent les modèles ;
- d'une démarche générique qui permet de mettre en oeuvre les différentes méthodes.

La méthodologie GRAI permet de :

- rechercher les améliorations de performance (détection des points à améliorer et des points forts du système étudié, restructuration, re-engineering ...);
- établir les spécifications et le cahier des charges pour développer des applications informatiques, choisir des progiciels du marché ou bien servir de base à des développements logiciels ;
- concevoir et implanter un système d'indicateurs de performance ;
- réaliser des études de benchmarking de processus ou de benchmarking d'entreprise ;
- simuler le comportement des processus ;
- gérer les connaissances (acquisition, mémorisation, réutilisation) ;

- contribuer à une démarche qualité.

GRAI s'attache tout particulièrement à mettre en évidence le système décisionnel de l'entreprise. Elle définit un centre de décision comme étant le croisement d'une fonction et d'un horizon / période de décision. Les centres de décision sont positionnés au sein d'une grille GRAI (comme le montre la figure 4.6). Chaque cadre de décision précise les performances attendues de cette décision (les objectifs), les éléments sur lesquels on peut jouer (variables de décision), les limites de ces variables (contraintes) et une aide au choix parmi les actions possibles (critères) [36].

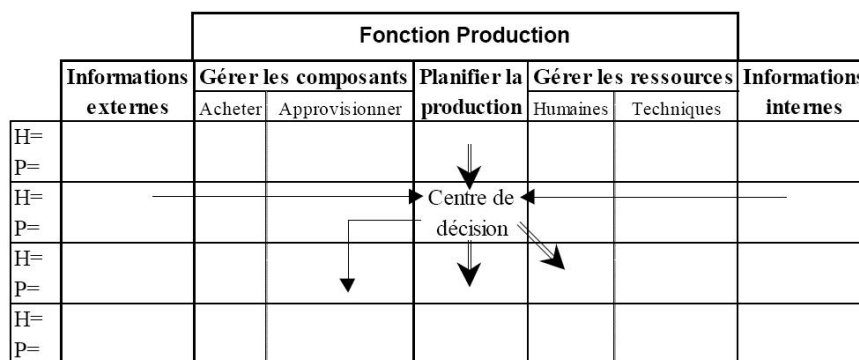


Figure 4.6 – Un exemple de grille GRAI

La grille GRAI permet de différencier les liaisons dites décisionnelles (double flèche, transmission d'une consigne) des liaisons dites informationnelles (simple flèche, transmission d'un flux d'information). Cette grille offre finalement, à travers une syntaxe simple, une confrontation entre un point de vue fonctionnel et informationnel (colonnes) et des niveaux de décision (lignes). Il existe deux types de grilles :

- la grille fonctionnelle dans laquelle les fonctions indiquées représentent les fonctions de l'entreprise ;
- la grille de conduite dans laquelle les fonctions indiquées représentent les fonctions élémentaires de conduite (planifier, gérer les produits ou les ressources).

Le fonctionnement de chaque centre de décision est détaillé par l'intermédiaire d'un réseau. La principale caractéristique de ce réseau réside dans la différenciation des activités d'exécution de celles de décision. Le formalisme du réseau GRAI insiste sur les éléments déclencheurs, supports et résultats qui caractérisent les activités d'exécution ou de décision [92] [68].

Au début des années 1990, la méthode GRAI a fait l'objet d'extensions pour donner naissance à la méthodologie GIM (GRAI Integrated Methodology) [104]. GIM propose un cadre de modélisation basé sur des niveaux d'abstraction similaires à la norme CEN ENV 40003 et de quatre vues :

- une vue informationnelle qui utilise les formalismes de MERISE et UML ;
- une vue physique et une vue fonctionnelle qui exploitent la méthodologie SADT/IDEF0 ;
- une vue décisionnelle qui utilise les outils originels de GRAI : la grille et le réseau.

4.4.4 Les autres langages traditionnels de modélisation des processus

Parmi les langages traditionnels de modélisation des processus métier [94], on peut citer aussi :

- **OLYMPIOS** : L'objectif du modèle OLYMPIOS est de construire le système d'information d'un exploitant donné (utilisateur, fournisseur, client. . .) de l'entreprise. OLYMPIOS permet de travailler sur plusieurs étapes du cycle de vie d'un système d'information : la spécification, la conception, l'implémentation, la mise en œuvre (et la validation) et la maintenance [21] ;
- **CIMOSA** : Contrairement aux méthodes présentées jusqu'ici, CIMOSA ne met pas l'accent sur la représentation graphique des activités et des processus, mais fournit un langage formel de description des processus opérationnels et de leurs activités. CIMOSA considère que toute entreprise se compose d'un :
 - grand ensemble de processus communicants chargés de réaliser les objectifs fixés ;
 - ensemble d'entités fonctionnelles réalisant les processus opérationnels en fonction de l'état du système.
- **Role Activity Diagrams (RAD)** : ce formalisme fait partie de STRIM (Systematic Technique for Role and Interaction Modeling) qui est une méthodologie développée par Praxis pour la modélisation et l'analyse des processus métier. Le but de ce formalisme est de développer des modèles qui soient "révélateurs et communicatifs" [86]. On identifie cinq concepts essentiels à RAD qui sont : rôles, acteurs, activités, interactions, et entités ;
- **Event Process Chains (EPC)** : ce formalisme est employé pour décrire les processus métier d'une manière informelle et est censé être manipulé par des utilisateurs métier plutôt qu'une manipulation formelle ;
- **Resource-Event-Agent (REA)** : dans sa version initiale, ce formalisme permettait de représenter les informations comptables pour résoudre les problèmes rencontrés par les procédures et les logiciels de comptabilité. Maintenant, REA est considéré comme un langage pour modéliser les processus inter-entreprises ;
- **The Architectural Modeling Box for Enterprise Redesign (AMBER)** : ce formalisme a été conçu dans le but d'avoir un langage compréhensible par les utilisateurs humains et qui soit aussi analysable par les outils appropriés.

Parmi les formalismes qui peuvent être considérés comme des langages traditionnels de modélisation des processus métier, il y a Business Process Modeling Language (BPML) (voir la section 5.3.1) et Business Process Modeling Notation (BPMN) (voir la section 5.3.2).

4.5 Les langages d'intégration de processus

4.5.1 Langage de modélisation de Workflow

Le Workflow Management Coalition (WfMC), qui est une coalition de centaines d'organismes comprenant des industriels, des utilisateurs, des fournisseurs de service et des organismes de recherche, essaie de développer l'utilisation des Workflow par le développement des standards [8]. Le WfMC a concentré ses travaux sur un format d'échange des descriptions des processus. Ceci est réalisé en combinant deux éléments :

- 1) un métamodèle extensible de description des processus ; la figure 4.7 montre les entités principales et les relations du métamodèle ;
- 2) un langage de description des processus du Workflow (Workflow Process Description Language - WPDL) pour la "sérialisation" des modèles de processus décrits dans la figure 4.7, ce langage est basé sur le XML.

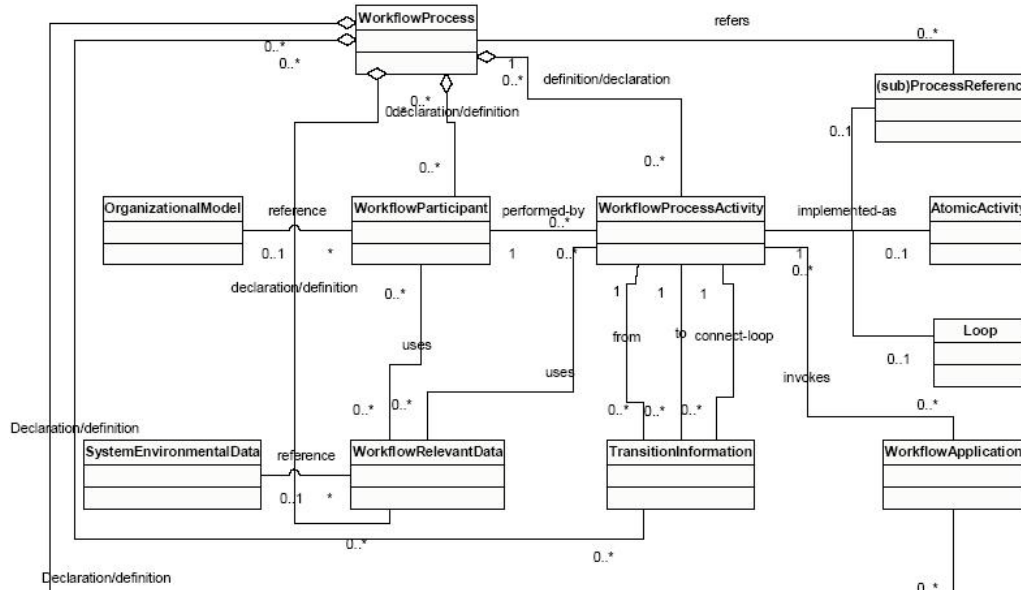


Figure 4.7 – Un métamodèle des processus réalisé par le WfMC

4.5.2 RosettaNet

RosettaNet est un consortium groupant plus de quatre cents compagnies (Directions informatiques, constructeurs de semi-conducteurs, télécommunication. . .) qui travaille pour "créer et implémenter à l'échelle industrielle des normes ouvertes de processus métier". La philosophie de RosettaNet postule que pour mettre en place des processus métier inter-entreprises, les compagnies doivent rendre compatibles les différents niveaux de l'infrastructure informatique. En conséquence, le consortium a cherché à normaliser ces niveaux. La figure 4.8 illustre le standard proposé. La partie gauche montre les différentes couches définies par RosettaNet et la partie droite les standards mis en place [31].

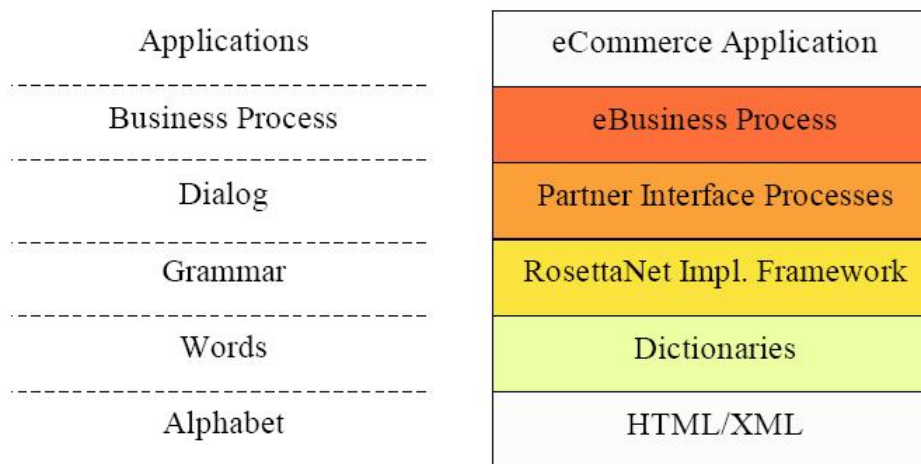


Figure 4.8 – Structure du standard RosettaNet

4.5.3 BPEL4WS

Ce standard (voir la section 5.3.3) va être traité plus en détail dans le chapitre 5.

4.6 Les langages orientés objets

Les langages orientés objets ont longtemps été considérés comme des langages de représentation d'information compréhensibles autant par les responsables métier que les responsables informatiques. Cependant, ces langages sont souvent plus utilisés pour la modélisation des applications et/ou des systèmes d'information plus que la modélisation des processus métier.

Le langage le plus connu et le plus utilisé est UML qui va être présenté succinctement dans ce qui suit.

4.6.1 Unified Modeling Language (UML)

UML propose des diagrammes spécialisés (dont les diagrammes d'activité, de séquence, de classes...) ayant chacun une fonction précise. Il n'existe pour le moment pas de diagramme UML spécialisé pour la modélisation des processus métiers. UML propose cependant un mécanisme d'extensibilité permettant de spécialiser chaque diagramme pour une utilisation particulière. Il est par exemple possible de spécialiser les diagrammes d'activité pour la modélisation des processus métiers [40].

4.7 Conclusion

Dans ce chapitre, nous avons présenté les principaux langages de modélisation des processus métier. Cependant, le plus difficile reste le choix de la méthode à utiliser. La figure 4.9 montre un comparatif entre les méthodes de modélisation d'entreprise présentées dans ce document en termes de couverture des différentes vues d'un processus métier (vue informationnelle, vue fonctionnelle, vue dynamique et vue organisationnelle). La figure 4.10 montre les champs d'utilisation (description, analyse et implémentation) des méthodes de modélisation d'entreprise présentées dans ce document. Ces deux figures peuvent constituer une aide pour le choix de la méthode à adopter pour un projet de modélisation d'entreprise [76].

Language	Informational view	Functional view	Dynamic view	Organizational view
IDEF0		Definitely		
IDEF1	Definitely			
IDEF3			Definitely	
Petri nets			Definitely	
RAD	Somewhat		Definitely	Definitely
EPC	References		Definitely	References
REA		Definitely	Definitely	Definitely
BPML	References		Definitely	Somewhat
WPDL	References			
RosettaNet	Definitely		Definitely	
ebXML	Definitely		Definitely	Somewhat
BPEL4WS	References		Definitely	Definitely
OORAM	Definitely	Somewhat	Definitely	Somewhat
EDOC	Definitely	Definitely	Definitely	Definitely
UML2	Definitely	Definitely	Definitely	Has the ingredients

Figure 4.9 – Comparaison des langages de modélisation en terme de couverture des différentes vues [76]

Language	Description		Analysis		Enactment	
	Human	Machine	Qualitative	Quantitative	Simulation	Execution
IDEF0	Yes		Yes			
IDEF1	Yes		Yes			
IDEF3	Yes		Yes			
Petri nets	Somewhat	Yes	Somewhat	Yes	Yes	Somewhat
RAD	Yes		Yes	Somewhat	Somewhat	
EPC	Yes	Yes	Yes	Possibly		
REA	Yes		Somewhat	Yes		
BPML	Yes				Yes	
WPDL		Yes			Yes	Yes
RosettaNet	Yes		Yes		Somewhat	
ebXML		Yes	Yes		Somewhat	
BPEL4WS		Yes		Yes	Yes	Yes
OORAM	Yes	Yes		Yes		
EDOC	Yes	Yes	Yes	Possibly	Yes	
UML2	Yes	Yes	Yes	Possibly	Yes	

Figure 4.10 – L'utilisation possible de chaque langage de modélisation [76]

Le Business Process Management

5.1 Introduction

Les premières théories de gestion d'entreprise se sont concentrées sur le fonctionnement de l'entreprise et le contrôle de ses départements efficacement mais en se focalisant sur chaque département à part. Avec l'apparition du BPM (Business Process Management), une révolution a eu lieu : au lieu de se concentrer sur chaque fonction de l'entreprise séparément, et de ce fait ne pas remettre en cause la structure globale des processus, les chercheurs ont commencé à préconiser qu'on devrait regarder les processus métier en entier de bout en bout, afin d'optimiser le processus en entier. Ce chapitre nous permet de présenter le BPM et la gestion des processus d'entreprise.

5.2 Le concept de Business Process Management (BPM)

Le terme BPM est apparu pour la première fois au milieu des années 1990 dans le contexte de la gestion des stratégies d'entreprise en se basant sur les processus métier. Depuis, plusieurs définitions ont été énoncées. Nous avons retenu celle qui semble être la plus complète et la plus pertinente. En effet, **Le Business Process Management (BPM)** peut être défini comme "Un support des processus métier en utilisant des méthodes, des techniques et des outils logiciels afin de concevoir, modéliser, commander et analyser les processus opérationnels impliquant des humains, des organisations, des applications et tout autres sources d'information" [52].

La deuxième partie de cette définition montre la nécessité de mettre en place un support pour

les processus opérationnels. Ainsi, le terme "Gestion des Processus Métier" (Business Process Management) inclut l'analyse, la définition et la redéfinition des processus de l'entreprise, l'allocation des ressources, la planification, la gestion des processus, la mesure de la qualité et de l'efficacité à l'aide d'indicateurs et l'optimisation des processus. En plus, l'optimisation des processus nécessite la collecte et l'analyse de mesures en temps réels (monitoring) et de mesures stratégiques (gestion des performances) et leur corrélation, en tant que base pour l'amélioration et l'innovation [73].

L'approche idéale de BPM n'est pas de forcer une organisation à se comporter d'une certaine manière formelle, mais plutôt de comprendre ce comportement par des concepts et des principes qu'offre le BPM. C'est un processus de découverte de la connaissance (Knowledge Discovery Process) qui nécessite un effort considérable.

5.3 Les standards et BPM

Les standards qui sont relatifs au BPM peuvent être classés, comme le montre le tableau 5.1, selon leurs fonctionnalités.

Analysis and Design	Execution	Monitoring and Analysis
<ul style="list-style-type: none"> • Process knowledge discovery and analysis • Graphical process modeling notation • Business process specification • Technical process specification • Metrics specification • Resource specification • Business transaction definition • Technical transaction definition. 	<ul style="list-style-type: none"> • Resource assignment • Scheduling • Executable process languages • Service/activity centric orchestration • Process-to-process interoperation • Federated/distributed process invocation and coordination • Transaction execution. 	<ul style="list-style-type: none"> • Process query languages • Event/metrics query languages • Analytical languages • Activity monitoring • Process monitoring • Event monitoring • Process audit trail schemas • Process audit trail query.

Figure 5.1 – Les catégories des fonctionnalités des standards relatifs au BPM.

Beaucoup de travail reste à faire au niveau de la standardisation des fonctionnalités relatives au BPM. En effet, beaucoup de fonctionnalités relatives au BPM n'ont pas encore de standard.

Dans ce qui suit, nous allons fournir un guide des standards de BPM, dont la plupart sont toujours en cours de développement ou à leur version initiale.

5.3.1 BPML - Business Process Modeling Language

C'est le premier standard définissant un langage de modélisation des processus métier exécutable. Il a été développé par le Business Process Management Initiative (BPMI.org). Ce standard fournit un modèle abstrait pour exprimer les processus métier et les entités supportées. BPML définit un modèle formel pour exprimer les processus abstraits et exécutables qui concernent tous les aspects des processus métier de l'entreprise, y compris les activités de complexité variable, les transactions, la gestion des données, la concurrence, la gestion des exceptions et la sémantique opérationnelle. BPML fournit également une grammaire sous forme de schéma XML pour permettre la persistance et l'échange de définitions à travers des systèmes et des outils de modélisation hétérogènes [23].

5.3.2 BPMN - Business Process Modeling Notation

Le BPMN fournit une notation graphique pour exprimer les processus métier en Business Process Diagram (BPD). Il a été initialement développé par le Business Process Management Initiative (BPMI.org) puis il a été intégré dans les standards de l'Object Management Group (OMG). L'objectif du BPMN est double :

- fournir une notation graphique complète permettant de représenter un processus métier en découplant les informations métiers des informations techniques ce qui fournit un cadre de travail commun aux utilisateurs métiers et techniques ;
- fournir un mapping complet vers les langages d'exécution. Ainsi, une fois les processus modélisés par les utilisateurs métiers, et les informations techniques renseignées pour rendre le processus exécutable, il est possible de générer automatiquement, et de manière standard, le processus BPEL à exécuter par le moteur de processus.

La figure 5.2 montre un exemple de processus BPMN.

La spécification finale du BPMN, en version 1.0, a été adoptée par l'OMG en Février 2006. Cette spécification est le gage d'interopérabilité des outils de modélisation de processus métiers [84] [29]. La spécification du BPMN, en version 2.0, est actuellement en cours de révision par l'OMG [85].

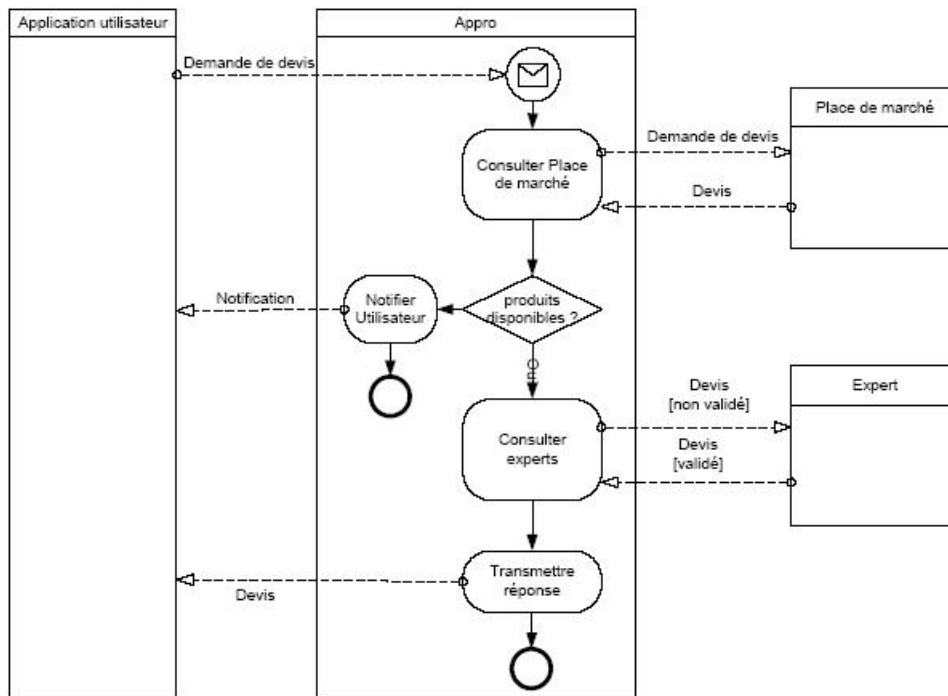


Figure 5.2 – Un exemple de processus décrit en BPMN

5.3.3 BPEL - Business Process Execution Language

Ce standard offre un langage d'exécution des processus métier distribués et transactionnels. Il supporte l'orchestration de multiples processus indépendants mais communiquants entre eux. En effet, BPEL est la représentation XML d'un processus exécutable, qui peut être déployée sur n'importe quel moteur de processus métier. L'élément atomique d'un processus BPEL est une "activité", qui peut être l'envoi d'un message, la réception d'un message, l'appel d'une opération ou une transformation de données. BPEL offre les fonctionnalités suivantes :

- organisation des activités : activités séquentielles, parallèles, switch conditionnel ...
- gestion des erreurs : il est possible de définir des gestionnaires d'erreurs pour des erreurs spécifiques, métier ou techniques. Des erreurs peuvent être déclenchées, traitées ou ignorées ;
- gestion des transactions : il existe deux modes de transactions : les transactions courtes, XA ou 2-phase commit, et les transactions longues, ou transactions compensées ;
- organisation du processus en sous processus.

Un processus BPEL (voir figure 5.3) définit, en XML, les activités réalisées dans le cadre de l'exécution du processus métier. Toutes les informations techniques nécessaires sont décrites.

Le nom exact de la spécification BPEL est BPEL4WS (Business Process Execution Language for Web Services). Ce nom est trompeur car un processus BPEL ne se limite pas à l'orchestration de Web Services : il est aussi possible d'inclure des connecteurs transactionnels comme des EJB ou des procédures stockées SQL [29].

```

<process>
  <partners/>           → Définition des partenaires (WebServices)
  <containers/>        → Définition des conteneurs de données
  <sequence>
    <receive/>         → Réception d'une requête
    <assign/>          → Transformation de données
    <invoke/>          → Appel de Web Service
    <assign/>          → Transformation de données
    <reply/>           → Envoi de la réponse
  </sequence>
</process>

```

Figure 5.3 – Un exemple de processus décrit en BPEL

5.3.4 BPQL - Business Process Query Language

Ce standard a été développé par le Business Process Management Initiative (BPMI.org). Il fournit un langage d'interrogation des processus métier. Le BPQL permet le déploiement des définitions des processus sur un serveur de processus et l'envoi de requête en temps-réel aux instances des processus [56].

5.3.5 BAML - Business Activity Monitoring Language

Ce standard supporte la définition des métriques des processus, le monitoring et le Key Performance Indicators (KPI). Le BAML permet de fournir un tableau de bord pour les processus afin de mesurer leurs performances [56].

5.3.6 BPATS - Business Process Audit Trail Schema

Cette norme fournit un schéma XML standard pour la sérialisation des instances des processus basé sur la définition des processus de BPEL4WS. Les 2 standards BPQL (5.3.4) et BAML (5.3.5) prennent en compte BPATS [56].

5.3.7 Les autres standards de BPM

Il existe un ensemble d'autres standards relatifs au BPM comme [56] :

- **WSCI - Web Service Choreography Interface** : définit des interfaces publiques des processus métier.
- **BTP - Business Transaction Protocol** : permet la coordination des requêtes, des réponses et les résultats pour les applications distribuées appelant des entités métier multiples.
- **ebXML BPSS - ebXML Business Process Specification Schema** : fournit un schéma pour des processus métier partiellement exécutable.
- **XPDL - XML Process Description Language** : fournit un langage de description des processus de Workflow basé sur les réseaux de Pétri.
- **PSL - Process Specification Language** : définit une représentation neuronale pour la construction des processus.

5.4 Les apports du BPM

Grâce au BPM, l'entreprise améliore ses processus sur 5 niveaux [32] [97] :

- l'agilité de l'entreprise à travers son système d'information : le BPM permet d'adapter rapidement les processus gérés par celui-ci à tout changement organisationnel, à une nouvelle offre produit, à un nouveau fournisseur. . .
- la rationalisation des processus : la mise en place d'une solution de BPM conduit l'entreprise à rationaliser ses processus afin de les modéliser et de les exécuter de la façon la plus automatisée possible dans le système d'information. En effet, l'outillage des processus d'entreprise, y compris pour les tâches "manuelles" réalisées par des utilisateurs, nécessite de mettre à plat les procédures et de les optimiser si besoin. Le BPM peut donc être utilisé pour accompagner le changement lorsqu'une entreprise modifie son organisation, son offre ou ses processus ;
- l'amélioration de l'efficacité des opérations : à travers l'automatisation de certaines tâches et le fait que seules les exceptions soient traitées par des interventions humaines, le BPM permet d'accélérer l'exécution des processus et d'accroître la capacité de traitement de l'entreprise et par conséquent sa qualité de service à travers la réduction des risques d'erreurs ;
- l'amélioration de la traçabilité et de la transparence dans l'exécution des processus : la mise en place d'indicateurs de suivi, transverses à l'organisation de l'entreprise et à l'ensemble de ses applications, va permettre à l'entreprise de mieux suivre l'avancement de ses processus, et surtout d'être en mesure de communiquer sur toute anomalie ou retard dans leur exécution. Par exemple, un retard de livraison d'un produit pourrait être expliqué aux clients, en précisant s'il s'agit d'un retard dû à une rupture de stock

d'un fournisseur, ou bien dû au retard de livraison du transporteur ;

- la capitalisation et la valorisation du système d'information de l'entreprise : en effet, le BPM va permettre de conserver l'existant et de l'utiliser au mieux pour répondre aux nouveaux besoins. Les solutions de BPM ne se substituent pas aux applications de l'entreprise, mais y ajoutent un niveau d'abstraction et des interfaces normalisées, ainsi qu'une capacité à orchestrer l'appel aux fonctions et données de l'entreprise, offrant ainsi un meilleur niveau de souplesse au système d'information et permettant de créer de nouveaux services en réutilisant l'existant.

5.5 Le BPMS : Business Process Management System

Un BPMS est une suite de logiciels intégrés conçue pour permettre la mise en place du BPM. L'architecture d'une solution BPMS est présentée dans la Figure 5.4 où les fonctionnalités principales sont en gras (définition de processus, exécution de processus, suivi de processus, analyse et création de rapport sur les processus). Font également partie d'une solution BPM la console d'administration, la couche de connexion et le dépôt de données. Certaines solutions proposent également des outils que l'on trouve dans des produits de EAI ou de B2Bi et prennent en charge l'interaction avec les utilisateurs au travers d'une interface propriétaire (comme ces fonctionnalités ne sont pas encore offertes en standard, elles ont été mises un peu en dehors de l'architecture principale) [65] [97].

Certaines solutions proposent tous les composants alors que d'autres se spécialisent dans une ou deux des fonctionnalités.

5.5.1 Les principales fonctionnalités d'un BPMS

5.5.1.1 Définition des processus

Un processus est une séquence de tâches. Une des promesses la plus importante des technologies de BPM est la possibilité pour un analyste métier de définir les processus métier sans aucune compétence de programmation. Les analystes métier conçoivent les processus par l'intermédiaire d'une interface facile d'utilisation qui apporte une grande assistance dans la définition de processus métier complexes. Une fois défini, le processus est déployé sur le système d'information. Dans de nombreux cas, l'utilisation d'objets métier, présents dans une bibliothèque de l'entreprise et maintenu par le département IT, permet la définition de processus directement fonctionnels sans devoir passer par une autre personne.

Actuellement, les outils existant proposent une interface graphique avec des technologies de

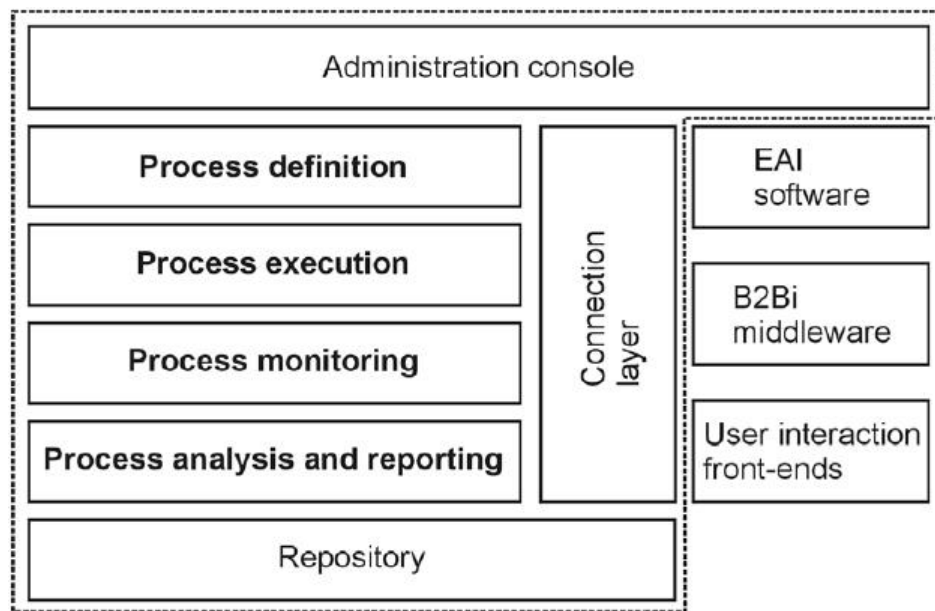


Figure 5.4 – Architecture d'une solution BPMS

tirer-déposer (drag-and-drop) qui permettent un design intuitif des processus métier. Un module performant de création (ou modification) des processus doit supporter toutes les parties d'un processus (informations et personnes), les sous-processus, les processus parallèles, la création de règles et la gestion des exceptions.

5.5.1.2 Exécution des processus

Un moteur permet le déroulement des tâches définies dans les processus. L'exécution est programmée ou déclenchée par des événements. Comme exemples de tâches on peut citer le transfert de fichier, le lancement d'un script, la lecture de message dans une file de messages, la transformation du type de données, une demande de confirmation d'un employé. Les tâches sont basées sur des transactions qui mettent en jeu des applications ou la livraison asynchrone à un utilisateur ou une application externe.

Plusieurs possibilités sont offertes par les outils BPMS d'aujourd'hui :

- possibilité de modifier un processus lorsqu'il est en cours d'exécution, ce qui permet à un utilisateur de rapidement effectuer des changements dans un processus lorsqu'il se produit un problème, sans avoir à arrêter le processus et le recommencer au début après modification ;
- possibilité d'équilibrer la charge de travail, que ce soit pour l'infrastructure informatique

avec la gestion de clusters de machines et des environnements distribués mais également et surtout la gestion des employés et des rôles qu'ils représentent en tenant compte de leur disponibilités (horaires, vacances . . .) ;

- possibilité de gérer les versions différentes des processus métier et donc la possibilité de pouvoir revenir en arrière sur l'évolution qu'a suivi un processus donné (pour des raisons légales ou lors d'optimisation).

5.5.1.3 Surveillance des processus

La surveillance des processus est fondamentale. Un des buts majeurs du BPM est de permettre un contrôle permanent et une amélioration constante des processus. Les processus sont suivis en temps réel. Lorsqu'une erreur ou une exception se produit, une alerte est envoyée par l'outil de surveillance aux utilisateurs concernés, ce qui permet une meilleure communication et une amélioration facilitée des processus.

5.5.1.4 Analyse et création de rapport sur les processus

Les analystes métier ont besoin d'indicateurs complets sur les différentes instances des processus qu'ils suivent (en cours ou terminés). Cela leur permet de comparer les activités basées sur les processus par rapport aux attentes. Basée sur de solides analyses, l'amélioration continue des processus est rendue possible.

5.5.1.5 Autres composants

Un système de BPM doit également gérer les éléments suivants :

Console d'administration : Une console d'administration qui permet aux administrateurs de gérer la partie technique du système. La console rend possible la gestion des connecteurs avec les autres applications (internes ou externes) et l'intégration avec l'infrastructure de sécurité de l'entreprise ;

Outils d'intégration : Une couche de connexion qui permet l'intégration des applications de l'entreprise (ERP, CRM, bases de données . . .) et des partenaires (B2Bi) dans les processus métier. Les connexions sont rendues possibles par l'adoption de protocoles et de langages standards comme le HTTP, le SMTP et le FTP. Des connecteurs spécifiques et des adaptateurs sont utilisés pour certaines applications ou les services Web. Pour connecter les applications informatiques, les fournisseurs de BPMS utilisent également les APIs ou des services de messagerie ;

Interface utilisateur : Pour connecter les gens (et leurs connaissances), la plupart des BPMS font appel à des interfaces utilisateurs au travers du navigateur Internet pour présenter un espace de travail qui ressemble à un outil de gestion des e-mails où sont rapportés les processus auxquels ils doivent participer suivant les différents rôles que ces personnes représentent.

L'interface utilisateur est une des composantes les plus critiques d'un système informatique dans l'acceptation du système par l'utilisateur. Cette partie qui touche l'utilisateur final est responsable de l'impression que l'utilisateur se fera sur la valeur du système. Actuellement, les utilisateurs se trouvent confrontés à de multiples interfaces pour les diverses applications de l'entreprise comme un ERP et un CRM. Une solution adéquate de BPMS permet d'unifier en une seule interface utilisateur tout ce dont il a besoin, au regard des rôles qu'il joue. De plus, la simplification de l'interface pour les utilisateurs réduit considérablement les coûts de formation ;

Dépôt de données : Un dépôt électronique qui contient la définition des processus métier et les états dans lesquels se trouvent toutes les instances existantes. Tous les composants d'un système de BPM se branchent sur ce dépôt lors de la conception, la modification, l'exécution et l'analyse des processus.

5.5.2 Les principaux BPMS du marché

Le portail dédié au management par les processus [20] constitue une bonne source pour le choix et la comparaison des BPMS. Dans ce qui suit, nous présentons les principaux BPMS du marché.

5.5.2.1 ARIS Platform

ARIS Platform est la partie modélisation de la suite d'outils ARIS : elle autorise suivant les besoins, le descriptif et la modélisation de processus, d'organisation de systèmes d'information ou de connaissances au sein d'un référentiel unique [57].

Elle permet de modéliser l'ensemble des éléments d'une démarche processus :

- l'organisation de l'entreprise : processus ;
- les ressources humaines : organigramme, cartographie des connaissances ;
- les moyens techniques ;
- le système d'information (urbanisation) ;
- les risques opérationnels.

La suite d'ARIS intègre entre autres :

- ARIS Quality Management Scout : QMS est un outil méthodologique expliquant par le détail le déroulement d'un projet Qualité ISO 9000 :2000 autour de la suite ARIS. Le package se compose :
 - d'un outil méthodologique conséquent (200 pages environ) déroulant le film d'un projet Qualité étape par étape ;
 - de documents supports au format Word (questionnaire d'auto évaluation, matrice d'évaluation, plans de réunion . . .) ;
 - de scripts s'incorporant à ARIS et permettant de générer un manuel Qualité au format HTML ou Word.
- ARIS Balanced Score Card : BSC permet de définir des tableaux de bord prospectifs, en déclinant les objectifs stratégiques en initiatives à mener et en attachant chacun des objectifs à des indicateurs ;
- Aris Process Performance Manager : Cet outil permet de contrôler de manière continue la performance des processus clés de l'entreprise ;
- Aris Simul : Cet outil s'appuie sur les processus modélisés par Aris Toolset pour en simuler la performance ;
- ARIS Web Publisher : En complément d'Aris Toolset, ce module permet de générer au format HTML la documentation de l'ensemble des processus (graphes et attributs). C'est un accès au référentiel par la cartographie, par nom ou type de modèle.

5.5.2.2 Casewise Corporate Modeler

Le descriptif de cet outil [25] :

- outil de modélisation des processus métier et des systèmes d'information ;
- gestion d'un référentiel central des processus ;
- les différents prismes d'un processus se traduisent par 5 types de diagramme : processus, hiérarchie (organisation), données, entités (systèmes entités/rerelations) et génériques (flowcharts personnalisés) ;
- fonctions de simulation permettant de comparer scénarios actuels et alternatifs ;
- export des rapports au format Excel ;
- publication de rapports, manuels et référentiels au format Word, HTML dont le contenu peut être automatiquement adapté au profil des destinataires ;
- héritage des propriétés du diagramme parent par le diagramme enfant ;
- vue matricielle en saisie et restitution ;
- des modules complémentaires BSC, IT ainsi que la fourniture d'un référentiel ITIL sont disponibles.

La suite Casewise comporte aussi :

- Casewise Best Practice and Compliance Accelerator : qui permet :
 - la visualisation graphique de la documentation ITIL via des diagrammes ;
 - la mise en œuvre des indicateurs de mesures ITIL pour évaluer le retour sur investissement ;
 - la gestion des projets ITIL en sous projet par un module "roadmap" et de les publier sur un intranet ;
 - les fonctionnalités de reporting avec accord de niveau de service (SLA), des catalogues de services... ;
 - la possibilité de publications des processus ITIL sur un intranet.
- Casewise Simulator : c'est un moteur de simulation incorporé à Corporate Modeler, qui permet :
 - la visualisation des résultats sous forme de tables ou de graphes (possibilité de visualisation dynamique pendant la simulation) ;
 - la gestion de scénarios ("what if") via le module Analyseur : recherche automatique du rapport ressources / coût optimal par exemple.

5.5.2.3 MEGA Process

MEGA Process couvre les besoins de modélisation des processus, des organisations à partir de 3 cartes principales : les acteurs, les activités et les processus.

La description du processus (sous forme de logigramme commenté) met en évidence les relations du processus concerné avec les acteurs, les activités, les procédures et les moyens informatiques de l'entreprise [74].

La suite MEGA comporte aussi :

- MEGA Designer : qui couvre au sein d'un environnement unique toutes les dimensions nécessaires à la conception des architectures informatiques modernes : architecture de services et de processus automatisés, architecture des données, architecture de composants objets, en utilisant le langage de modélisation UML ;
- MEGA Business Data : Les principales fonctionnalités de MEGA Business Data sont :
 - spécification des entités métiers ;
 - spécification des associations ;
 - définitions de contraintes.
- MEGA Documentation : Module complémentaire à MEGA Process permettant de personnaliser les restitutions produites, les représentations des modèles ainsi que d'analyser les données du référentiel ;

- MEGA Simulation : Moteur de simulation intégré à MEGA Process. Les fonctionnalités sont regroupées en 3 modes : débutant, standard et expert. La simulation s'appuie sur le modèle créé dans MEGA Process.

5.5.2.4 ADONIS

Adonis permet :

- la modélisation des processus selon une palette d'objets prédéfinie et personnalisable, au niveau graphique comme au niveau données (attributs) ;
- la possibilité de définir des rapports sur les modèles, en mode tableau ou graphique et de les exporter. Il permet aussi la publication des modèles au format HTML, RTF ou XML [13].

La suite ADONIS comporte aussi ADOit : Module dédié à la modélisation des systèmes d'informations, ADOit est compatible avec la méthode ITIL. Modélisation de l'architecture applicative, de l'infrastructure technique et des processus informatiques.

5.5.2.5 Microsoft Biztalk server

Serveur d'intégration s'appuyant sur l'architecture .Net et notamment Visual Studio.Net, intégré à l'outil, pour la programmation et Visio pour la description des processus.

La gestion des processus métiers repose sur un moteur d'orchestration et d'échange de messages évolutifs supportant le standard BPEL [75].

5.5.2.6 Intalio

Intalio est un système BPMS élaboré pour répondre à deux impératifs majeurs : optimiser les actifs existants de l'entreprise et couvrir l'ensemble du cycle de vie des processus. Intalio est conçu pour satisfaire à ce double objectif en préfigurant une architecture orientée processus [59] [29].

À cette fin, Intalio est architecturé autour de quatre composants :

- Intalio Server : Composant essentiel du système, en charge de l'exécution des processus métiers. Trois composants additionnels s'articulent autour de celui-ci pour assurer les liaisons nécessaires avec les éléments d'actif existants que sont les processus, le personnel et les systèmes ;
- Intalio Designer : Editeur permettant d'une part d'importer les processus métiers, les procédures et les stratégies conçues en utilisant les outils de modélisation du marché ; et

d'autre part de réaliser les extensions nécessaires pour rendre ces processus exécutables et gérables. Les processus capitalisent sur les applications du SI de l'entreprise, et en offrent une vision métier ;

- Intalio Director : Composant permettant aux employés, clients et partenaires de piloter l'exécution des processus métiers par l'intermédiaire d'interfaces utilisateurs fournis par les systèmes de travail collaboratif, de workflow, ou les portails d'information d'entreprise ;
- Intalio Projectors : Composant permettant d'exposer toute application, base de données ou système propriétaire du SI sous la forme d'un processus réutilisable. Les utilisateurs métiers peuvent alors avoir une vue complète du système d'information sous la forme de composition de processus réutilisables.

5.5.2.7 GraiTools

GraiTools propose un environnement complet de BPM, de modélisation d'entreprise et de pilotage de projet en s'appuyant sur des techniques et des concepts éprouvés (modèle GRAI et Gestion de projet). L'ensemble de ses outils sont intégrés dans une seule et même plateforme, facilitant la mise en œuvre des projets et la maintenance des référentiels d'entreprise ; ils accompagnent ainsi l'effort d'adaptation permanent à l'évolution des marchés.

5.6 Conclusion

Nous avons présenté dans ce chapitre le Business Process Management (BPM) qui constitue le standard de gestion des processus métier d'entreprise. Ce standard constitue une façon d'aborder les processus d'entreprise afin de les améliorer. À travers ce chapitre, nous avons présenté la démarche, les principes et les apports de BPM ainsi que l'ensemble des standards et outils relatifs au BPM.

La notion de qualité en entreprise

6.1 Introduction

Les chapitres 3, 4 et 5 nous ont permis de montrer comment aborder les processus d'entreprise pour les comprendre et les modéliser. Ce chapitre nous permet de présenter les bonnes pratiques et recommandations desquelles nous allons nous inspirer pour organiser les processus. Ces bonnes pratiques sont très souvent organisées dans des normes et standards qualité. Ce chapitre présente donc la notion de qualité en entreprise et un survol des principaux standards et normes qualité.

6.2 la Qualité

La "**Qualité**" peut se définir comme la capacité à atteindre les objectifs opérationnels visés. La norme ISO 9001v2000 [6] la définit comme ceci :

"Aptitude d'un ensemble de caractéristiques intrinsèques à satisfaire des exigences."

Dans la pratique [16], la qualité se décline sous deux formes :

La qualité externe, correspondant à la satisfaction des clients. Il s'agit de fournir un produit ou des services conformes aux attentes des clients afin de les fidéliser et ainsi améliorer la part de marché. Les bénéficiaires de la qualité externe sont les clients d'une entreprise et ses partenaires extérieurs. Ce type de démarche passe ainsi par une néces-

saire écoute des clients mais doit permettre également de prendre en compte des besoins implicites, non exprimés par les bénéficiaires ;

La qualité interne, correspondant à l'amélioration du fonctionnement interne de l'entreprise. L'objet de la qualité interne est de mettre en œuvre des moyens permettant de décrire au mieux l'organisation, de repérer et de limiter les dysfonctionnements. Les bénéficiaires de la qualité interne sont la direction et les personnels de l'entreprise. La qualité interne passe généralement par une étape d'identification et de formalisation des processus internes réalisés grâce à une démarche participative.

L'objet de la qualité est donc de fournir une offre adaptée aux clients, avec des processus maîtrisés tout en s'assurant que l'amélioration ne se traduit pas par un surcoût général, auquel cas on parle de "**sur-qualité**". Il est possible d'améliorer un grand nombre de dysfonctionnements à moindre coût, mais, à l'inverse, plus on souhaite approcher la perfection plus les coûts grimpent !

L'opposé de la qualité, nommé "**non-qualité**", possède également un coût. En effet, il s'avère généralement plus coûteux de corriger les défauts ou les erreurs que de "faire bien" dès le départ. D'autre part, le coût de la non-qualité est d'autant plus important qu'elle est détectée tardivement. A titre d'illustration, réaliser à nouveau un produit défectueux coûtera au final plus du double du prix de production du produit initial s'il avait été réalisé correctement. Qui plus est, la différence de prix sera moins grande si le défaut est détecté en cours de production que s'il est détecté par le client final (insatisfaction du client, traitement de l'incident, suivi du client, frais de port...).

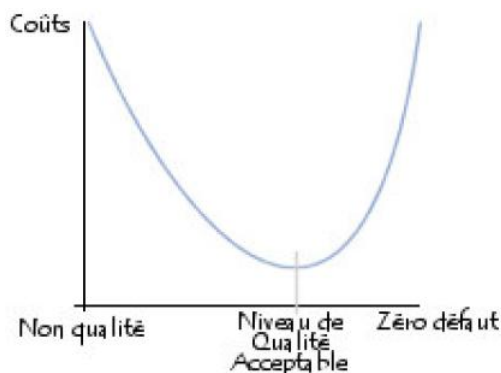


Figure 6.1 – La relation entre le coût et la qualité

L'amélioration de la qualité, interne et externe, permet à l'entreprise de travailler dans de meilleures conditions avec ses bénéficiaires, ce qui se traduit par une relation de confiance et des gains sur le plan financier (augmentation des bénéfices...) ou humain (clarification des rôles, des besoins et de l'offre, motivation du personnel...).

Il s'agit toutefois d'un effort impliquant l'ensemble de l'entreprise et conduisant la plupart du temps à des modifications des habitudes de travail, voire des changements organisationnels. Ainsi appelle-t-on "**démarche qualité**" l'approche organisationnelle permettant un progrès permanent dans la résolution des non-qualités. Il s'agit d'une démarche participative, c'est-à-dire à laquelle doit nécessairement participer l'ensemble de l'entreprise et par conséquent nécessairement portée au plus haut niveau hiérarchique.

Nous pouvons distinguer une autre typologie de la qualité :

La qualité produit liée aux caractéristiques même du produit.

La qualité des processus liée aux méthodes, outils et moyens utilisés pour réaliser le produit.

C'est à travers cette typologie que nous décrirons les principales normes et modèles pouvant être utilisés dans le monde de l'informatique, avec une focalisation sur les deux normes qui nous intéressent (ISO 9001v2001 et CMMI).

6.3 La Qualité Produit

6.3.1 La norme ISO 9126, Technologies de l'Information : "Qualité des produits logiciels"

La norme ISO 9126 est un dérivé du modèle de Mc Call créé en 1977 au cours d'une étude pour l'US AIR FORCE [27]. Mc Call définit une approche de la qualité à partir de la définition de caractéristiques :

- externes : facteurs de qualité ;
- internes : critères de qualité ;
- mesurables : métriques.

Il a donc défini 23 critères sur 11 facteurs. Chaque critère correspond à au moins une métrique. Le modèle de Mc Call différencie utilisateur et réalisateur.

Comme le montre la figure 6.2, La norme ISO 9126, "Technologies de l'Information : Qualité des produits logiciels", définit et décrit une série de caractéristiques qualité d'un produit logiciel (caractéristiques internes et externes, caractéristiques à l'utilisation) qui peuvent être utilisées pour spécifier les exigences fonctionnelles et non fonctionnelles des clients et des utilisateurs. Chaque caractéristique est détaillée en sous-caractéristiques, et pour chacune d'elles, la norme propose une série de mesures à mettre en place pour évaluer la conformité du produit développé par rapport aux exigences formulées.



Figure 6.2 – Les caractéristiques du produit logiciel selon ISO 9126

Fonctionnalité : Ensemble d'attributs portant sur l'existence d'un ensemble de fonctions et leurs propriétés données. Les fonctions sont celles qui satisfont aux besoins exprimés ou implicites ;

Fiabilité : Ensemble d'attributs portant sur l'aptitude du logiciel à maintenir son niveau de service dans des conditions précises et pendant une période déterminée ;

Utilisabilité : Ensemble d'attributs portant sur l'effort nécessaire pour l'utilisation et sur l'évaluation individuelle de cette utilisation par un ensemble défini ou implicite d'utilisateurs ;

Efficacité : Ensemble d'attributs portant sur le rapport existant entre le niveau de service d'un logiciel et la quantité de ressources utilisées, dans des conditions déterminées ;

Maintenabilité : Ensemble d'attributs portant sur l'effort nécessaire pour faire des modifications données ;

Portabilité : Ensemble d'attributs portant sur l'aptitude de logiciel à être transféré d'un environnement à l'autre.

Ce modèle indique un lien de causalité entre des caractéristiques internes et des caractéristiques externes d'un produit, mais ne permet toutefois pas de les prédire, ni de connaître les « bonnes » et « mauvaises » plages de valeurs.

6.4 Le management de la qualité par les processus

6.4.1 Qu'est ce qu'un processus ?

La version 2000 de la norme ISO 9001 incite les entreprises à une "approche processus" dans leur démarche qualité. Dans ce contexte, **un processus** constitue l'ensemble des moyens, d'activités et de pratiques, s'appuyant sur des outils, des équipements et des méthodes, mis en œuvre depuis l'expression d'un besoin client jusqu'à la satisfaction de ce besoin (comme le montre la figure 6.3) [32].

PROCESSUS : ensemble de moyens et d'activités qui transforment des "entrants" en "sortants".

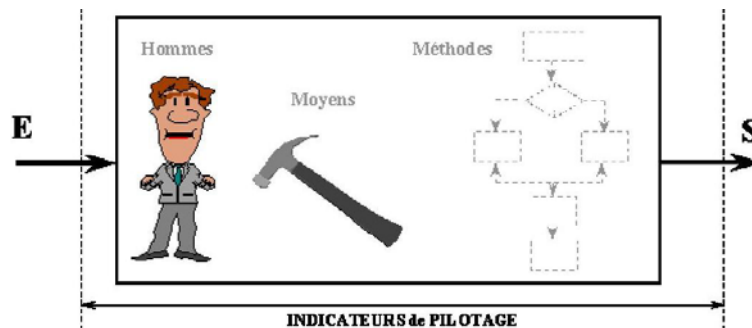


Figure 6.3 – Un processus

"L'approche processus", comme le définit l'AFIS (Association Française de l'Ingénierie Système), désigne l'application d'un système de processus au sein d'un organisme, ainsi que l'identification, les interactions et le management de ces processus (comme le montre la figure 6.4).

6.4.2 L'amélioration des processus

La performance des produits dépend :

- des hommes qui les développent ;
- des technologies utilisées ;
- et aussi de la qualité des processus de développement et de maintenance.

La course au plus vite, moins cher et mieux ne va pas sans risque. Dans un contexte de développement et de production très tendu, la moindre erreur a de lourdes conséquences



Figure 6.4 – L'approche processus



Figure 6.5 – Les 3 axes d'amélioration des processus

sur la satisfaction du client, la rentabilité de l'entreprise, ses parts de marché et sa capacité d'investissement.

Relever ce défi, passe obligatoirement par **la maîtrise et l'amélioration du processus de développement**.

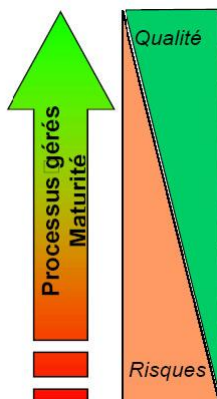


Figure 6.6 – La gestion de la maturité des processus

Les objectifs de la gestion et l'amélioration des processus sont :

- comprendre comment les produits sont réellement développés ;
- prédire et contrôler la qualité des produits ;
- prendre des décisions justifiées concernant les coûts, le retour sur investissement (méthodologies, technologie, processus) ;
- contrôler les coûts de développement.

Elle couvre les trois axes :

- 1) **les Hommes** : par la formation et la motivation des équipes par la participation au projet d'amélioration,
- 2) **les processus** : par leur définition et la mesure de leur efficacité et leur amélioration,
- 3) **les technologies** : par la mise en œuvre des technologies modernes en support des processus.

Il existe aujourd'hui de nombreux modèles et normes traitant de l'amélioration des processus. Certains sont non spécifiques à l'informatique tels que :

- ISO 9001
- Six Sigma

D'autres sont dédiés aux métiers de l'informatique :

- COBIT
- ITIL

- CMMI

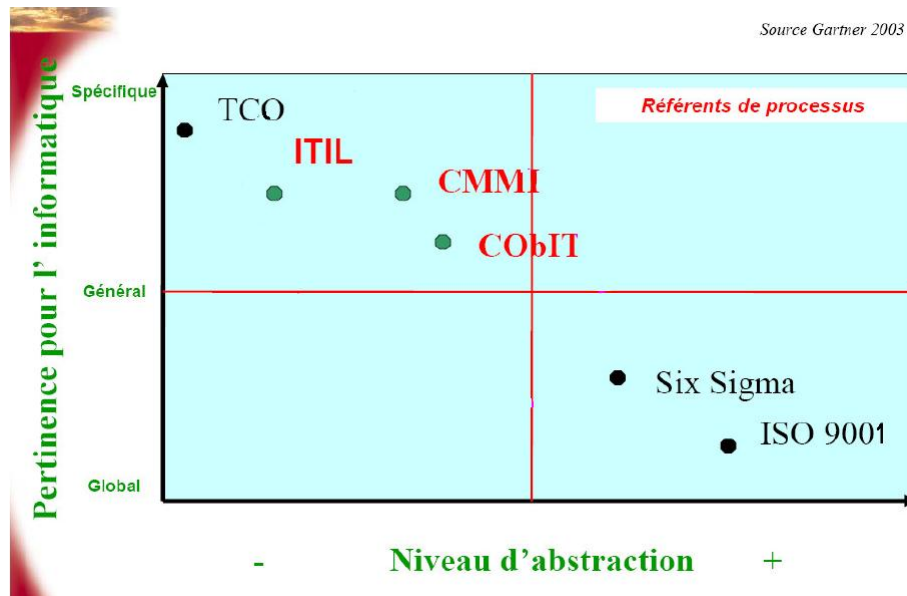


Figure 6.7 – Les modèles et normes traitant l'amélioration des processus

Il existe aussi de nombreux corpus de connaissances fortement complémentaire à l'utilisation des modèles et normes. Ils font ainsi le lien entre les activités des opérationnels et la description de ces activités faite par la qualité et la direction. Les corpus les plus connus et utilisés sont :

- PMBoK du PMI concernant la gestion de projet
- SWEBoK concernant le génie logiciel

6.4.3 Pourquoi une approche par les processus ?

Le management par les processus est adopté dans le but principal de satisfaire les clients. Cela consiste à adopter une vision transversale de l'entreprise, par un alignement coordonné et un pilotage des différentes activités créatrices de valeur pour le client, ce dernier étant de plus en plus exigeant.

Pour cela, ce management décloisonne les organisations traditionnelles, évitant ainsi l'effet "guichet" où le client est perdu de vue, offrant alors plus de flexibilité à des organisations jusque là bureaucratiques. L'entreprise utilise alors ses ressources de manière optimale, responsabilisant les individus à tous les niveaux. Chaque acteur a ainsi une meilleure perception de la valeur ajoutée qu'il peut apporter au client et au fonctionnement de l'entreprise, ayant pour effet de l'impliquer et de le motiver encore plus dans son travail.

L'utilisation de modèles ou de normes accroît ainsi la productivité, réactivité et flexibilité des entreprises ayant adopté ces démarches.

Une des techniques les plus utilisées est le Business Process Management (BPM) qui peut être défini comme "Un support des processus métier en utilisant des méthodes, des techniques et des outils logiciels afin de concevoir, modéliser, commander et analyser les processus opérationnels impliquant des humains, des organisations, des applications et toutes autres sources d'information"[52].

6.4.4 Six Sigma

6.4.4.1 Présentation de Six Sigma

Six sigma est à l'origine une démarche qualité née au coeur même de Motorola en 1979. Limitée dans un premier temps aux techniques de SPC (Statistical Process Control / MSP Maitrise Statistique des Procédés), elle est rapidement devenue une véritable méthode de management englobant l'ensemble des fonctions de l'entreprise. Six sigma a ensuite été perfectionnée par d'autres groupes, comme General Electric, l'ayant mise en oeuvre avec succès. Six sigma est une méthode de management du progrès particulièrement efficace. Issue d'une démarche fortement connotée qualité à l'origine, Six sigma est relativement simple sur le plan du principe. Elle est fondée sur une règle : pour satisfaire les clients, il faut délivrer des produits de qualité [87].

La méthode Six Sigma offre techniques et outils pour améliorer drastiquement la capacité de production des processus tout en réduisant les défauts. Orientée processus de production à l'origine, la méthode recherche la régularité absolue. La variabilité est en effet source d'insatisfaction du client.

Le client attend un produit avec une certaine qualité, selon un standard précis. Ne pas être capable de garantir la totalité de la production en respectant ce standard est particulièrement coûteux pour l'entreprise.

En fait la variabilité de la qualité finale est essentiellement la conséquence de l'instabilité des composants entrant dans la fabrication du produit, de l'imprécision des procédures de travail et plus globalement de la complexité des processus. 6 sigma impose de rester dans les limites en appliquant le principe : "if you can measure how many defects you have in a process, you can systematically figure out how to eliminate them and get as close to zero defects as possible." Autrement dit, en résumé : "Si on peut mesurer on peut corriger".

L'objectif est de recentrer la courbe sur la cible. À terme, Six Sigma n'autorise pas plus de 3.4 défauts par million pour chaque produit ou service (comme le montre la figure 6.8).

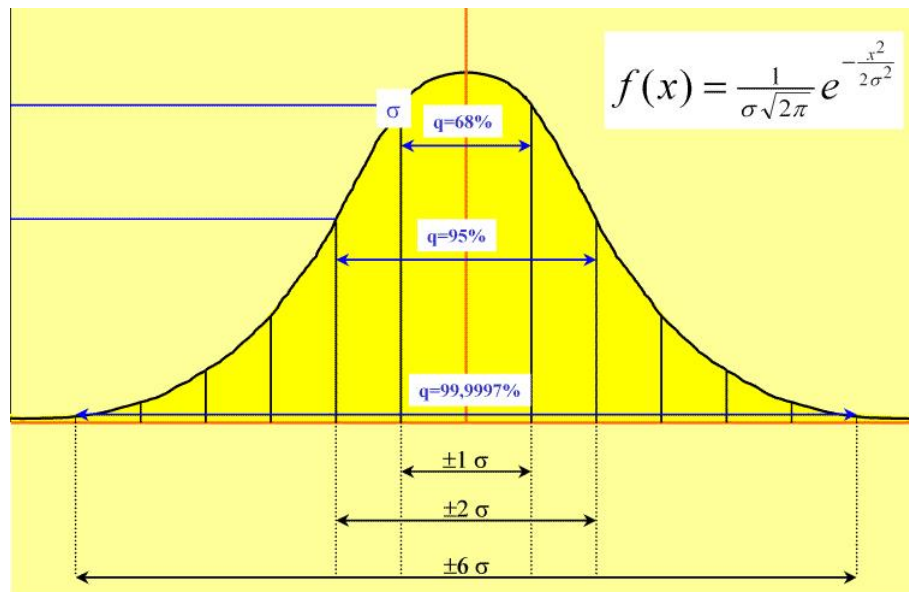


Figure 6.8 – La courbe de Six Sigma

6.4.4.2 Les apports de Six Sigma

Economie de coûts :

- réduction de la non-qualité comme les rebuts, les reprises, les retouches, les retours clients ainsi que tous les problèmes inhérents à la non-qualité (perte de temps, problèmes de communication, blocage aux interfaces des processus et activités...);
- meilleure exploitation des ressources et optimisation des processus.

Amélioration des revenus :

- meilleure satisfaction des clients donc fidélisation renforcée, amélioration du CA par client, accroissement de la part de marché.

Dynamique de progrès continu :

- meilleure disposition pour lancer des projets de grande ampleur : nouveaux produits ou nouveaux processus ;
- instauration d'une culture du pilotage par la mesure.

6.4.4.3 Six Sigma, la culture de la mesure au service de l'amélioration continue

Six Sigma n'est pas une amélioration ponctuelle. C'est une réforme de fond de l'ensemble de l'organisation et des mentalités. Il faut ainsi soigner la formation, l'accompagnement du changement et la mesure continue de la performance. L'effort doit être continuellement soutenu [87].

La réussite du projet repose sur des paramètres bien précis :

- une approche par projet, chacun bien délimité en termes de périmètre d'action, de délais et de budget ;
- des objectifs clairs, concrets, précis et connus ;
- une formation extensive de l'ensemble des acteurs directs et indirects du projet ;
- une coopération et une participation de tous les instants ;
- un système de mesure performant.

La généralisation de la mesure est à la base de la méthode Six Sigma. Cependant, cet esprit de mesure ne se limite pas à la phase de mise en oeuvre du projet. C'est en fait une véritable culture de la mesure qu'il s'agit d'instaurer au sein même de l'organisation.

Comme le montre la figure 6.9, il existe deux méthodes de mise en oeuvre de Six Sigma :

- la méthodologie DMAIC doit être utilisée quand un produit ou un processus sont déjà existants au sein de l'entreprise, mais ne rencontrent pas la spécification de client ou n'exécutent pas en juste proportion ;
- DFSS, encore appelé DMADV, est utilisé pour concevoir de tous nouveaux produits ou services.



Figure 6.9 – La mise en œuvre de Six Sigma

6.4.4.4 DMAIC : Define, Measure, Analyse, Improve, Control

La méthodologie DMAIC doit être utilisée quand un produit ou un processus sont déjà existants au sein de l'entreprise, mais ne rencontrent pas la spécification de client ou n'exécutent pas en juste proportion [90].

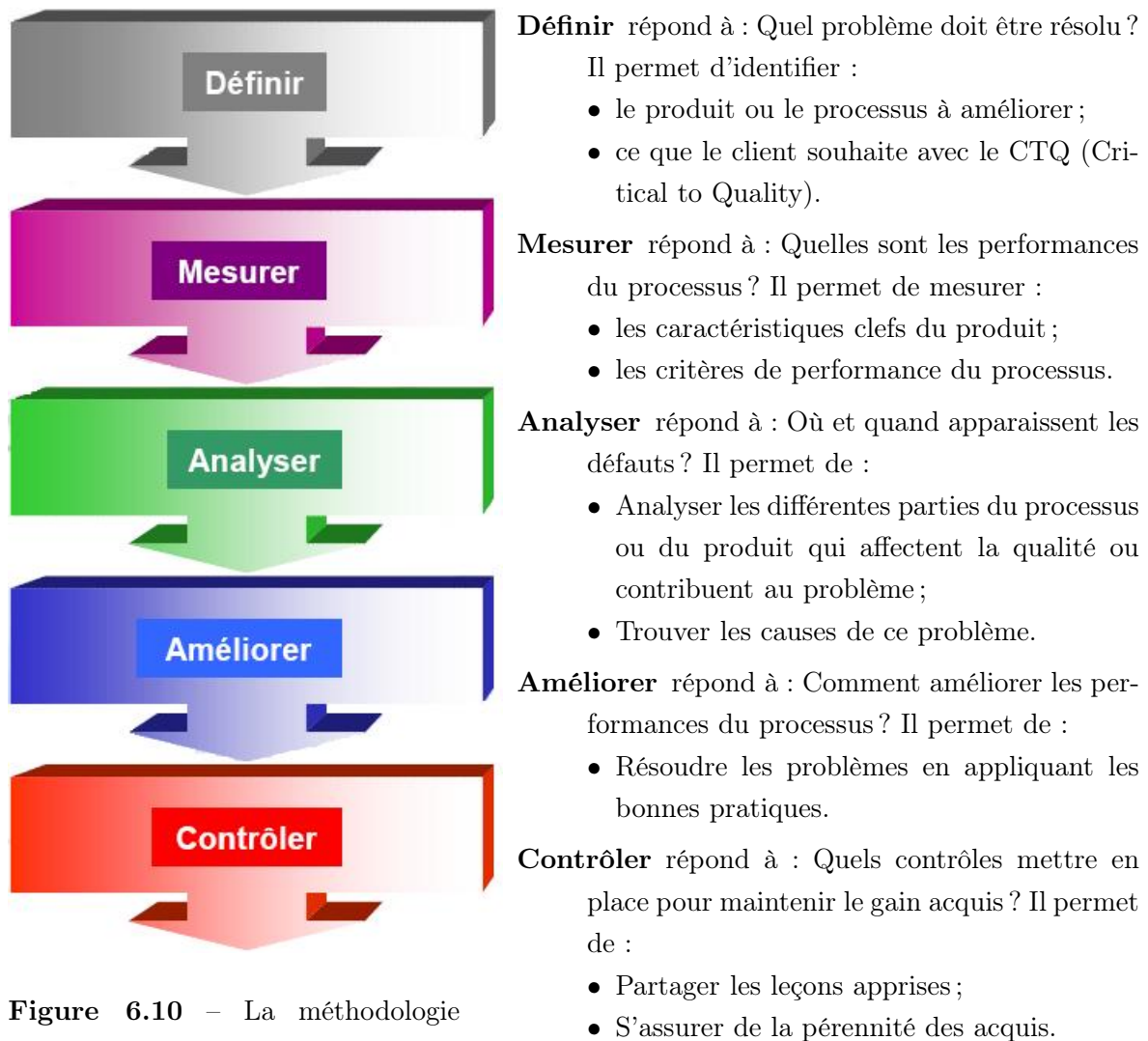


Figure 6.10 – La méthodologie DMAIC

6.4.4.5 DMADV : Define, Measure, Analyze, Design, Verify

DMADV, encore appelé DFSS pour Design For Six Sigma, est utilisé pour concevoir de nouveaux produits ou services [90].

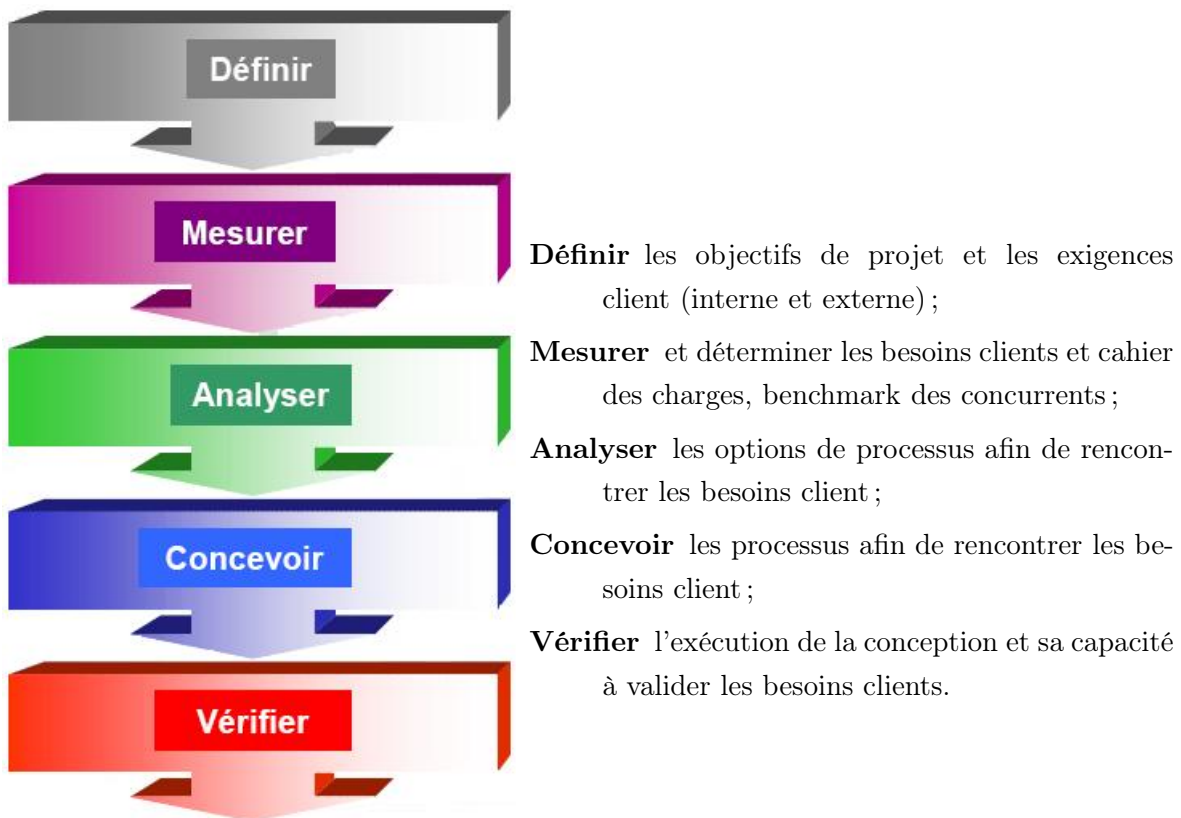


Figure 6.11 – La méthodologie DMADV

6.4.5 COBIT, Control Objectives for Business & Related Technology

6.4.5.1 Présentation de COBIT

Le modèle COBIT (Control Objectives for Business & Related Technology) est une méthode de Maîtrise des Systèmes d'Information (IT Gouvernance) et d'audit de systèmes d'information, éditée par l'Information System Audit & Control Association (ISACA) en 1996. C'est un modèle qui vise à aider le management à gérer les risques (sécurité, fiabilité, conformité) et les investissements [7] [78].

6.4.5.2 Les 5 parties de la méthode COBIT

La synthèse est une présentation des concepts et principes de COBIT. Elle présente les domaines, les objectifs de contrôle généraux (aussi appelés processus) et le cadre de référence. Le modèle COBIT constitue une structure de relations et de processus (cadre de référence ou framework) visant à diriger et contrôler l'entreprise pour qu'elle atteigne ses objectifs, par l'utilisation des technologies pour améliorer l'activité et répondre aux besoins métiers.

Le cadre de référence se décline en check lists méthodiques couvrant 4 domaines, 34 objectifs de contrôle généraux et 302 objectifs de contrôle détaillés. Chacun de ces objectifs répond à 3 familles d'impératifs : fiduciaire, sécurité et qualité.

Comme le montre la figure 6.12, voici la liste des domaines du cadre de référence de COBIT :

- le domaine Planification & Organisation : 11 objectifs couvrent tout ce qui concerne la stratégie et les tactiques. Ils identifient les moyens permettant à l'informatique de contribuer le plus efficacement à la réalisation des objectifs commerciaux de l'entreprise ;
- le domaine Acquisition & Mise en Place : 6 objectifs concernent la réalisation de la stratégie informatique, l'identification, l'acquisition, le développement, l'installation des solutions informatiques et leur intégration dans des processus commerciaux ;
- le domaine Distribution & Support : 13 objectifs regroupent la livraison des prestations informatiques exigées (l'exploitation, la sécurité, les plans d'urgences et la formation) ;
- le domaine Surveillance : 4 objectifs permettent au management d'évaluer la qualité et la conformité des processus informatiques aux exigences de contrôle.

Le guide d'audit permet d'évaluer et de justifier les risques et les faiblesses des objectifs généraux et détaillés et de mettre en place des actions correctives (voir la figure 6.13). Ce guide d'audit répond à 4 principes :

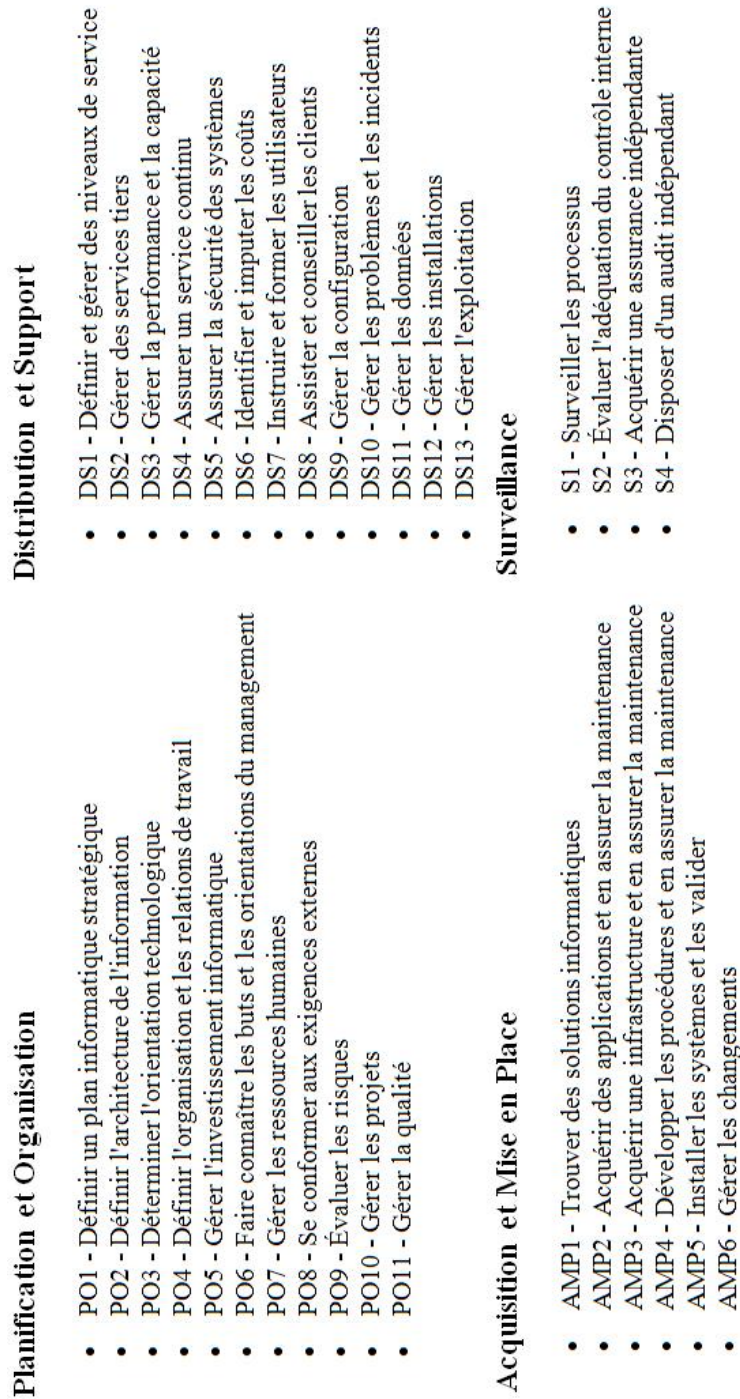


Figure 6.12 – Le cadre de référence de COBIT

- l'acquisition d'une bonne compréhension ;
- l'évaluation des contrôles ;
- la vérification de la conformité ;
- la justification du risque de ne pas atteindre les objectifs de contrôle.

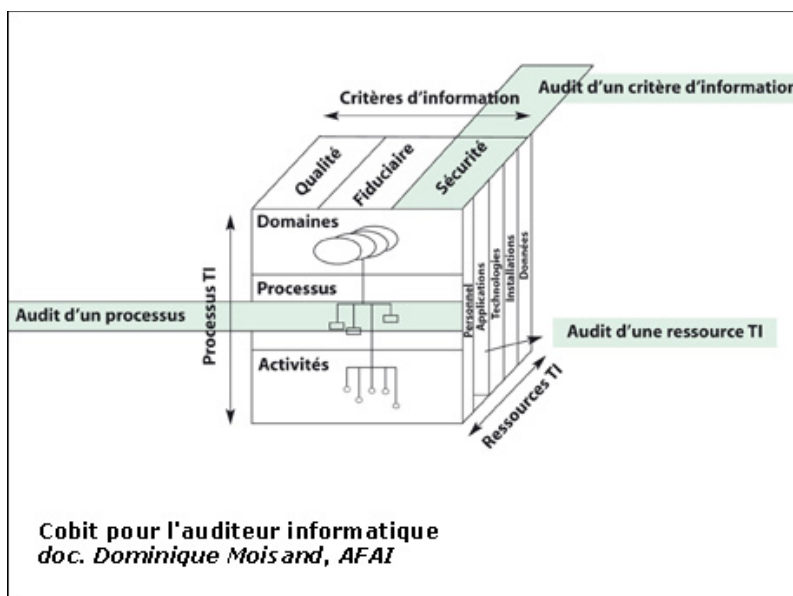


Figure 6.13 – Guide d'audit de COBIT

Le guide de management fournit des indicateurs clés d'objectifs et de performances et des facteurs clés de succès. C'est aussi dans ce guide que l'on trouve le modèle de maturité. Il évalue l'atteinte d'un ou plusieurs objectifs généraux sous forme d'une échelle de 0 à 5 :

- 0 : Inexistant ;
- 1 : Existant mais non organisé (initialisé au cas par cas) ;
- 2 : Décrit (reproductible mais intuitif) ;
- 3 : Défini (avec documentation) ;
- 4 : Surveillé et mesuré ;
- 5 : Optimisé.

Les outils de la mise en œuvre contiennent une présentation de "success story" d'entreprises qui ont mis en place rapidement et avec succès la méthode COBIT. Cette partie intègre deux outils d'analyse de sensibilisation du management et de diagnostic de contrôle informatique.

Le COBIT est donc étroitement lié aux objectifs de l'entreprise tout en s'intéressant plus parti-

culièrement à l'informatique. Il permet de rassurer le management, d'uniformiser les méthodes de travail et de garantir la sécurité et les contrôles de leurs services informatiques.

6.4.6 ITIL, Information Technology Infrastructure Library

Créé à la fin des années 80, à l'initiative du secteur public britannique, ITIL (Information Technology Infrastructure Library) est un référentiel de bonnes pratiques pour la gestion des services informatiques (internes et externes) [37].

ITIL s'intéresse à la production, qu'il s'agisse de fourniture de services informatiques ou d'exploitation interne.

6.4.6.1 Principe d'ITIL

ITIL vise à améliorer les services informatiques (prestation de service, service aux utilisateurs...) suivant plusieurs principes :

- constituer un catalogue de bonnes pratiques à partir de démarches reconnues ;
- mettre le client au centre de la démarche de la DSI ;
- organiser les activités suivant des processus bien établis ;
- jeter les bases d'une évolutivité dans le temps des processus et services.

ITIL met en place plusieurs moyens pour :

- organiser les processus informatiques :
 - définir les processus et leurs interactions ;
 - définir les rôles et les responsabilités.
- améliorer les relations clients-fournisseurs :
 - instituer un point central de relation entre client et fournisseur : le service desk ;
 - définir des moyens d'échanges entre l'entreprise et sa DSI aux niveaux stratégique, tactique et opérationnel.
- définir des moyens de contrôle des résultats obtenus ;
- utiliser des processus ou des méthodologies connues par ailleurs.

6.4.6.2 Description d'ITIL

Comme le montre la figure 6.14, ITIL comporte 6 modules principaux :

- Service Support : support aux utilisateurs ;
- Service Delivery : gestion du service ;
- Software Asset Management : gestion des logiciels ;

- ICT Infrastructure Management : contrôle des processus ;
- Application Management : gestion des projets ;
- Security Management : gestion de la sécurité.

ITIL comporte aussi un module sur l'utilisation d'ITIL et un autre sur l'avenir des services IT. Les 2 premiers (Service Support et Service Delivery) constituent la base d'ITIL. Dans le Service Support, le Service Desk est le point de contact avec l'utilisateur [26].

Le Service Support est destiné au support des utilisateurs et à la gestion des services, suivant 6 processus :

- Service Desk : point central de relation entre client et fournisseur ;
- Incident Management : définition des activités et opérations pour la reprise rapide après incident ;
- Problem Management : surveillance du niveau de service et analyse des incidents ;
- Configuration Management : description détaillée du SI et de ses processus ;
- Change Management : mécanismes de conduite du changement ;
- Release Management : processus de mise en service des nouvelles activités.

Le Service Delivery organise la gestion des services (besoins, ressources, niveau de service, contrôle des coûts. . .) suivant 6 processus :

- Service Level Management : description des services et mises en place avec l'utilisateur. Niveaux de service (SLA) ;
- Capacity Management : gestion des ressources et définition des objectifs de l'entreprise ;
- Availability Management : contrôle de la conformité du niveau de service par rapport aux engagements ;
- Service Continuity Management : définition et utilisation d'un plan de surveillance de la continuité du service (sécurité) ;
- Financial Management : définition des seuils de rentabilité, décisionnel financier ;
- Customer Relationship Management.

6.4.7 PMBoK, Project Management Body of Knowledge

Le Project Management Body of Knowledge, ou PMBoK, est un standard international élaboré par le PMI (Project Management Institute) qui décrit l'ensemble des pratiques et des techniques avancées intervenant dans le management d'un projet (corpus de connaissances). Il fournit un référentiel commun avec une terminologie claire et précise des phases, livrables et compétences nécessaires au chef de projet. Tous les aspects du management de projet y sont traités avec une approche " process " qui permet de systématiser les meilleures pratiques et donc de s'améliorer en permanence [88].

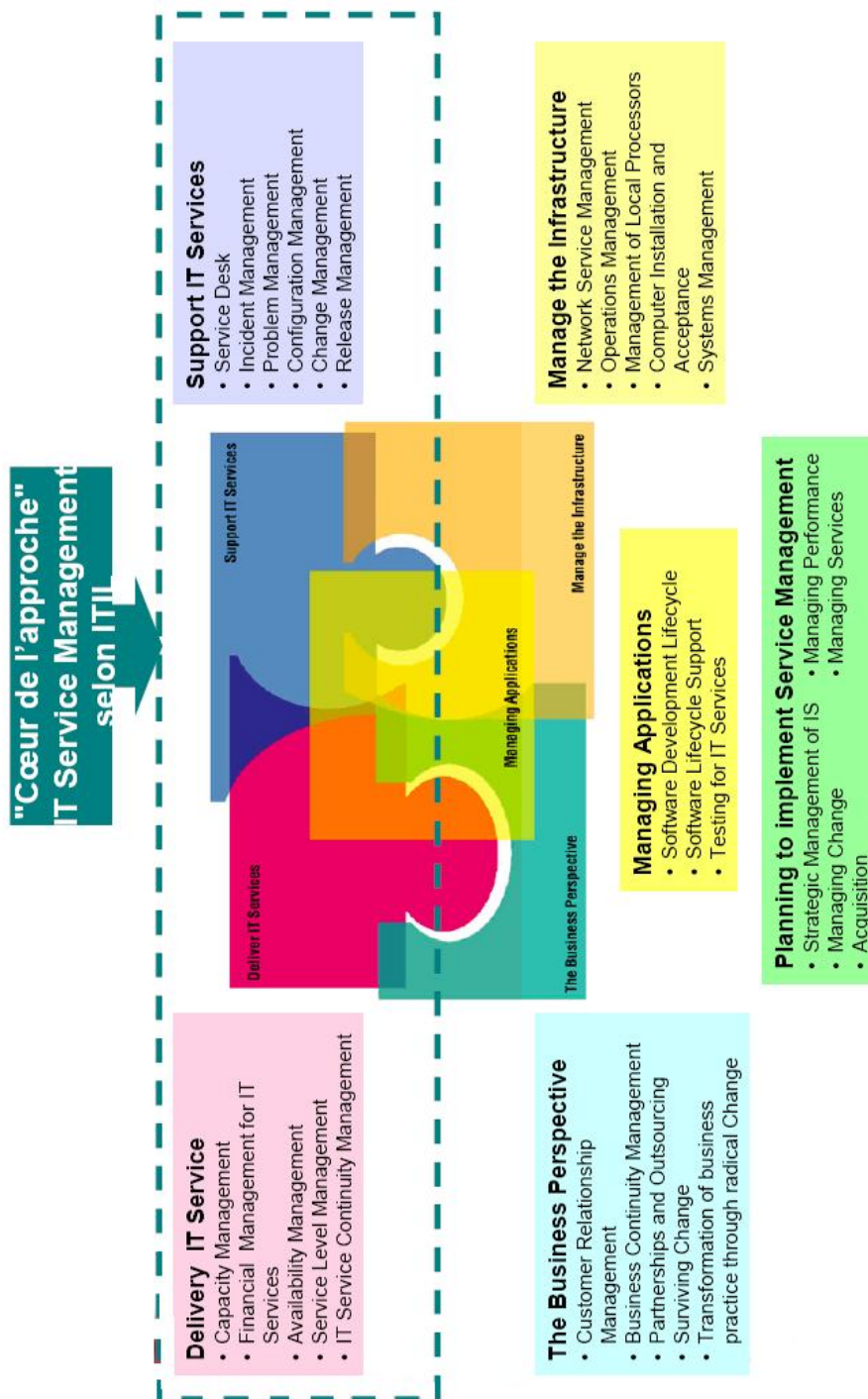


Figure 6.14 – Le framework d'ITIL

PMI a catégorisé les groupes de processus en cinq groupes cohérents :

- Démarrage ;
- Planification ;
- Réalisation ;
- Contrôle ;
- Clôture.

Comme le montre la figure 6.15, ces groupes de processus, qui se répètent dans chacune des phases du projet, s'avèrent critiques pour s'assurer qu'aucune étape n'est involontairement oubliée. De même, les disciplines que le chef de projet sera amené à maîtriser sont divisées en neuf domaines de Management :

- management de l'intégration du projet ;
- management du contenu du projet ;
- management des délais du projet ;
- management des coûts du projet ;
- management de la qualité du projet ;
- management des ressources humaines du projet ;
- management de la communication du projet ;
- management des risques du projet ;
- management des approvisionnements du projet.

Le PMBoK s'intègre parfaitement dans une démarche d'amélioration continue des processus basée sur une approche CMMI ou ISO 9001 en proposant des pratiques éprouvées issues de l'expérience de plus de 90000 personnes travaillant ensemble au sein du PMI dans plus de 100 pays.

Le PMBoK sert également de support à la certification de chef de projet PMP (Project Management Professional) actuellement décernée à 44000 personnes à travers le monde par le PMI.

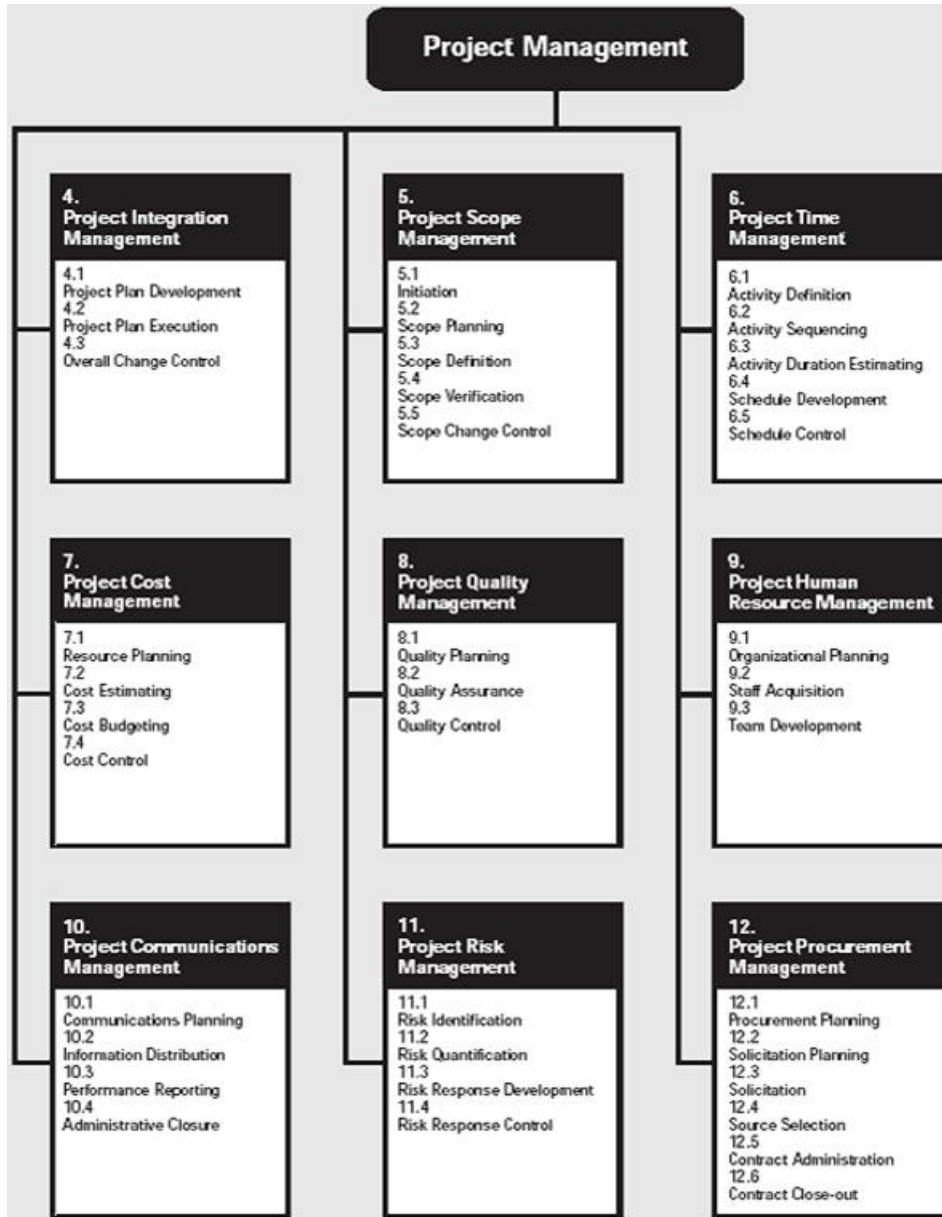


Figure 6.15 – Les processus de PMBOK

6.4.8 SWEBoK, SoftWare Engineering Body of Knowledge

Version génie logiciel du PMBoK, SWEBoK est un corpus de connaissances en génie logiciel (de 200 pages environs). Ce corpus a été élaboré par l'IEEE entre 1998 et 2004. La version de 2004 a reçu l'approbation de " Industrial Advisory Board " américain [58].

L'IEEE a reconnu 8 disciplines reliées au génie logiciel :

- Computer engineering ;
- Project management ;
- Computer science ;
- Quality management ;
- Management ;
- Software ergonomics ;
- Mathematics ;
- Systems engineering.

Comme le montre la figure 6.16, à travers ce guide, l'IEEE [58] traite 10 domaines du génie logiciel :

- Software requirements ;
- Software design ;
- Software construction ;
- Software testing ;
- Software maintenance ;
- Software configuration management ;
- Software engineering management ;
- Software engineering process ;
- Software engineering tools and methods ;
- Software quality.

Tout comme le PMBoK, le SWEBoK s'intègre parfaitement dans une démarche d'amélioration continue des processus basée sur une approche CMMI ou ISO 9001 en proposant des pratiques éprouvées issues de l'expérience.

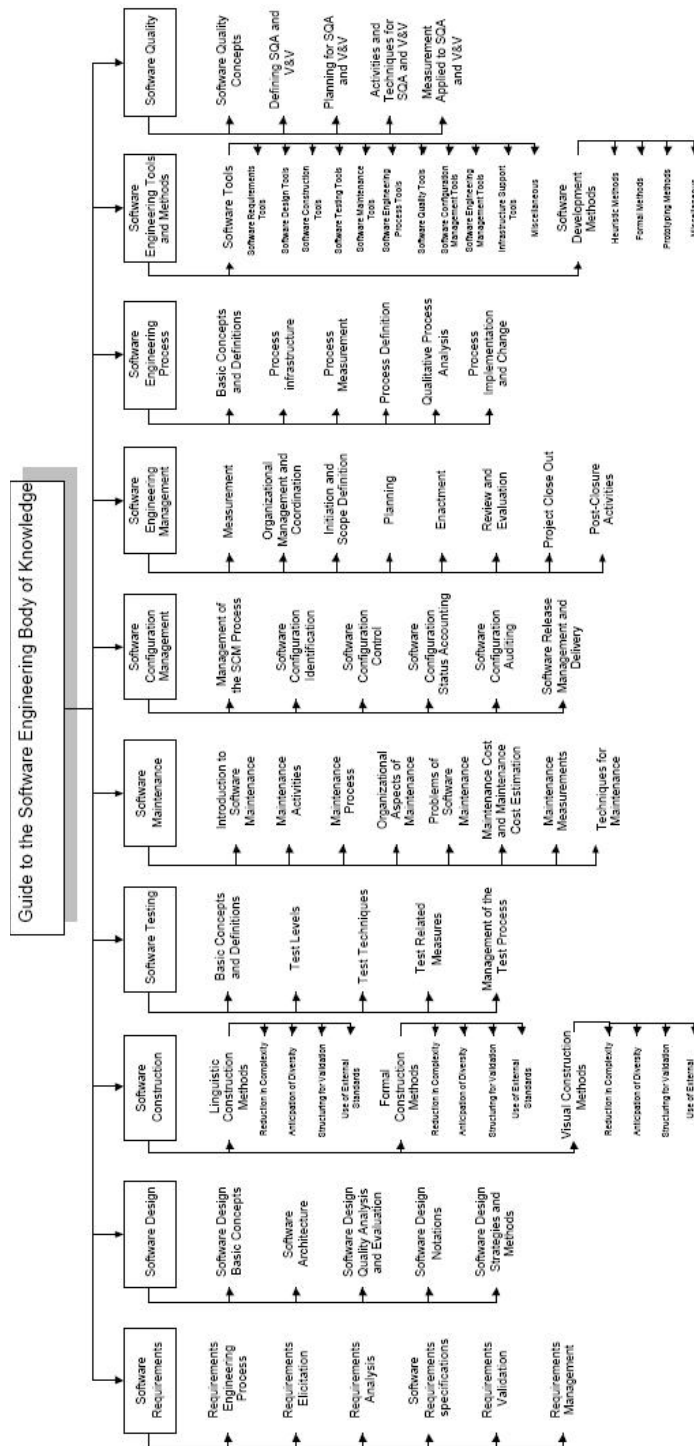


Figure 6.16 – Les 10 domaines de SWEBOK

6.4.9 ISO 9001v2000 "Systèmes de management de la qualité - Exigences"

6.4.9.1 Présentation ISO 9001v2000

Les normes ISO 9000 ont été originellement écrites en 1987, puis elles ont été révisées en 1994 et à nouveau en 2000. La norme ISO 9001v2000 porte essentiellement sur les processus permettant de réaliser un service ou un produit alors que la norme ISO 9001 :1994 était essentiellement centrée sur le produit lui-même [6]. Voici une présentation synthétique des différentes normes de la famille ISO 9000 :

ISO 9000 : "Systèmes de management de la qualité - Principes essentiels et vocabulaire".

La norme ISO 9000 décrit les principes d'un système de management de la qualité et définit la terminologie ;

ISO 9001 : "Systèmes de management de la qualité - Exigences". La norme ISO 9001 décrit les exigences relatives à un système de management de la qualité pour une utilisation soit interne, soit à des fins contractuelles ou de certification. Il s'agit ainsi d'un ensemble d'obligations que l'entreprise doit suivre ;

ISO 9004 : "Systèmes de management de la qualité - Lignes directrices pour l'amélioration des performances". Cette norme, prévue pour un usage en interne et non à des fins contractuelles, porte notamment sur l'amélioration continue des performances ;

ISO 10011 : "Lignes directrices pour l'audit des systèmes de management de la qualité et/ou de management environnemental".

La présente norme internationale encourage l'adoption d'une approche processus lors du développement, de la mise en œuvre et de l'amélioration de l'efficacité d'un système de management de la qualité, afin d'accroître la satisfaction des clients par le respect de leurs exigences [34]. Lorsqu'elle est utilisée dans un Système de Management de la Qualité (SMQ), cette approche souligne l'importance de :

- 1) comprendre et de satisfaire les exigences ;
- 2) considérer les processus en termes de valeur ajoutée ;
- 3) mesurer la performance et l'efficacité des processus ;
- 4) améliorer en permanence des processus sur la base de mesures objectives.

Le Système de Management de la Qualité (SMQ) s'appuie ainsi sur divers documents permettant de décrire, avec un niveau de précision grandissant, les activités de l'entreprise et leurs interactions :

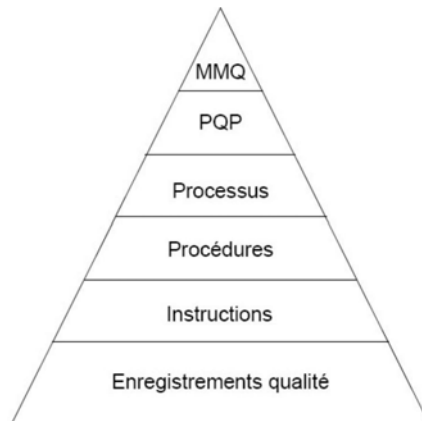


Figure 6.17 – Pyramide documentaire ISO 9001

Le Manuel de Management de la Qualité (MMQ) qui présente la politique, l'organisation et les principes mis en œuvre dans le cadre du SMQ ainsi que les dispositions pour assurer la conformité aux exigences contractuelles, internes et réglementaires.

Le Plan Qualité Projet (PQP) qui décrit l'organisation de l'équipe de projet, la gestion des configurations, les processus de réalisation et les règles établies au sein du projet en adéquation avec le SMQ mis en place et les souhaits des clients.

Les processus décrivent l'enchaînement des activités réalisées, les responsabilités, les éléments d'entrées et de sorties.

Les procédures qui décrivent les tâches à réaliser pour une activité donnée faisant partie d'un processus.

Les instructions qui décrivent la façon d'effectuer une tâche donnée.

Les documents d'enregistrement qualité sont mis en place pour faciliter la gestion et l'enregistrement des activités. Constitués de modèles de document, ils permettent de rendre compte, d'établir et de conserver l'information.

La réalisation du Système de Management de la Qualité (SMQ) doit répondre aux exigences des 8 chapitres de la norme décrit ci-dessous :

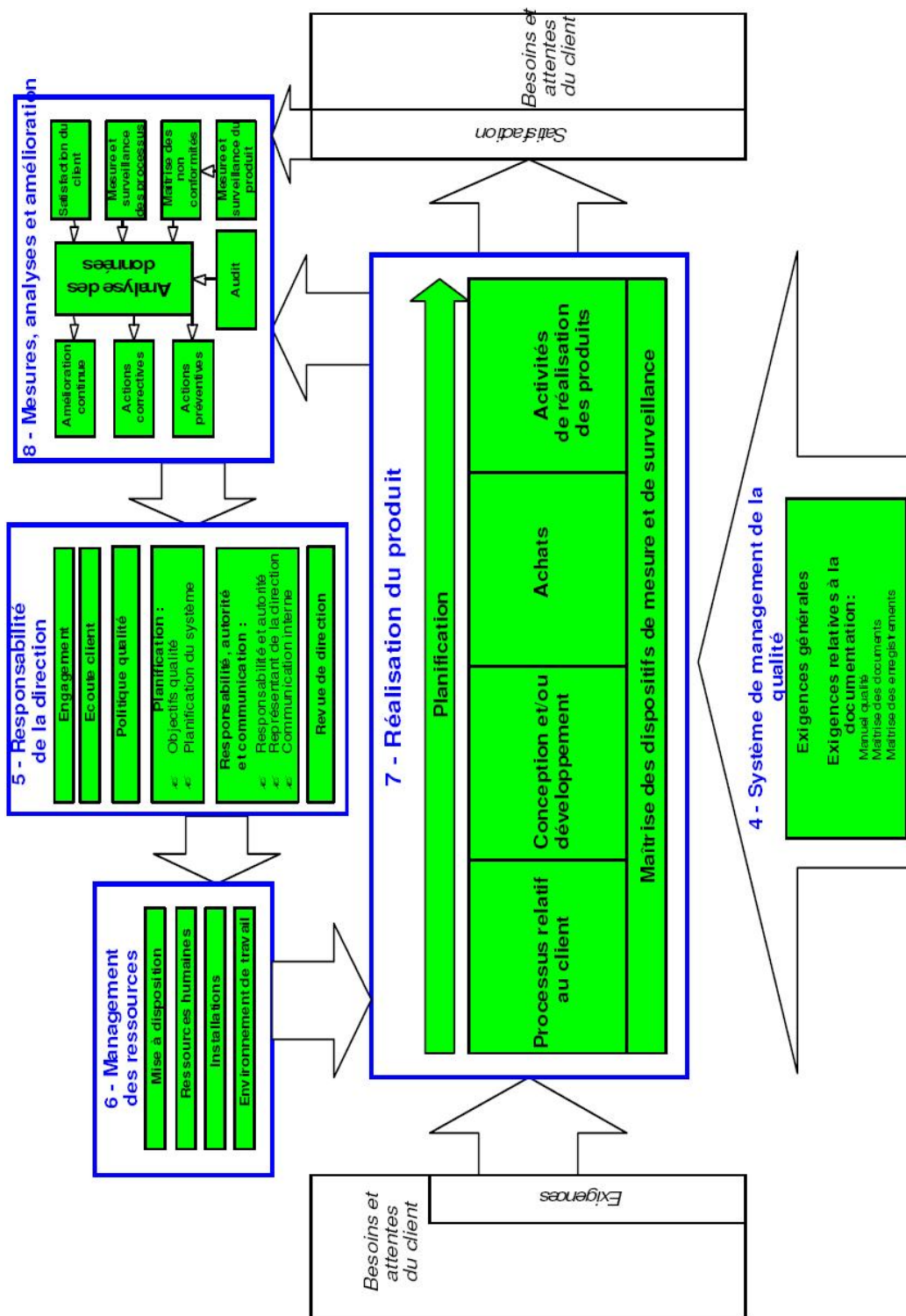


Figure 6.18 – Les chapitres de la norme ISO

6.4.9.2 Contenu d'ISO 9001v2000

La figure 6.19 montre le contenu des chapitres d'ISO 9001v2000 [103] [38].

1	Domaine d'application	7	Réalisation du produit
2	Référence normative	7.1	Planification de la réalisation du produit
3	Termes et définitions	7.2	Processus relatifs aux clients
4	Système de management de la qualité	7.2.1	<i>Détermination des exigences relatives au produit</i>
4.1	Exigences générales	7.2.2	<i>Revue des exigences relatives au produit</i>
4.2	Exigences relatives à la documentation	7.2.3	<i>Communication avec les clients</i>
4.2.1	<i>Généralités</i>	7.3	Conception et développement
4.2.2	<i>Manuel qualité</i>	7.3.1	<i>Planification de la conception et du développement</i>
4.2.3	<i>Maîtrise des documents</i>	7.3.2	<i>Éléments d'entrée de la conception et du développement</i>
4.2.4	<i>Maîtrise des enregistrements</i>	7.3.3	<i>Éléments de sortie de la conception et du développement</i>
5	Responsabilité de la direction	7.3.4	<i>Revue de la conception et du développement</i>
5.1	Engagement de la direction	7.3.5	<i>Vérification de la conception et du développement</i>
5.2	Écoute client	7.3.6	<i>Validation de la conception et du développement</i>
5.3	Politique qualité	7.3.7	<i>Maîtrise des modifications de la conception et du développement</i>
5.4	Planification	7.4	Achats
5.4.1	<i>Objectifs qualité</i>	7.4.1	<i>Processus d'achat</i>
5.4.2	<i>Planification du système de management de la qualité</i>	7.4.2	<i>Informations relatives aux achats</i>
5.5	Responsabilité, autorité et communication	7.4.3	<i>Vérification du produit acheté</i>
5.5.1	<i>Responsabilité et autorité</i>	7.5	Production et préparation du service
5.5.2	<i>Représentant de la direction</i>	7.5.1	<i>Maîtrise de la production et de la préparation du service</i>
5.5.3	<i>Communication interne</i>	7.5.2	<i>Validation des processus de production et de préparation du service</i>
5.6	Revue de direction	7.5.3	<i>Identification et traçabilité</i>
5.6.1	<i>Généralités</i>	7.5.4	<i>Propriété du client</i>
5.6.2	<i>Éléments d'entrée de la revue</i>	7.5.5	<i>Préservation du produit</i>
5.6.3	<i>Éléments de sortie de la revue</i>	7.6	Maîtrise des dispositifs de surveillance et de mesure
6	Management des ressources	8	Mesures, analyse et amélioration
6.1	Mise à disposition des ressources	8.1	Généralités
6.2	Ressources humaines	8.2	Surveillance et mesures
6.2.1	<i>Généralités</i>	8.2.1	<i>Satisfaction du client</i>
6.2.2	<i>Compétence, sensibilisation et formation</i>	8.2.2	<i>Audit interne</i>
6.3	Infrastructures	8.2.3	<i>Surveillance et mesure des processus</i>
6.4	Environnement de travail	8.2.4	<i>Surveillance et mesure du produit</i>
		8.3	Maîtrise du produit non conforme
		8.4	Analyse des données
		8.5	Amélioration
		8.5.1	<i>Amélioration continue</i>
		8.5.2	<i>Action corrective</i>
		8.5.3	<i>Action préventive</i>

Figure 6.19 – Le contenu de la norme ISO 9001v2000.

6.4.9.3 PDCA : Plan, Do, Check, Act, "la roue de Deming"

Défini par Deming [109], PDCA est basé sur le concept de la remise en cause des méthodes de travail, des flux d'information, des fonctions, des documents de travail utilisés... sur le concept de l'amélioration continue. La remise en cause comprend l'analyse de toutes les anomalies rencontrées dans la vie d'une entreprise, vues ou pas encore vues par les clients. Ce processus constitué de 4 phases (Planifier, Faire, Vérifier, Améliorer) permet de structurer

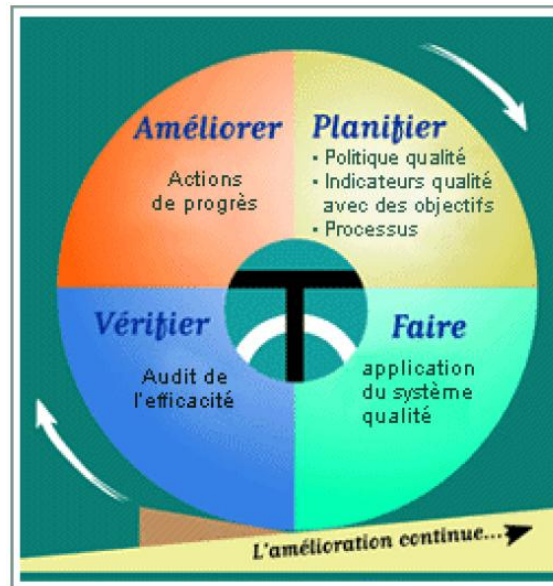


Figure 6.20 – PDCA : Plan, Do, Check, Act, "la roue de Deming".

la démarche d'amélioration continue définie par ISO 9001v2000. En effet, le cycle PDCA est une méthode qui permet d'exécuter un travail de manière efficace et rationnelle (comme le montre la figure 6.21).

Mots clés :	Étapes :
<p>1 Plan / Préparer</p> <p>Objectif : poser le vrai problème, trouver les causes racines et choisir les solutions optimums.</p>	<p>Étape 1 : Cette étape est constituée de cinq phases :</p> <ul style="list-style-type: none"> ▪ Identifier le problème ▪ Construire un groupe de travail ▪ Formaliser le problème (outils QOOQCCP, où le "Qui" peut enrichir le groupe de travail) ▪ Mesurer la situation actuelle grâce à la définition d'indicateurs représentatifs du problème ▪ Définir l'objectif <p>Étape 2 : Trouver les causes racines Cette étape est constituée de quatre phases :</p> <ul style="list-style-type: none"> ▪ Rechercher les causes (brainstorming), ▪ Visualiser les causes (Diagramme d'Ishikawa), ▪ Hiérarchiser les causes (vote pondéré), ▪ Valider les causes principales (Diagramme de Pareto). <p>Étape 3 : Choisir des solutions optimums Cette étape est constituée de deux phases :</p> <ul style="list-style-type: none"> ▪ Re-chercher les solutions (brainstorming), ▪ Sélectionner les solutions (analyse multicritères).
<p>2 Do / Dérouler</p> <p>Objectif : Établir le plan d'action, mettre en place toutes les actions indiquées dans le plan d'action.</p>	<p>Étape 4 : Mettre en œuvre la solution retenue Cette étape est constituée de trois phases :</p> <ul style="list-style-type: none"> ▪ Définir la zone d'expérimentation, ▪ Rédiger un plan d'action, ▪ Réaliser toutes les actions définies
<p>3 Check / Contrôler</p> <p>Objectif : Vérifier que les actions mises en place sont efficaces et atteignent l'objectif défini.</p>	<p>Étape 5 : Mesurer les résultats des solutions mises en place et les comparer à la situation initiale.</p>
<p>4 Act / Assurer</p> <p>Objectif : Vérifier que les actions mises en place sont efficaces dans le temps.</p>	<p>Étape 6 :</p> <p>Cette étape est constituée de trois phases :</p> <ul style="list-style-type: none"> ▪ Formaliser les solutions et dans certains cas mettre en place des systèmes anti-erreur, ▪ Généraliser les solutions si possible ▪ Valoriser le groupe de travail et les personnes ayant mis en œuvre les actions.

Figure 6.21 – Les étapes de PDCA.

6.4.10 CMMI, Capability Maturity Model Integration

6.4.10.1 Présentation de CMMI

Le Capability Maturity Model Integration SE/SW/IPPD/SS v1.1 a été publié en 2003 par le Software Engineering Institute (SEI) de l'université Camegie Mellon [9]. Une mise à jour a été publiée en 2006 portant le nom de CMMI for Development, Version 1.2 [99] [28]. CMMI est un modèle de bonnes pratiques qui repose sur une amélioration progressive des processus de développement informatique de l'entreprise.

Comme le montre la figure 6.22, CMMI intègre d'anciens modèles développés dans les années 90. Chacun de ces modèles est spécifique à des domaines distincts de l'ingénierie informatique.

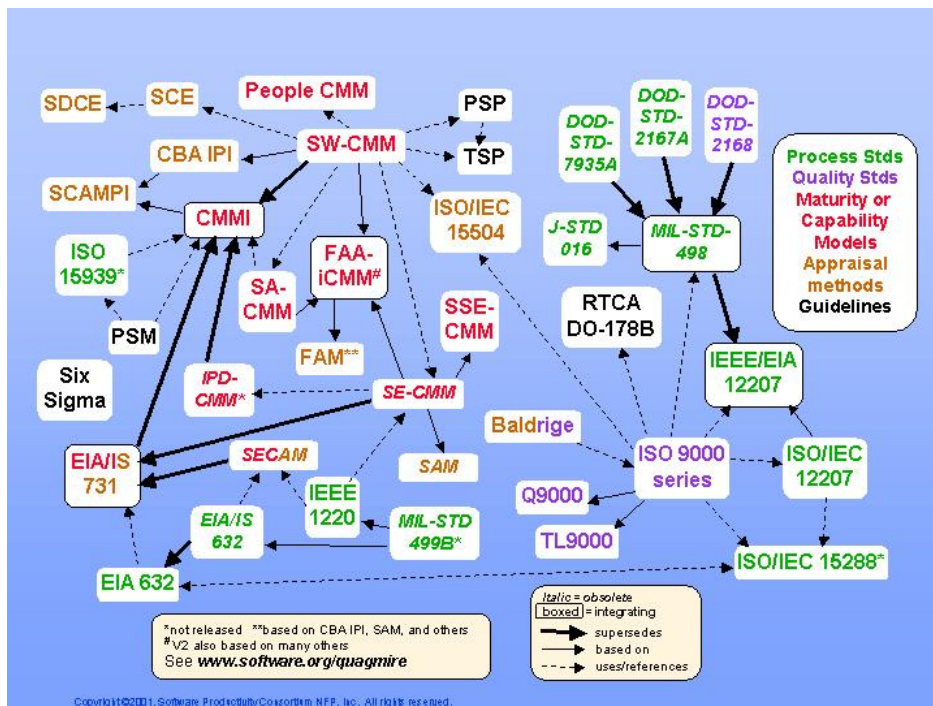


Figure 6.22 – Les anciens modèles à l'origine de CMMI.

Ce besoin en intégration est apparu afin de faire parler le même langage et utiliser des processus communs à des ingénieurs de disciplines multiples attachés à un même projet de développement :

- SE (Systems Engineering) : Ingénierie système ;
- SW (SoftWare) : Développement logiciel ;
- SS (Supplier Sourcing) : Gestion des fournisseurs ;

- IPPD (Integrated Product and Process Development) : Développement intégré des produits et processus.

CMMI regroupe 25 domaines de processus "Process Area" (PA).

Deux modèles de représentation coexistent, correspondant à deux points de vue légèrement différents. Les deux représentations s'appuient sur les mêmes PAs mais ceux-ci sont utilisés différemment :

représentation continue : La représentation continue privilégie de regarder l'évolution au niveau de chaque domaine de processus (PA) et d'y associer des stades de développement de la capacité du processus ;

représentation étagée : La représentation étagée préconise une évolution par étage au niveau de toute l'organisation de développement. Elle permet ainsi de définir le niveau de maturité (ML) de l'organisation.

6.4.10.2 CMMI représentation étagée

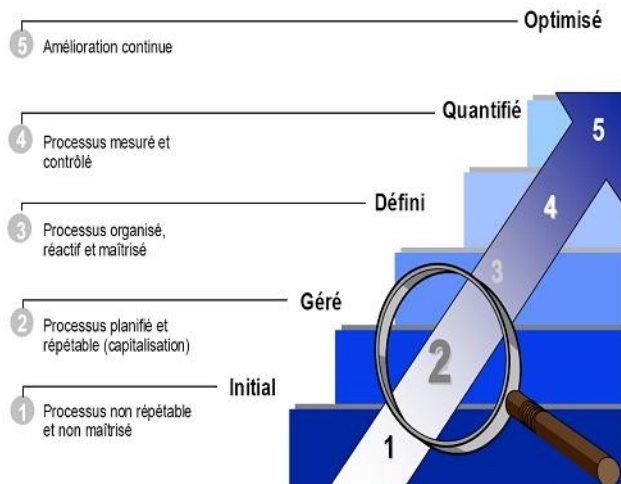


Figure 6.23 – Représentation étagée.

Comme le montre la figure 6.23, la représentation étagée exprime l'évolution des pratiques de développement en fonction d'une vue organisationnelle.

La première étape est de définir le périmètre à observer : "l'organisation". Puis, on observe son comportement au regard des 5 niveaux de maturité (ML) regroupant les 25 PAs (voir la figure 6.24) [9] :

- ML 1 : Niveau "Initial" où les processus sont non répétables et non maîtrisés.

- ML 2 : Niveau "Managed" où les processus sont planifiés et répétables. Le ML 2 traduit la mise en œuvre effective de pratiques de base au niveau des projets mais sans harmonisation au niveau de l'organisation (par exemple chaque projet gère différemment sa planification, ses configurations. . .) ; il s'adresse essentiellement aux Chefs de Projet. Le ML 2 est constitué de 7 PAs :
 - Gestion des exigences ;
 - Planification de projet ;
 - Suivi et contrôle de projet ;
 - Gestion des ententes avec les fournisseurs ;
 - Mesure et analyse ;
 - Assurance qualité processus et produit ;
 - Gestion de configuration.
- ML 3 : Niveau "Defined" où les processus sont organisés, réactifs et maîtrisés. Le ML 3 traduit le déploiement harmonisé de pratiques définies au niveau de l'organisation ; il s'adresse à toute la population et met l'accent sur l'engineering produit et l'organisation centrale. Le ML 3 est constitué de 14 PAs :
 - Développement des exigences ;
 - Solution technique ;
 - Intégration produit ;
 - Vérification ;
 - Validation ;
 - Focalisation sur le processus organisationnel ;
 - Définition du processus organisationnel ;
 - Formation organisationnelle ;
 - Gestion de projet intégrée dans un contexte IPPD ;
 - Gestion du risque ;
 - Équipe intégrée ;
 - Gestion de fournisseur intégrée ;
 - Analyse et prise de décision ;
 - Environnement organisationnel en vue de l'intégration.
- ML 4 : Niveau "Qualitatively managed" où les processus sont mesurés et contrôlés. Le ML 4 traduit une gestion quantitative des processus, basée sur des analyses statistiques (les performances de l'organisation sont prévisibles). Le ML 4 est constitué de 2 PAs :
 - Performance du processus organisationnel ;
 - Gestion de projet quantitative.
- ML 5 : Niveau "Optimizing" où l'organisation est en amélioration continue. Le ML 5

traduit la maturité de l'organisation où veille technologique et amélioration continue concourent à l'excellence du livrable. Le ML 5 est constitué de 2 PAs :

- Innovation et déploiement organisationnels ;
- Analyse causale et résolution.

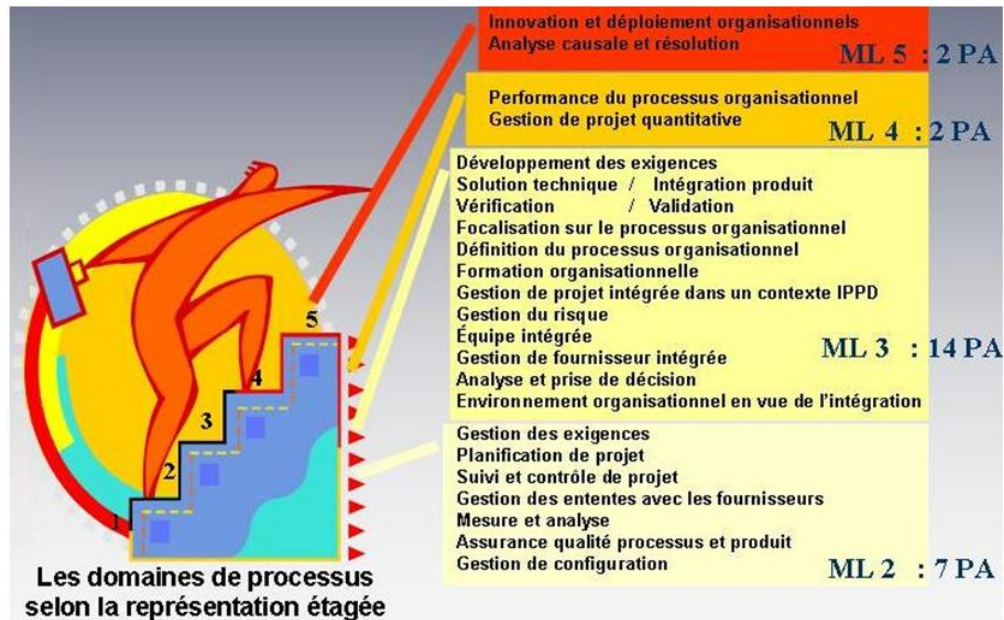


Figure 6.24 – Les domaines de processus (PA) par niveau de maturité (ML)

L'annexe A permet d'expliquer succinctement les Process Areas de CMMI et leurs objectifs.

À partir de ce niveau de maturité, il est possible de repositionner son organisation face à ses concurrents ayant adopté une démarche CMMI.

Chacun des Processus Areas (PA) doit répondre à des objectifs qui peuvent être :

- spécifiques (SG), intrinsèquement reliés à la nature de l'activité couverte par le PA ;
- génériques (GG), transversaux et identiques d'un PA à l'autre.

Ces objectifs se déclinent en pratiques qui décrivent le comportement attendu de la part de l'organisation ou d'un projet (Pratique spécifique (SP), Pratique générique (GP)).

Afin de prouver la bonne implémentation de chacune des pratiques, des preuves PII (Practice Implementation Indicator) doivent être collectées.

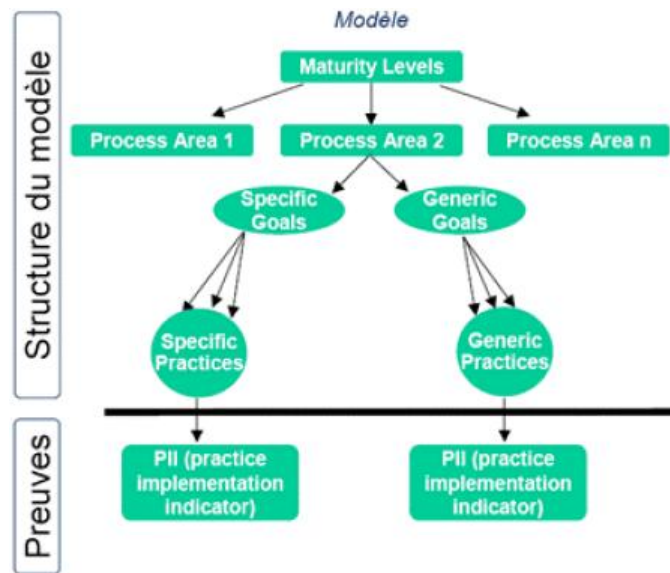


Figure 6.25 – CMMI - Structure du modèle et preuves.

6.4.10.3 CMMI représentation continue

La représentation continue utilise une vision ciblée de l'évolution des pratiques de développement. Contrairement à la représentation étagée qui regroupe sous des niveaux de maturité les différents PAs, la représentation continue exprime la capacité de chaque PA au sein de l'organisation suivant une échelle de 6 niveaux [9].

Cette représentation permet à l'entreprise d'entreprendre un programme d'amélioration sur un bouquet de PAs qu'elle a sélectionnés (voir la figure 6.26).

Une décomposition proposée par le SEI regroupe les 25 Process Area en 4 catégories définissant le **CMMI Framework** :

- Engineering Process Areas : Ils couvrent les activités de développement et de maintenance qui sont partagées entre les différentes disciplines d'ingénierie (ingénierie système, ingénierie logiciel). L'Engineering Process Areas regroupe 6 PAs :
 - Gestion des exigences ;
 - Développement des exigences ;
 - Solution technique ;
 - Intégration produit ;
 - Vérification ;
 - Validation.

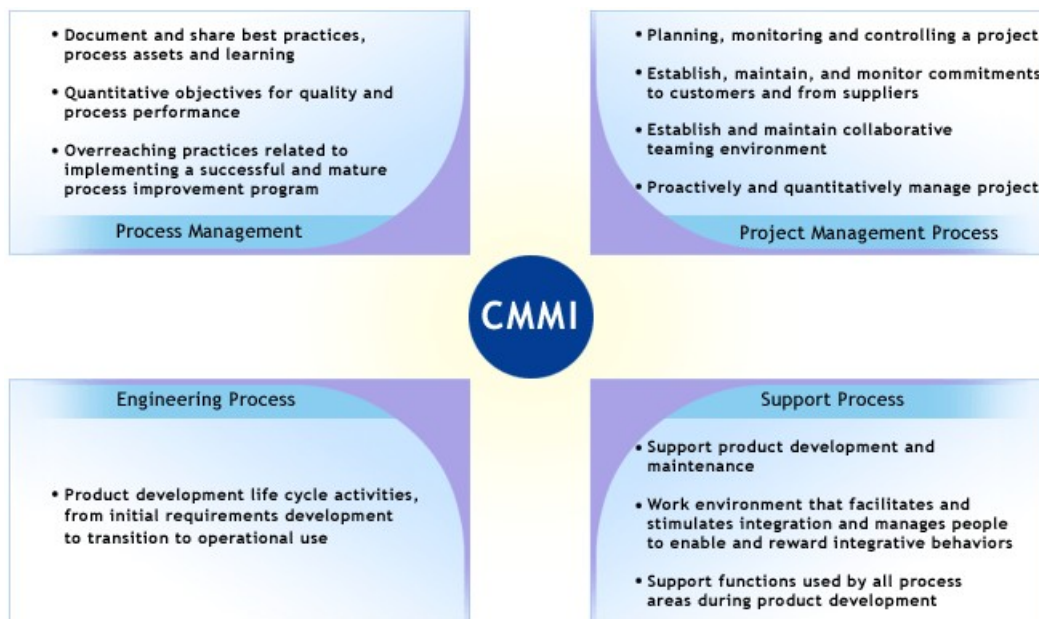


Figure 6.26 – Représentation continue.

Ces PAs définissent un processus de développement du produit plutôt que des processus spécifiques à chaque discipline comme l'ingénierie système ou l'ingénierie logiciel.

- **Project Management Process Areas** : Ils couvrent les activités de gestion de projet liées à la planification, au suivi et au contrôle du projet. Le project management regroupe 8 PAs :
 - Gestion des ententes avec les fournisseurs ;
 - Planification de projet ;
 - Suivi et contrôle de projet ;
 - Gestion de fournisseur intégré ;
 - Gestion de projet intégré ;
 - Équipe intégrée ;
 - Gestion de projet quantitative ;
 - Gestion des risques.
- **Support Process Areas** : Ils couvrent les activités qui soutiennent le développement et la maintenance du produit. Ce sont des processus qui permettent d'exécuter d'autres processus. En général, ils s'adressent à des processus liés au projet mais peuvent aussi s'adresser aux processus de l'organisation. Par exemple, le PA "Process and Product Quality Assurance" peut être utilisé avec tous les autres PAs pour fournir une évaluation objective des processus et des produits de travail décrits dans tous les secteurs de

processus. Le support regroupe 6 PAs :

- Gestion de la configuration ;
 - Assurance qualité processus et produit ;
 - Mesure et analyse ;
 - Environnement organisationnel en vue de l'intégration ;
 - Résolution et analyse pour décision ;
 - Résolution et analyse causale.
- **Process Management Process Areas** : Ils concernent toutes les activités transversales du projet liées à la définition, la planification, le réapprovisionnement, le déploiement, l'exécution, le contrôle, la direction, l'évaluation, la mesure et les processus d'amélioration. Le process management regroupe 5 PAs :
 - Focalisation de l'organisation sur les processus ;
 - Définition des processus de l'organisation ;
 - Formation dans l'organisation ;
 - Performances des Processus de l'organisation ;
 - Innovation et déploiement organisationnel.

6.4.10.4 Apports de CMMI

A travers les retours d'évaluation, le SEI a publié des statistiques sur les apports de l'application de CMMI au sein des organisations [53]. Le tableau 6.1 montre les chiffres clés :

Bénéfices	Médiane	Variation
Amélioration de la productivité	61%	11%-329%
Amélioration de la détection des défauts avant les tests	22%	6%-25%
Amélioration des coûts	34%	3%-87%
Amélioration des délais de livraison	50%	2%-95%
Diminution des anomalies post démarrage	39%	10%-94%
Satisfaction des clients	14%	-4%-55%
Retour sur investissement	4 : 1	1,7 : 1 - 27,7 : 1

Tableau 6.1 – Les apports de CMMI

6.4.10.5 IDEALSM : Initiating, Diagnosing, Establishing, Acting, Learning

Tout comme PDCA pour ISO 9001v2000, le SEI a défini un modèle de support d'amélioration continue [67] [98]. Les paragraphes suivants fournissent une définition générale de chaque étape du modèle (voir la figure 6.27).

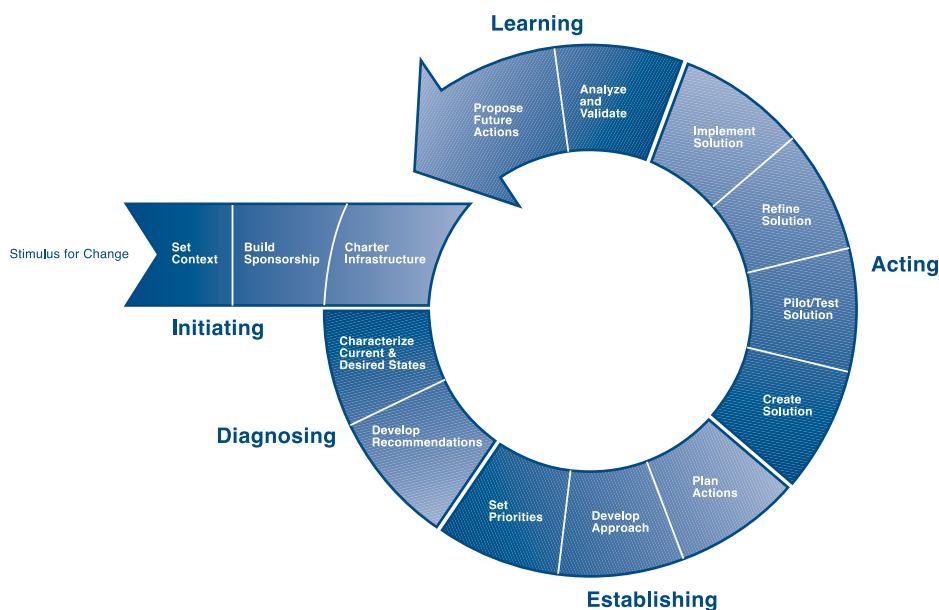


Figure 6.27 – Le modèle IDEAL.

Phase initiale : C'est ici que l'infrastructure d'amélioration initiale est établie, que les rôles et les responsabilités de l'infrastructure sont définis et que les premières ressources sont assignées. Au cours de cette étape, un plan d'amélioration continue des processus est établi pour guider l'organisation jusqu'à la fin des phases initiales, de diagnostic et d'établissement. L'approbation de l'initiative est obtenue de pair avec un engagement relatif aux ressources futures requises pour le travail qu'il reste à faire. Les objectifs généraux du programme sont définis au cours de la phase initiale. Ils sont établis en fonction des besoins de gestion de l'organisation et seront peaufinés et précisés au cours de la phase d'établissement du modèle IDEALSM.

Deux éléments clés sont habituellement établis, soit un groupe directeur de gestion et un groupe d'exécution des processus. En outre, au cours de la phase initiale, des plans sont établis pour annoncer le démarrage de l'initiative et il est suggéré d'effectuer des évaluations organisationnelles pour établir l'état de préparation de l'organisation à l'égard d'une initiative d'amélioration.

Phase de diagnostic : La phase de diagnostic du modèle IDEALSM lance l'organisation sur la voie de l'amélioration continue des processus. Elle sert à établir les fondements pour les prochaines étapes. Au cours de cette phase, le plan d'action est mis en œuvre conformément à la vision de l'organisation, à son plan d'activités stratégiques, aux leçons tirées à la suite d'efforts d'amélioration antérieurs, aux questions de gestion clés que doit régler l'organisation et à ses objectifs à long terme. Des activités d'évaluation sont exécutées afin d'établir la ligne de référence de l'état actuel de l'organisation. Les résultats obtenus et les recommandations formulées à partir de ces évaluations et de toutes les autres activités servant à établir la base de référence seront comparés aux efforts d'amélioration en cours ou prévus afin d'être intégrés dans le plan d'action.

Phase d'établissement : Au cours de la phase d'établissement, les enjeux sur lesquels l'organisation a décidé de se pencher dans le cadre de ses activités d'amélioration sont classés par ordre de priorité. Des stratégies en vue d'apporter des solutions sont mises au point. Le projet de plan d'action sera élaboré conformément à la vision de l'organisation, à son plan d'activités stratégiques, aux leçons tirées à la suite d'efforts d'amélioration antérieurs, aux questions de gestion clés que doit régler l'organisation et à ses objectifs à long terme.

Au cours de la phase d'établissement, des objectifs mesurables sont dégagés à partir des objectifs généraux définis au cours de la phase initiale et seront inclus dans la version finale du plan d'action. Les paramètres requis pour surveiller les progrès réalisés sont également définis, des ressources sont engagées et la formation est dispensée aux groupes de travail technique ou aux équipes d'exécution des processus. Le plan d'action mis au point servira à orienter les activités d'amélioration puisqu'il tient compte des constatations et des recommandations priorisées de la phase de diagnostic. En outre, au cours de cette phase, des gabarits de plan d'action tactique sont créés et mis à la disposition des équipes d'exécution des processus qui les complèteront et les réaliseront.

Phase d'exécution : La phase d'exécution du modèle IDEALSM sert à l'élaboration de solutions touchant les secteurs d'amélioration relevés au cours de la phase de diagnostic, à leur mise à l'essai et à leur diffusion dans toute l'organisation. Des plans seront élaborés afin d'exécuter des projets pilotes pour vérifier et évaluer les processus nouveaux ou améliorés. Lorsque l'exécution réussie de projets pilotes aura démontré que les nouveaux processus peuvent être adoptés, diffusés et institutionnalisés dans toute l'organisation, des plans pour les instaurer seront alors élaborés et mis en œuvre.

Phase d'amélioration : La phase d'amélioration a pour objectif de rendre la prochaine utilisation du modèle IDEALSM plus efficace. À cette étape, les solutions ont été élaborées, les leçons ont été tirées et les paramètres relatifs au rendement et à la réalisation des

objectifs ont été cernés. Ces objets ont été ajoutés à la base de données sur les processus qui deviendra une source d'information pour le personnel chargé d'utiliser à nouveau le modèle. À l'aide de l'information recueillie, il sera possible d'évaluer la stratégie, les méthodes et l'infrastructure utilisées au cours du programme d'amélioration. Ainsi, des correctifs ou des ajustements peuvent être apportés à la stratégie, aux méthodes ou à l'infrastructure avant le début du prochain cycle d'amélioration des processus. Les questions à se poser sont notamment les suivantes : Le rendement de l'infrastructure (groupe directeur de gestion, groupe d'exécution des processus, équipes d'exécution des processus...) a-t-il été adéquat ? Les méthodes employées par les équipes d'exécution des processus au cours de leurs activités d'élaboration des solutions ont-elles été satisfaisantes ? Les activités de communication du programme ont-elles été suffisantes ? Le parrainage doit-il être réaffirmé ? Faut-il accomplir une autre activité de référence ? Le point de réinsertion dans le modèle IDEALSM au cours du prochain cycle dépend étroitement des réponses données à des questions comme celles-ci.

6.4.10.6 SCAMPI : "Standard CMMI Appraisal Method for Process Improvement"

Présentation de SCAMPI

SCAMPI, Standard CMMI Appraisal Method for Process Improvement, est une méthode qui utilise le CMMI comme référentiel pour identifier les forces et faiblesses du processus logiciel et/ou système d'une entreprise ou d'un organisme. Cette méthode s'appuie sur une discipline et des règles rigoureuses qui assure l'exhaustivité et l'objectivité de l'évaluation [10]. Elle se caractérise par la large place faite au consensus (c'est une équipe qui accomplit l'évaluation) et à l'implication de l'entité évaluée dans sa propre évaluation. Le SEI a développé cette méthode, gère un programme d'accréditation de chefs évaluateurs et diffuse périodiquement des statistiques.

Bien que conçue pour pouvoir évaluer des projets d'envergure, la méthode SCAMPI n'est pas réservée qu'aux grandes entreprises ou aux grands organismes. Plusieurs variations définies dans la méthode permettent de moduler un SCAMPI en accord avec chaque profil d'entreprise ou organisme à évaluer.

Toutefois, il y a des limites importantes à la capacité d'appliquer la méthode SCAMPI dans de petites entités (moins de 20 personnes dans l'unité évaluée). Dans ce cas, il faut se tourner vers d'autres méthodes d'évaluation, mieux adaptées aux petites entités, mais qui peuvent néanmoins utiliser le CMMI comme référentiel.

Présentation SCAMPI Class C, Class B et Class A

Il existe trois types de SCAMPI : SCAMPI Class C, Class B et Class A (comme le montre la figure 6.28). La distinction principale entre ces trois évaluations est leur centre d'application :

- SCAMPI C est une évaluation d'"Approche" ;
- SCAMPI B est une évaluation de "Déploiement" ;
- SCAMPI A est une évaluation d'"Institutionnalisation".



Figure 6.28 – SCAMPI Class C, Class B, Class A

Seule l'évaluation SCAMPI Class A pourra déterminer officiellement le niveau de maturité de l'entreprise. Elle est généralement précédée d'une ou plusieurs évaluation SCAMPI Class B et/ou C.

SCAMPI Class C L'évaluation initiale SCAMPI Class C mesure la cohérence du référentiel de l'organisation avec les exigences du modèle CMMI. Les objectifs d'évaluation Class C :

- fournir une formation sur CMMI et SCAMPI ;
- s'assurer d'une interprétation appropriée des pratiques ;
- fournir une compréhension commune du référentiel de l'entreprise ;
- identifier les forces et les faiblesses ;
- aider le SEPG à planifier des actions d'amélioration de processus ;
- aider la direction à identifier les objectifs d'organisation ;
- fournir les informations stratégiques pour des activités d'amélioration de processus ;
- Identifier ou valider les membres clefs pour l'amélioration de processus.

SCAMPI Class B L'évaluation SCAMPI Class B analyse la profondeur du déploiement des pratiques du référentiel. Elle permet de :

- examiner l'adéquation entre l'approche sélectionnée et le contexte :
 - comprendre les variations entre les pratiques actuelles ;
 - déterminer les membres qui vont guider la mise en œuvre ;

- identifier les corrections nécessaires.
- comprendre les diverses mises en œuvre du référentiel :
 - impliquer et ouvrir le dialogue avec les équipes techniques sur ce qui fonctionne ;
 - examiner les " direct artefacts " résultant de la mise en œuvre ;
 - chercher des éléments d'entrée pour mettre à jour l'approche.
- s'assurer d'avoir centré l'approche avec les objectifs business :
 - valider le plus d'éléments possible du modèle ;
 - réfléchir aux éléments restant du modèle.
- préparer l'évaluation de Class A.

SCAMPI Class A Les entreprises souhaitant faire reconnaître et communiquer leur niveau de maturité CMMI doivent soumettre leurs processus à une évaluation SCAMPI Class A, seule évaluation déterminant le niveau de maturité de l'organisation.

Cette approche intègre des intervenants internes formés à l'évaluation de maturité organisationnelle. Les résultats définitifs indiquent la notation obtenue (niveau de maturité) ainsi que la description des forces et opportunités d'amélioration de chaque domaine de processus du périmètre évalué. La figure 6.29 montre les étapes de la préparation et de l'évaluation SCAMPI A.



Figure 6.29 – Les phases d'évaluation SCAMPI A

L'annexe B présente en détail le déroulement du SCAMPI A.

6.5 Conclusion

Tous ces modèles et normes ont leurs forces et leurs points faibles. Parmi les différents modèles, normes et corpus de connaissances présentés dans l'état de l'art, certains englobent toutes les activités d'une entreprise et adoptent une démarche descendante comme ISO 9001 d'autres comme CMMI et ITIL, sont des guides de bonnes pratiques et donc spécifiques à certaines activités adoptant une approche ascendante.

Il serait donc intéressant de combiner ou tout du moins de rapprocher plusieurs modèles, normes et corpus de connaissance afin de voir apparaître les différentes synergies : d'adopter une démarche multi-modèle afin de bénéficier des apports de plusieurs standards.

L'interopérabilité

7.1 Introduction

"L'interopérabilité est le fait que plusieurs systèmes, qu'ils soient identiques ou radicalement différents, puissent communiquer sans ambiguïté et opérer ensemble. Compte tenu du fait que ces éléments sont produits par des constructeurs divers, avec des méthodes variées, et qu'ils répondent à des besoins spécifiques, l'idée la plus simple consiste à définir une base explicite, une norme ou un ensemble de normes, que chaque élément va 'implanter' dans son propre fonctionnement" [18]. Cette définition a été donnée par le groupe de travail TG2 d'INTEROP (Interoperability Research for Networked Enterprises Applications and Software). L'objectif principal de ce groupe est d'étudier les possibilités d'exécuter des transformations de modèle suivant une approche conduite par les modèles (model-driven approach) pour résoudre des problèmes d'interopérabilité. Le groupe TG2 précise bien que l'interopérabilité est " la possibilité pour différents systèmes de communiquer entre eux sans dépendre d'un acteur particulier ". Elle repose sur la définition d'un standard ouvert.

Donc l'interopérabilité est la capacité qu'ont deux systèmes à communiquer de façon non ambiguë, que ces systèmes soient similaires ou différents. L'interopérabilité n'est pas la compatibilité. Il est possible de rendre interopérables deux systèmes à l'origine incompatibles. On peut dire que rendre interopérable, c'est créer de la compatibilité [41].

7.2 Les principaux travaux en interopérabilité

Dans cette section, nous allons nous intéresser aux travaux des différents organismes et laboratoires de recherche dans le domaine de l'interopérabilité. Certes l'interopérabilité peut se faire à différents niveaux c'est pour cela que nous avons choisi de restreindre notre travail de recherche aux organismes et laboratoires qui traitent l'interopérabilité au niveau des systèmes d'information et nous accorderons plus d'importance à ceux qui traitent le problème d'interopérabilité relatif à la modélisation d'entreprise et aux processus.

7.2.1 Le groupe de travail TG2 d'INTEROP

"INTEROP est une association de 47 partenaires de 15 pays venant de différents secteurs : des institutions académiques, de centres de recherche, des industriels et des communautés de standardisation. Le réseau rassemble 204 chercheurs et 140 doctorants et est coordonné par l'université de Bordeaux 1" [60]. A partir de Mars 2007, l'"European virtual laboratory for Enterprise Interoperability" (INTEROP-VLab) a été lancé pour prendre le relais d'INTEROP pour la poursuite des recherches.

Pour INTEROP, l'interopérabilité est assurée si l'interaction prend place entre toutes les couches de l'entreprise : la couche métier, connaissances, sémantique et communication (ICT : information and communication technology). Dans ses travaux d'interopérabilité, TG2 du réseau INTEROP a publié trois livrables TG2.1, TG2.2 et TG2.3 concernant une étude cas du groupe Telco S.A (détaillant de produits électriques et de communications) que nous présenterons dans cette section.

Dans le premier livrable TG2.1 [17], le groupe TG2 a travaillé pour faire communiquer deux modèles d'entreprise l'un en GRAI et l'autre en UML au niveau CIM : Computation Independent Model (niveau élevé d'abstraction par rapport aux détails des technologies : politique de l'entreprise, règles de gestion, processus métier et terminologie).

La figure 7.1 montre comment peut se faire la transformation entre ces deux langages à un niveau élevé c'est-à-dire au niveau des diagrammes :

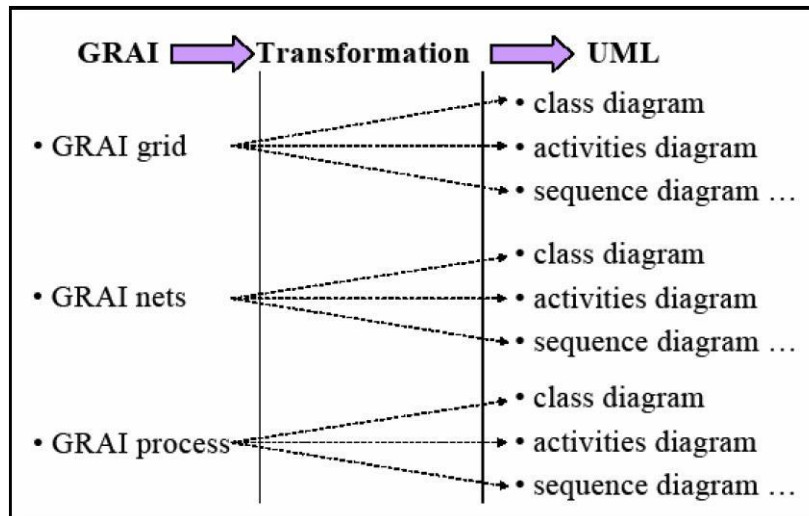


Figure 7.1 – Transformations de GRAI à UML

Deux possibilités ont été retenues pour faire cette transformation :

- la première consiste à travailler sur l'interopérabilité au niveau de deux méta-modèles relatifs à GRAI et à UML ;
- la deuxième consiste à travailler sur l'interopérabilité au niveau d'un méta-modèle commun à GRAI et UML.

La figure 7.2 montre respectivement ces deux solutions :

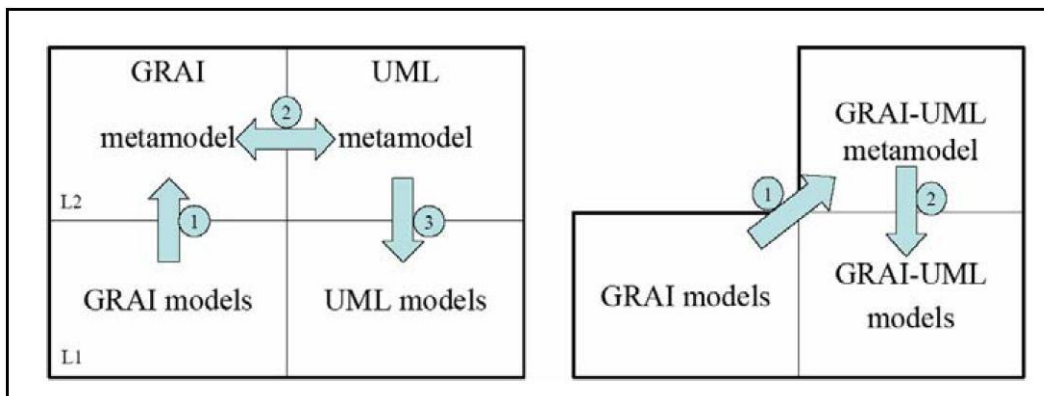


Figure 7.2 – Les deux solutions retenues pour l'interopérabilité entre les méta-modèles de GRAI et UML

Dans le livrable TG2.2 [18], le groupe TG2 a travaillé pour faire communiquer un progiciel de gestion intégrée (ERP : Enterprise Resource Planning) et un gestionnaire de la relation client (CRM : Customer Relation Management) de deux organismes. La figure 7.3 montre les

deux processus métier de ces deux organismes en interaction :

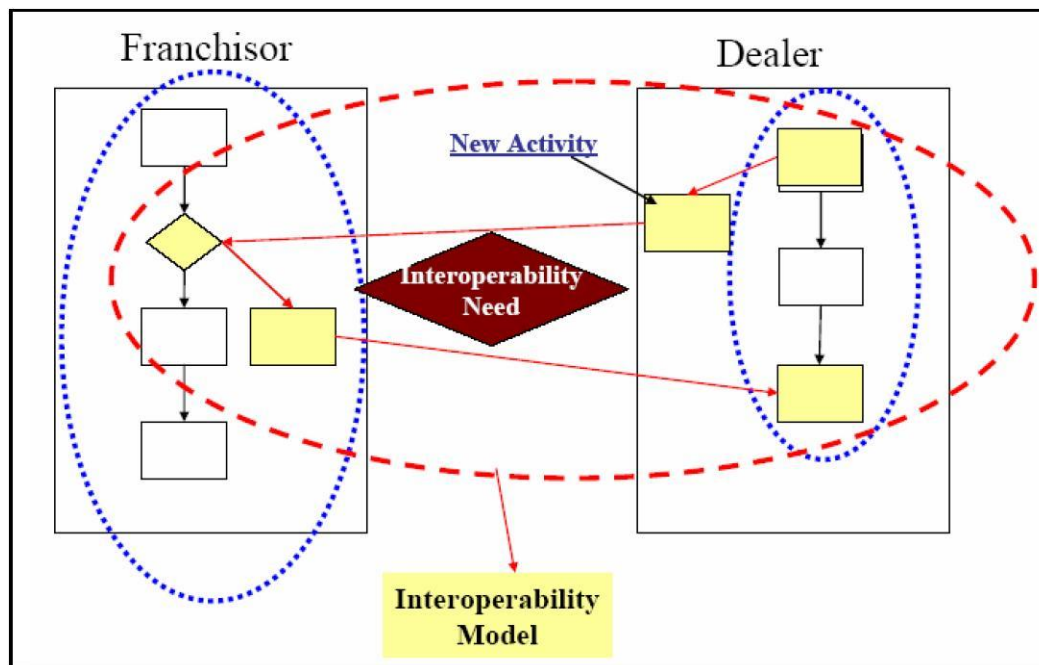


Figure 7.3 – Le besoin d'interopérabilité entre les modèles de deux entreprises

"L'approche dirigée par les modèles permet de fournir beaucoup d'avantages pour améliorer l'interopérabilité. C'est dans ce contexte que l'architecture dirigée par les modèles (MDA : Model Driven Architecture), adoptée par l'OMG (Object Management Group) en 2001 était destinée à promouvoir l'utilisation des modèles comme une manière fondamentale de conception et de développement de différents types de systèmes en assurant la transformation de modèles d'une manière la plus automatique possible. Par conséquent, cette architecture définit une hiérarchie des modèles à partir de trois points de vue : Computation Independent Model (CIM), the Platform Independent Model (PIM), and the Platform Specific Model (PSM)" [83].

Computation Independent Model (CIM) : "représente les besoins du système dans l'environnement dans lequel il va opérer concernant les modèles métier avec un point de vue indépendant de toutes les caractéristiques techniques des technologies" ;

Platform Independent Model (PIM) : "sert à modéliser les fonctionnalités du système sans définir ni comment ni sur quelle plate forme ce dernier va être implémenté" ;

Platform Specific Model (PSM) : "c'est la transformation du PIM avec une plate forme spécifique basée sur une technologie bien définie" [17].

Le TG2 adopte cette philosophie de hiérarchie des modèles. La figure 7.4 montre l'interopérabilité à différents niveaux de modélisations : du moins spécifique (CIM) jusqu'à celui qui donne les détails de la technologie (PSM) :

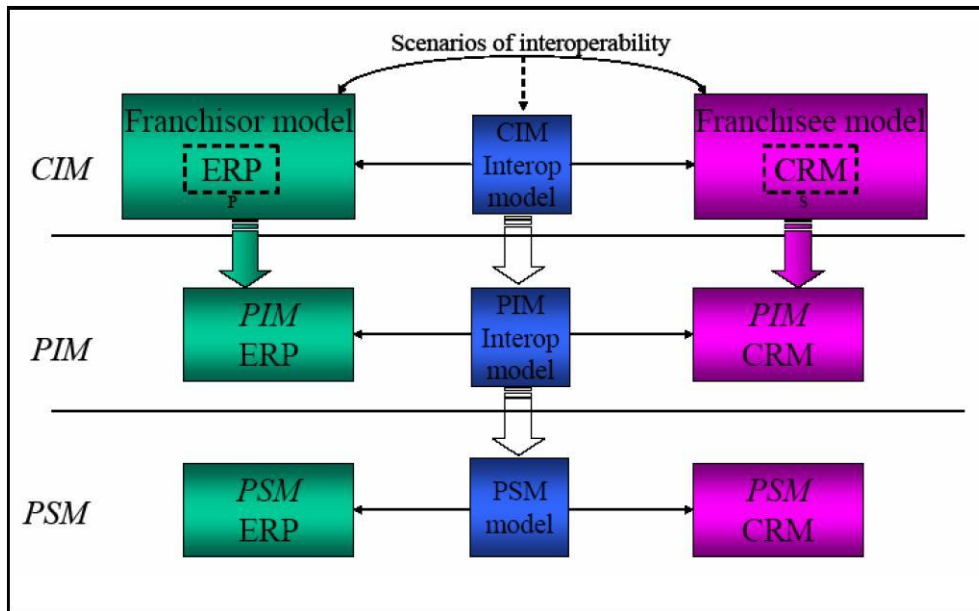


Figure 7.4 – L'interopérabilité à différents niveaux de modélisation

Dans la première partie du TG2.3 [19], le groupe de travail décrit une méthode qui collecte toutes les expériences du TG2 afin de comprendre comment l'approche dirigée par les modèles (model-driven approach) peut supporter l'interopérabilité. Ce travail repose sur la synthèse du cas d'étude fourni par Singular. Le résultat principal est la formalisation d'une méthode qui peut être suivie par deux entreprises qui veulent mettre en place une interopérabilité entre leurs systèmes d'informations. Cette méthode repose sur trois principes :

- 1) commencer à mettre en place l'interopérabilité à haut niveau, c'est à dire au niveau modélisation d'entreprise ;
- 2) mettre en place l'interopérabilité au niveau le plus bas au moyen de transformations successives des modèles réalisées à différents niveaux d'abstraction suivant une approche MDA (Model Driven Architecture) ;
- 3) résoudre les éventuels problèmes d'interopérabilité, qui peuvent apparaître au cours de ce processus, et les transformations de modèles par un support sémantique.

La deuxième partie de ce livrable se focalise sur les principaux résultats obtenus par rapport au cas d'étude (Singular). Le livrable TG2.3 s'achève en fournissant la méthode MDI dans le but de proposer une démarche à adopter par les entreprises souhaitant mettre en place

une interopérabilité non seulement au niveau technique. La figure 7.5 montre le modèle de référence MDI appliqué au cas d'étude.

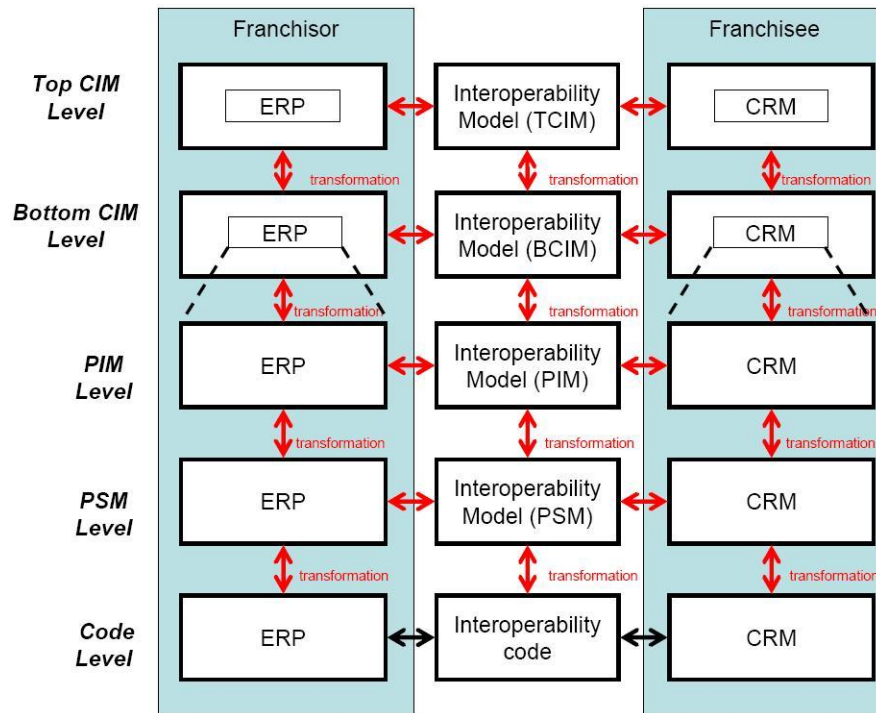


Figure 7.5 – Le modèle de référence de MDI appliqué au cas d'étude

7.2.2 Le projet ATHENA & UEML

"UEML (Unified Enterprise Modelling Language) est un langage de processus d'entreprise, développé dans le cadre du projet européen ATHENA, supportant l'interopérabilité des langages de modélisation, des outils et des applications. Il permet d'aider la communauté de la modélisation d'entreprise avec des mécanismes de partage de modèle d'entreprise standards accompagnés d'un support pour l'implémentation des solutions open source. Le but de ce langage est de jouer le rôle d'une passerelle entre les différents langages et méthodologies de modélisation d'entreprise" [11].

Les initiatives autour d'UEML s'orientent aujourd'hui vers un consensus entre les différents acteurs et les différents organismes de normalisation travaillant dans le domaine de l'ingénierie et de l'intégration d'entreprise. Ce consensus devrait se matérialiser par le développement d'un langage pivot de haut niveau permettant à plusieurs modèles d'échanger des informations, des données et des connaissances contenues et déjà prises en compte au sein de leurs

propres modèles établis au moyen de formalismes et de méthodes différentes : UML, IDEF, CIMOSA, GRAI. . .

Le développement d'UEML contribuera à une définition claire de la sémantique commune des formalismes et une meilleure délimitation du domaine de la modélisation et de l'ingénierie d'entreprise ; une meilleure interopérabilité et capacité de communication entre acteurs de la modélisation en environnement hétérogène ; une meilleure définition du corpus scientifique de la modélisation et de l'ingénierie d'entreprise et donc une augmentation de la visibilité de ces dernières dans le cadre de la communauté scientifique et finalement un vocabulaire accepté et utilisé par les organismes de normalisation à tous les niveaux (national, européen et international) travaillant dans le domaine [105].

7.2.3 OASIS

OASIS (Organization for the Advancement of Structured Information Standards) est un consortium international à but non lucratif qui conduit le développement et l'adoption des standards e-business. Ce consortium produit la plupart des standards Web services. Créé en 1993, OASIS plus de 5000 participants représentant plus de 600 organisations et de membres individuels dans 100 pays).

OASIS travaille sur le langage XML pour permettre l'interopérabilité et considère que ce langage peut permettre l'échange d'informations entre tous les langages vu sa grande généricité. OASIS fournit un Framework pour l'interopérabilité basé sur le langage XML ce qui est critique et décisionnel pour le futur succès des métiers d'électronique. OASIS laisse le contrôle des spécifications XML entre les mains des utilisateurs [51].

7.2.4 OWL et l'interopérabilité

Bien que des ontologies soient largement répandues pour résoudre quelques problèmes spécifiques d'interopérabilité, il n'y a aucune ontologie spécifique permettant de définir clairement ce qu'est l'interopérabilité actuellement, indépendamment de n'importe quel domaine. L'article de Naudet [82] propose et discute une première version d'une telle ontologie, à savoir l'ontologie de l'interopérabilité (OoI : Ontology of Interoperability), formalisée en utilisant le langage Ontology Web Language (OWL).

A part cette version de OWL, il existe d'autres modèles traitant l'interopérabilité comme par exemple les niveaux d'interopérabilité de système d'information (LISI : Level of Information System of Interoperability), et les modèles de Morphisme (MoMo : Model Morphism), qui concernent l'interopérabilité des modèles.

OoI (Ontology of Interoperability) pourrait être employé pour représenter les problèmes d'interopérabilité couverts par l'ontologie des modèles de Morphisme. Le Modèle Morphisme (MoMo) essaie de résoudre une partie du problème d'interopérabilité à savoir les capacités d'interopérabilité entre les modèles. Par conséquent, l'OoI peut être employé pour modéliser une interopérabilité entre deux normes et le MoMo peut être lié avec ce travail en tant qu'une solution possible à ce problème.

7.3 Conclusion

Dans ce chapitre, nous avons présenté les principaux travaux de recherche pour résoudre le problème d'interopérabilité. Ces travaux visent principalement à assurer l'automatisation des transferts d'informations entre deux systèmes distincts en passant par la création de passerelles entre des langages de modélisation différents.

Conclusion de la partie

Dans cette partie, nous avons présenté cinq chapitres qui correspondent à une vision, que nous pensons globale et couvrante, des éléments de connaissance nécessaires pour l'étude du concept d'intégration des processus métier et de la mise en place d'un référentiel qualité multi-vues.

Le premier chapitre s'est intéressé à la modélisation d'entreprise. Nous avons présenté les différentes approches de modélisation de l'entreprise et nous avons conclu par les principaux "frameworks" de modélisation d'entreprise. Ce chapitre nous a permis d'aborder les notions relatives à la modélisation d'entreprise et les différents travaux existants qui ont permis de mettre en évidence les apports des approches systémiques. Dans nos travaux, nous privilégierons ce type d'approche afin de traiter les processus organisationnels d'entreprise .

Le deuxième chapitre a présenté un panorama des langages et standards de modélisation d'entreprise. Ce chapitre nous a permis de comparer les langages existants. Devant la diversité, la richesse et la complémentarité de ces langages, notre méthodologie d'intégration constituera un ensemble de recommandations pour la mise en place des processus et sera indépendante du langage de modélisation. Pour le cas d'application présenté dans le chapitre 11, on utilisera un formalisme de modélisation simplifié afin de permettre toute transformation vers un autre langage.

Le Business Process Management (BPM), véritable standard d'organisation d'entreprise et de gestion des processus métier, est présenté dans le troisième chapitre en exposant successivement ses principes et ses différents standards et outils. Ce chapitre nous a permis d'aborder les concepts fondamentaux relatifs aux processus afin de les cartographier et de réaliser un référentiel qualité représentatif des processus et procédures de l'entreprise.

Le quatrième chapitre a présenté la notion de qualité en entreprise, coeur de nos travaux, et

a exposé les principaux standards et normes qualité. Ce chapitre nous a permis de mettre en évidence la nécessité d'intégrer les notions de qualité dans un référentiel de processus. En plus, la comparaison des différents standards et normes et qualité montre la complémentarité qui peut apparaître lorsqu'on veut intégrer les recommandations de différents standards et normes dans un seul référentiel.

Enfin, le cinquième chapitre a présenté un panorama des principaux travaux, solutions et standards d'interopérabilité. En effet, l'objectif de nos travaux est l'intégration des processus métier et la problématique d'interopérabilité apparaît naturellement. Dans ce domaine, il s'agit plus particulièrement pour nous de mettre en correspondance des organisations et la modélisation de leurs interactions et échanges.

Notre objectif dans les chapitres suivants est d'articuler tous ces concepts afin de répondre à notre problématique.

QUATRIÈME PARTIE

APPORT SCIENTIFIQUE

Référentiel qualité multi-vues

9.1 Introduction

L'objectif de notre thèse est de proposer une méthode basée sur la modélisation d'entreprise pour organiser une entreprise en conformité avec les recommandations des standards et normes qualité. Ce chapitre présente comment intégrer plusieurs processus à l'aide d'un référentiel commun offrant différents points de vue. Cette démarche est appliquée à l'intégration de deux standards qualité afin de générer un référentiel qualité multi-vues permettant une certification relative aux deux normes. Un cas d'application est présenté pour l'intégration des deux standards qualité ISO 9001v2000 et CMMI. Ce référentiel prend en compte la structure et les chapitres imposés par ISO et les recommandations de CMMI.

En partant des deux normes retenues par la direction de l'entreprise, il serait donc intéressant de combiner ou tout du moins de les rapprocher, afin de voir apparaître les différentes synergies : **de réaliser une cartographie multi-vues des processus**. Une telle cartographie permettra la mise en place et l'implémentation des deux normes qualité et l'obtention des certifications relatives.

Comme le montre la figure 9.1, l'objectif est de réaliser un référentiel qui :

- intègre les recommandations des différentes normes qualité ;
- permet l'évaluation du respect des différentes normes ;
- soit facilement utilisable par tous les collaborateurs ;
- soit facilement exploitable dans une boucle d'amélioration.

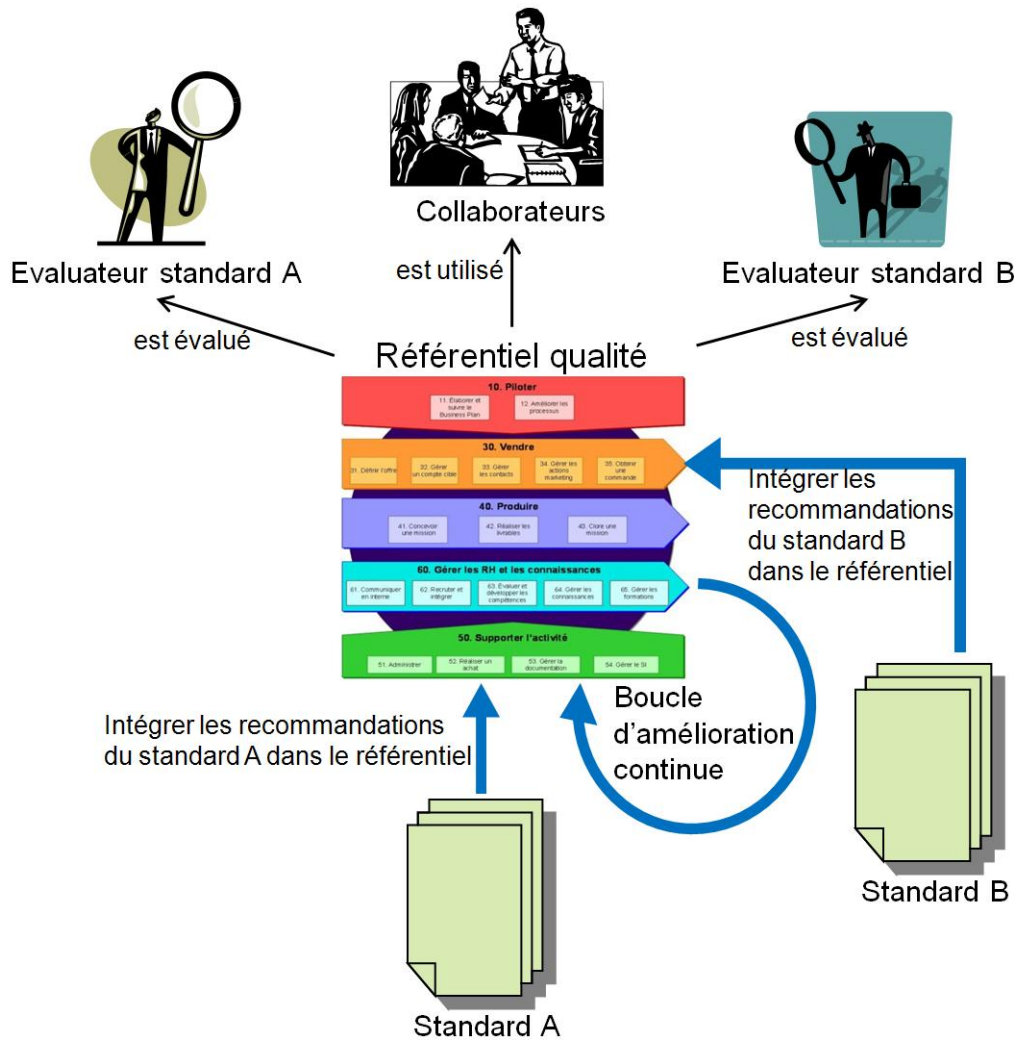


Figure 9.1 – Les objectifs du référentiel qualité

Notre démarche sera générique et permettra à n'importe quelle entreprise souhaitant intégrer plusieurs standards qualité de le faire. On présentera aussi un ensemble de référentiel (référentiel documentaire, référentiel des compétences et méthodologie d'audit) permettant de s'assurer de l'implémentation et du respect de notre référentiel qualité au sein des projets [44] [45] [47] [43] [46].

9.2 Le modèle proposé

9.2.1 La description du modèle

Chaque modèle, norme et corpus de connaissances décrit une partie des activités de l'entreprise avec un niveau de précision et un niveau de spécificité qui lui est propre. Ils peuvent décrire aussi des activités non présentes au sein de l'entreprise et omettre certaines activités quant à elles bien présentes (comme le montre la figure 9.2). George Box disait "Tous les modèles sont faux! Certains sont utiles". Cette citation reprend bien l'idée que, par définition, un modèle ne peut représenter de manière intégrale la réalité de l'entreprise. Sa description s'en approche avec un niveau de précision plus ou moins grand suivant le modèle.

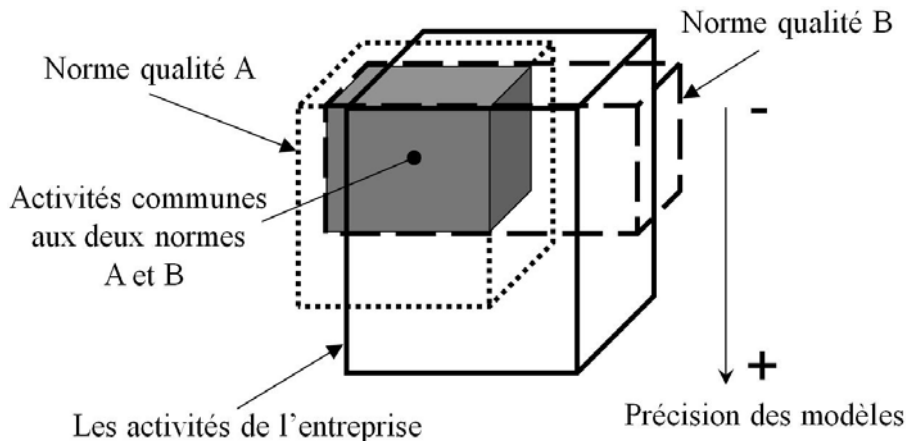


Figure 9.2 – Le champ de couverture des normes qualité des activités de l'entreprise

Généralement, on retrouve au sein de deux modèles la description d'activités communes et des activités spécifiques à chaque modèle (comme le montre la figure 9.3). Ces deux types d'activités peuvent être ou ne pas être en place au sein de l'entreprise.

En se basant sur ces constatations, nous avons défini une démarche en quatre étapes (comme le montre la figure 9.4). Cette démarche permet de rapprocher et de créer un seul modèle intégré en partant de deux modèles ou standards différents.

La sous-section suivante présente en détail les étapes de mise en place d'un référentiel intégré.

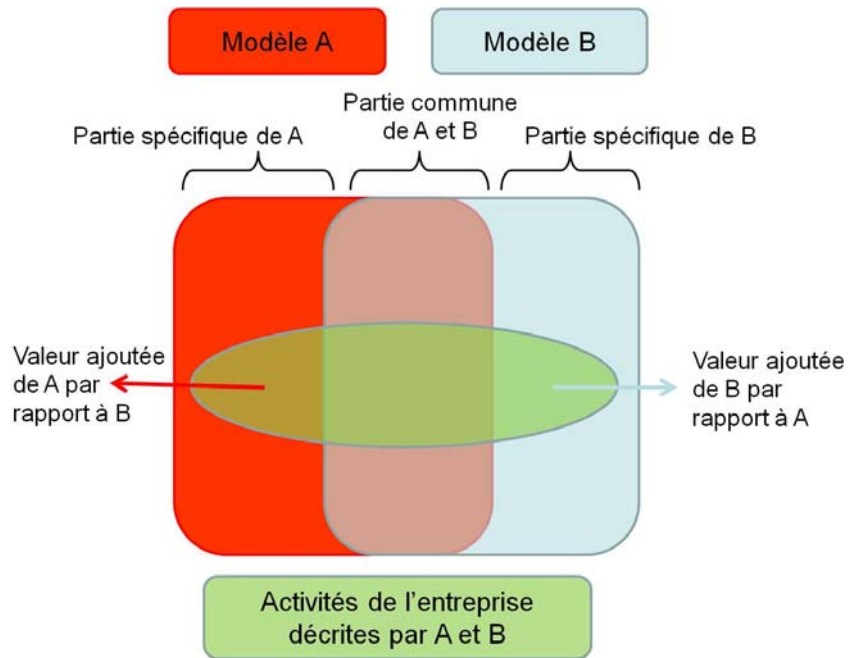


Figure 9.3 – Le champ de couverture commun des normes qualité des activités de l'entreprise

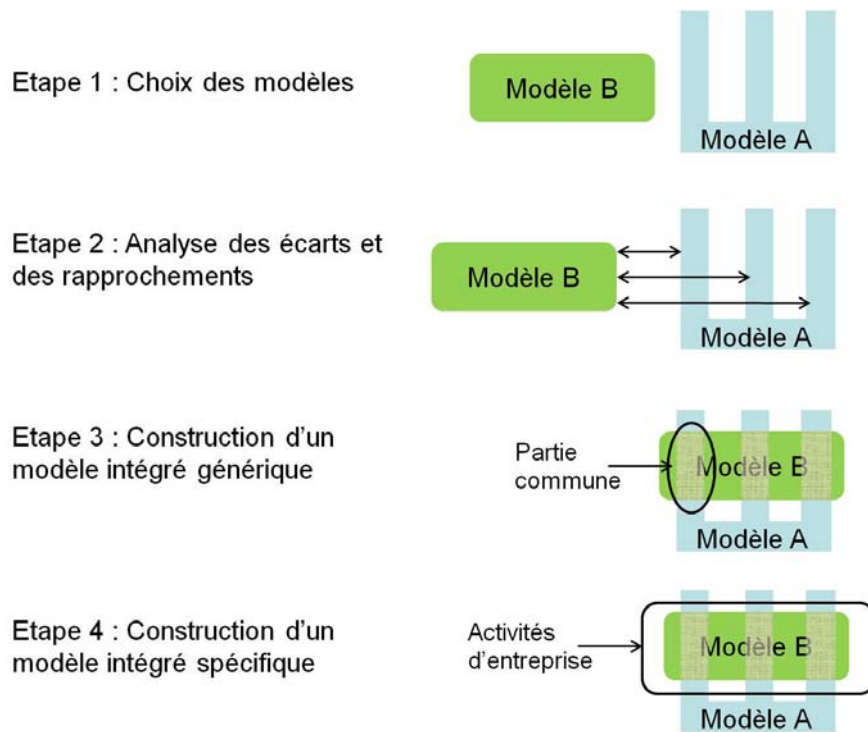


Figure 9.4 – L'approche multi-modèles proposée

9.2.2 Les étapes de mise en place d'un référentiel intégré

9.2.2.1 L'étape 1 : Choix des modèles

Cette étape consiste en un ensemble de questions que toute entreprise souhaitant implémenter le modèle doit se poser. La liste des questions est la suivante :

- Quel est l'état des lieux? Méthodes, outils, organisation, acteurs...
- quels sont les objectifs et les requis? Augmenter la satisfaction client, la productivité...
- quels sont les modèles envisagés? CMMI, ISO 9001v2000, ITIL...
- quel est le budget?
- quelles sont les échéances? 6 mois, 1 ans, 2 ans...
- quelles sont les ressources disponibles? Jeunes, expérimentés, consultants, en nombre important, faible...

À partir des réponses à ces questions, on peut déterminer la cohérence entre les modèles envisagés et les besoins et ressources de l'entreprise et ainsi sélectionner les modèles à intégrer. Par exemple, une entreprise souhaitant adopter CMMI en six mois ne pourra pas le faire car l'implémentation d'un niveau de maturité dure en moyenne entre 18 et 24 mois [9].

9.2.2.2 L'étape 2 : Analyse d'écart et rapprochements

Une fois les modèles sélectionnés, les écarts et rapprochements sont analysés. Pour ce faire, nous réalisons **un mapping** entre les modèles. Il doit permettre de déterminer :

- les niveaux d'abstractions des modèles sélectionnés;
- les secteurs fonctionnels traités;
- pour chaque élément d'un modèle, sa relation avec les éléments de l'autre modèle;
- un niveau de corrélation, afin de qualifier chaque relation.

Nous avons défini une ontologie (comme le montre la figure 9.5). Chaque standard qualité est composé d'un ou plusieurs ensembles de recommandations. Ces derniers regroupent une ou plusieurs pratiques. La terminologie diffère d'un standard à un autre. Par exemple, CMMI est composé de domaines de processus (Process Area (PA)) composés eux-mêmes d'un ensemble de pratiques (pratiques génériques (GP) et pratiques spécifiques (SP)). Pour ISO 9001v2000, nous avons des chapitres puis des sous-chapitres où on trouve les instructions. Parfois, les standards qualité adoptent des niveaux de maturité qui regroupent un ensemble de recommandations déjà définies dans le standard. En se basant sur cette description, nous avons implémenté une classe d'association "Mapping" afin de définir la synergie entre deux pratiques avec un niveau de corrélation (fort, moyen ou faible). Cette classe "Mapping" per-

met de définir si les deux pratiques veulent atteindre le même but et avec quel niveau de conformité.

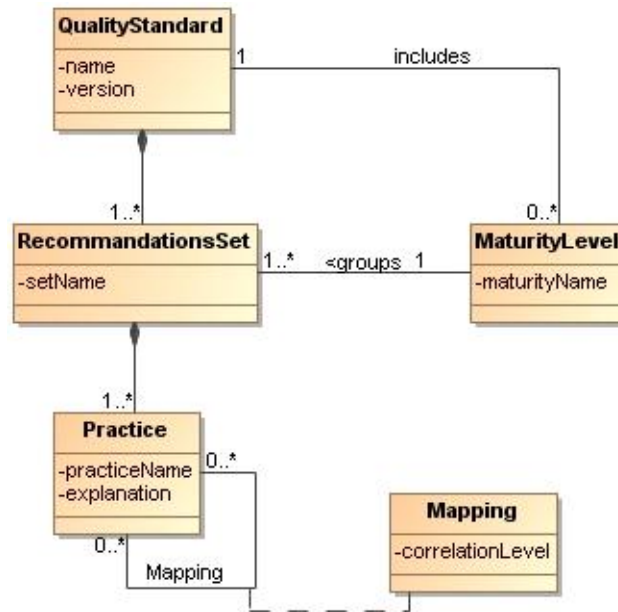


Figure 9.5 – Le diagramme de classe d’un standard qualité.

Notre ontologie nous permet de définir un mapping, comme le montre la figure 9.6. Ce mapping nous permet de parcourir toutes les recommandations d’un standard A et de trouver le (ou les) recommandation(s) d’un standard B qui lui correspondent et couvrent le même champ fonctionnel. Cette correspondance est spécifiée avec un niveau de corrélation. Ce dernier est à définir en fonction du contenu de chaque recommandation.

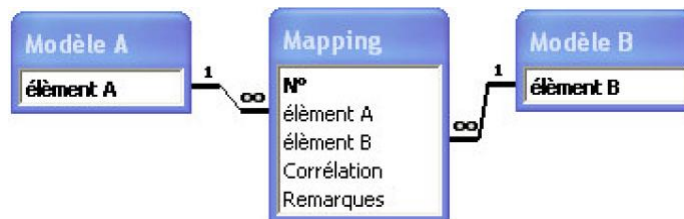


Figure 9.6 – Le mapping entre 2 modèles

9.2.2.3 L’étape 3 : Construction d’un modèle intégré

Cette étape permet de :

- résoudre toutes les relations de contradiction ;
- éviter le dédoublement du travail en consolidant les éléments avec des relations d'identité et d'inclusion ;
- maximiser le potentiel de synergies en combinant les éléments complémentaires.

Comme le montre la figure 9.7, à partir de cette étape, nous pouvons créer un modèle intégré théorique valable pour n'importe quelle entreprise répondant aux critères de deux modèles A et B. En effet, ce modèle théorique permet d'éviter le dédoublement de l'implémentation des recommandations communes aux deux modèles. Ainsi, en partant d'un modèle A et en se basant sur le mapping réalisé dans l'étape précédente, il suffit d'implémenter les recommandations du modèle B non prises en compte par A. Les recommandations non prises en compte peuvent être de deux types :

- aucune correspondance n'existe dans le mapping réalisé ;
- une (ou plusieurs) correspondance(s) existe(nt) mais avec des niveaux de corrélation moyen(s) ou faible(s).

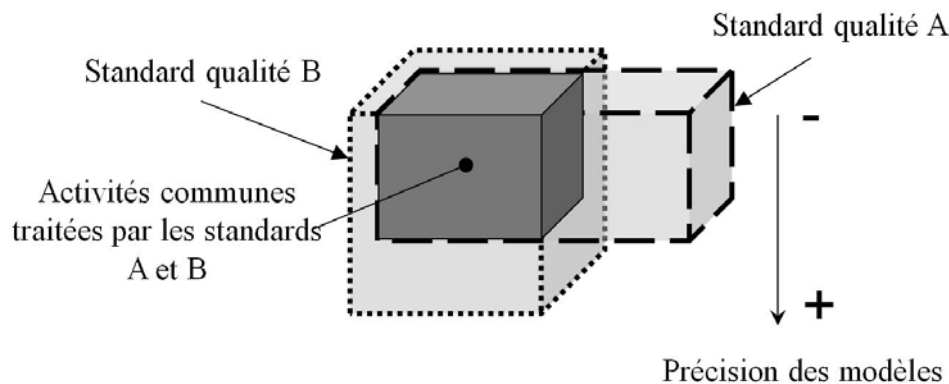


Figure 9.7 – Le modèle intégré

9.2.2.4 L'étape 4 : Adapter ce modèle au fonctionnement de l'entreprise : le rendre spécifique

En se basant sur le modèle générique réalisé à l'étape précédente, ne retenir que les éléments pertinents relativement à l'activité de l'entreprise et les adapter au contexte humain et culturel. Par exemple, le Process Area SAM (Supplier Agreement Management) de CMMI qui concerne la gestion des ententes avec les fournisseurs ne peut être implémenté si l'entreprise n'a pas de fournisseur. Donc il faudrait éliminer ce processus de notre modèle spécifique en justifiant cette action.

La figure 9.8 montre le croisement qui peut apparaître entre notre modèle générique et les

activités de l'entreprise.

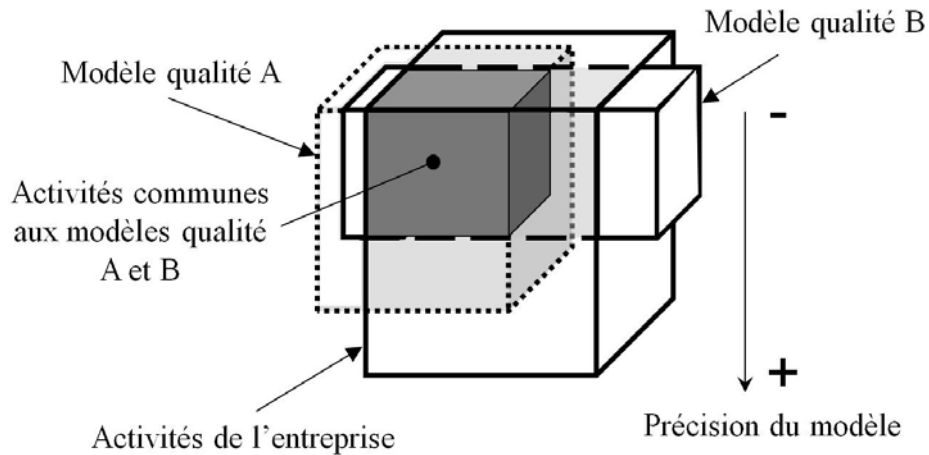


Figure 9.8 – Le modèle intégré spécifique

9.2.3 Cas d'application ISO 9001v2000 - CMMI

Cette sous-section présente l'application de notre approche multi-modèles aux deux standards qualité ISO 9001v2000 et CMMI.

9.2.3.1 Les synergies entre ISO 9001v2000 et CMMI

Le tableau 9.1, tiré en partie des travaux de Boris Mutafelija et Harvey Stromberg [81], présente une comparaison haut niveau d'ISO 9001v2000 et CMMI.

Tableau 9.1 – Comparaison haut niveau d'ISO 9001v2000 et CMMI

ISO 9001v2000	CMMI
Standard qualité certifiable	Modèle de bonnes pratiques certifiable grâce à SCAMPI (voir l'annexe B)
Large direction	Détaillé
Un ensemble de recommandations à satisfaire	Des niveaux de maturité progressifs
Aucune directive pour l'implémentation	Conseils d'institutionnalisation et d'implémentation
Couvre tous les secteurs d'une entreprise, parmi lesquels le développement logiciel et système	Applicable dans un contexte de développement de logiciels et de systèmes
Se limite à la définition des principes de gestion du système qualité	Identifie les pratiques opérationnelles à mettre en œuvre

Analopies Nous explorons les similarités entre les huit principes de management de la qualité d'ISO 9001v2000 et CMMI. A priori, ces principes de management de la qualité devraient correspondre aux pratiques génériques de CMMI puisque ces dernières fournissent une base pour l'institutionnalisation de processus. Voici le résultat de l'exploration :

- 1) L'orientation client : ce principe est traité par CMMI à travers la pratique générique GP 2.7 "Identifier et impliquer les parties prenantes" et la pratique spécifique SP 2.6 "Prévoir l'implication des parties prenantes identifiées" du domaine de processus "Planification du projet". Ce principe est traité aussi dans les Process Area "Développement Exigences" et "Solution Technique". Cependant, l'orientation client est plus fortement représentée dans ISO 9001v200 que dans CMMI ;
- 2) Le leadership : ce principe est couvert par de nombreuses pratiques génériques de CMMI : GP 2.1 "Établir et maintenir une directive organisationnelle", GP 2.4 "Assigner la responsabilité et l'autorité" et le GP 2.10 "Passer en revue avec la hiérarchie les activités, le statut et les résultats". En plus, le domaine de processus OPF "Focalisation de l'Organisation sur les Processus" couvre des aspects de leadership ;
- 3) L'implication du personnel : ce principe est couvert par CMMI à travers l'implémentation des pratiques génériques GP 2.3 "Fournir les ressources adéquates", GP 2.5 "Former selon les besoins les personnes" et GP 2.7 "Identifier et impliquer les parties prenantes concernées" ;
- 4) L'approche processus : ce principe est amplement couvert par les pratiques génériques GP 2.2 "Établir et maintenir le plan" et le GP 3.1 "Établir et maintenir la description

d'un processus". L'approche processus est explicitement supportée par les domaines de processus OPD "Définition des Processus de l'Organisation" et IPM "Gestion de Projet Intégré" et implicitement par le reste des domaines de processus ;

- 5) La gestion par approche système : ce principe est explicitement couvert par le GP 3.1 "Établir et maintenir la description d'un processus" ainsi que tous les domaines de processus ;
- 6) L'amélioration continue : CMMI est un modèle de bonnes pratiques pour l'amélioration continue. Donc, tout CMMI, spécialement avec ses niveaux de maturité, fournit une base pour réaliser ce principe ;
- 7) L'approche factuelle pour la prise de décision : CMMI supporte ce principe à travers les pratiques génériques GP 2.8 "Surveiller et contrôler le processus" et à travers de nombreux domaines de processus et spécialement les domaines de processus PMC "Maîtrise et suivi de projet", MA "Analyses et mesures", IPM "Gestion de Projet Intégré" et DAR "Résolution et Analyse pour Décision" ;
- 8) Les relations mutuellement bénéficiaires avec les fournisseurs : CMMI couvre la relation avec les fournisseurs spécialement à travers le domaine de processus SAM "Gestion des ententes avec les fournisseurs", du point de vue commande plutôt que du point de vue collaboration.

Différences La différence majeure entre ces deux standards réside dans leurs langages. Tandis que les recommandations d'ISO 9001v2000 sont clairement prescriptives, CMMI ne liste pas ses recommandations en utilisant l'impératif. Par exemple, ISO 9001v2000 spécifie ses recommandations pour le QMS (Quality Management System = Système de Management de la Qualité) sous la forme "L'organisation doit identifier les processus requis pour le QMS..." alors que la description de la pratique spécifique correspondante de CMMI SP 1.1 (du domaine de processus OPD "Définition des Processus de l'Organisation") est "Établir et maintenir l'ensemble de processus normalisés de l'organisation" puis continue en énumérant neuf sous-pratiques décrivant les détails requis pour implémenter avec succès cette pratique. Une autre différence majeure réside dans la compacité du langage d'ISO 9001v2000 qui utilise des expressions comme "établir et maintenir" ou "déterminer fournir". Par exemple, pour ISO 9001v2000 la recommandation "L'organisation doit déterminer et fournir..." définit deux actions différentes : d'abord déterminer les ressources puis fournir ces mêmes ressources. Pour CMMI, cette recommandation correspond au domaine de processus PP "Planification du Projet" pour la détermination des ressources ('déterminer') puis à la pratique générique GP 2.3 "Fournir les ressources adéquates" de tous les domaines de processus pour assurer que les

ressources sont disponibles ('fournir').

À cause de leur différence de cibles et de buts, la quantité de détails présentée par les deux standards est différente. D'un côté, CMMI couvre tous les détails nécessaires pour le développement des systèmes complexes. De l'autre côté, ISO 9001v2000 décrit simplement un ensemble de recommandations nécessaires pour développer des produits de haute qualité et satisfaire les exigences du client. Les détails pour répondre à ces exigences sont laissés à l'utilisateur, mais pour être conforme à ISO 9001v2000, toutes exigences doivent être satisfaites.

9.2.3.2 La possibilité de combiner ISO 9001v2000 et CMMI

En se basant sur la sous-section précédente, *le modèle CMMI et la norme ISO 9001v2000 sont non seulement compatibles, mais aussi complémentaires* [107].

Le modèle CMMI et le standard ISO 9001v2000, sont tous deux basés sur une approche processus. Cependant, du fait de leur historique différent, quelques différences majeures doivent être relevées :

- 1) Le modèle CMMI est applicable dans un contexte de développement d'applications et de systèmes logiciels, alors que le domaine d'application du standard ISO 9001 couvre tous les secteurs d'une entreprise, parmi lesquels le développement logiciel. Par contre, le modèle CMMI identifie les pratiques opérationnelles à mettre en œuvre dans son domaine d'application, contrairement au standard ISO 9001 où l'on se limite à la définition des principes de gestion du système qualité.
- 2) Les pratiques de gestion de système qualité au niveau société telles que requises par le standard ISO 9001 ne sont reprises dans le modèle CMMI qu'au niveau 3. Ceci s'explique par le fait que, dans le modèle CMMI, la mise en œuvre opérationnelle des pratiques de développement sur les projets prime sur la définition d'un système qualité applicable à tous les développements de l'organisation.
- 3) Du fait de son domaine d'application plus large, le standard ISO 9001 reprend des pratiques annexes aux développements de logiciel, mais "oubliées" par le CMMI. Notons par exemple le manque d'attention par rapport aux clients, aux équipements d'infrastructure et au contrôle des équipements de mesure.

Ceci étant, la certification ISO 9001v2000 d'une entreprise ayant atteint le niveau 3 CMMI engendrera moins d'efforts et aucune modification majeure de son système qualité; en effet, les pratiques opérationnelles préconisées par le CMMI répondent en grande partie à l'ISO 9001v2000 en matière d'amélioration continue des processus et de définition d'objectifs en matière de qualité. Par ailleurs, les acquis du programme d'amélioration basé sur le CMMI ne seront nullement remis en question par la certification ISO 9001v2000.

En dépit de priorités différentes (le standard ISO 9001 requiert d’abord un système qualité défini au niveau de l’entreprise, alors que le modèle CMMI se concentre d’abord sur la mise en œuvre des bonnes pratiques au niveau des projets de développement) les pratiques de développement logiciel du modèle CMMI des niveaux 2 et 3 constituent une base idéale pour la certification ISO 9001v2000 : l’effort requis pour la certification des entreprises ayant atteint le niveau 3 CMMI restera réduit.

9.2.3.3 Traduction des terminologies

Avant de réaliser le mapping entre ISO 9001v2000 et CMMI, nous nous sommes concentrés sur la réalisation d’une traduction entre les terminologies des deux standards afin de remédier à la différence majeure qui existe entre les deux standards. Le tableau 9.2, réalisé en se basant sur les travaux de Richard Basque [9] [10], montre le résultat de cette traduction.

Tableau 9.2 – Traduction des terminologies

ISO 9001v2000	CMMI
Top management	Management haut-niveau ; Management senior
Système de management de la qualité (QMS), Manuel qualité	Ensemble des processus standards de l’entreprise
Plan qualité	Plan projet, Plan de développement logiciel, Plan de gestion de l’ingénierie système, Plan de gestion des données
Client, Partie intéressée	Client, Partie prenante concernée
Procédure documentée	Plan pour la réalisation du processus, procédure
Enregistrement	Produit de travail, preuve d’implémentation

On remarque, que très souvent, à un terme d’ISO 9001v2000 correspondent plusieurs termes de CMMI. Ceci est dû à la compacité de la terminologie d’ISO 9001v2000, comme mentionné dans les différences entre les deux standards ci-dessus.

9.2.3.4 Implémentation de l'ontologie pour la réalisation du mapping ISO 9001v2000 - CMMI

En nous basant sur l'ontologie présentée dans la figure 9.5 (voir la sous-section 9.2.2.2), nous avons réalisé un diagramme de classe pour représenter le mapping entre ISO 9001v2000 et CMMI. La figure 9.9 présente une partie de ce mapping. Ce diagramme montre une relation avec un niveau de corrélation "fort" entre :

- La pratique spécifique SP 2.1 "Recueillir et analyser les écarts et déterminer les actions correctives nécessaires pour les traiter" du domaine de processus PMC "Maîtrise et suivi de projet" qui fait partie du niveau de maturité 2 de CMMI ;
- Le sous-chapitre "8.5.2-Actions correctives" faisant partie de la section 8 "Mesure d'analyse et d'amélioration continue" des chapitres d'ISO 9001v2000.

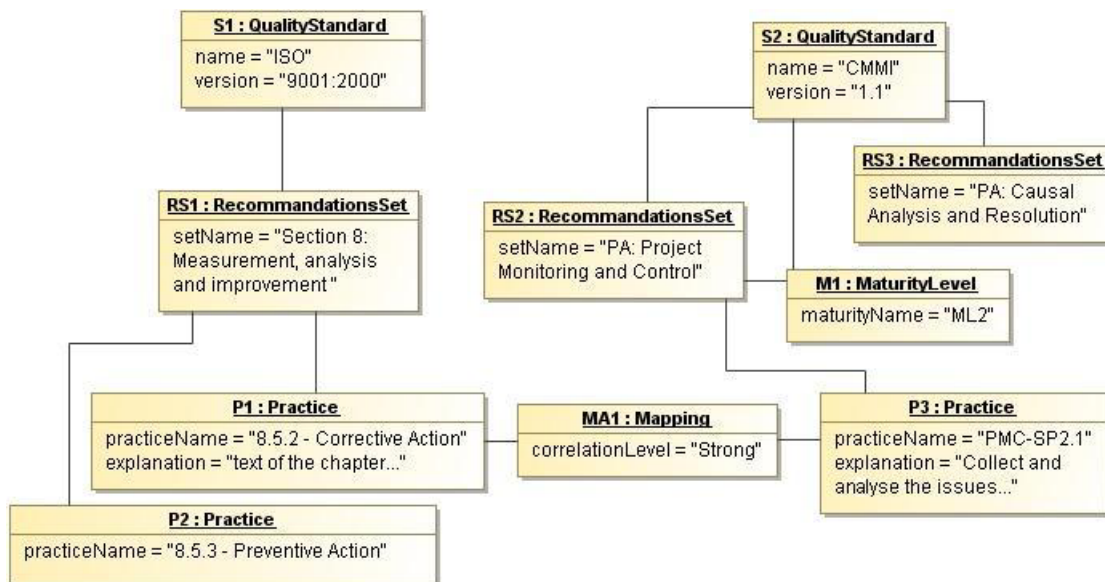


Figure 9.9 – Le diagramme d'objet des standards qualité ISO 9001v2000 et CMMI

9.2.3.5 Le mapping ISO 9001v2000 - CMMI

Il est toujours difficile de déterminer la granularité appropriée pour réaliser un "mapping" entre deux modèles. D'un côté, un "mapping" haut niveau peut ne pas fournir assez d'informations sur les similitudes et les différences. D'un autre côté, un "mapping" de très bas niveau peut engendrer un nombre très important de connexions entre les deux modèles rendant la correspondance très complexe.

Le "mapping" que nous avons réalisé est un compromis entre les deux niveaux (très haut

niveau et très bas niveau). On a effectué un "mapping" de chaque chapitre d'ISO 9001v2000 vers une pratique (ou plusieurs) de CMMI et vice versa. Le "mapping" ainsi réalisé sert comme indicateur de correspondance permettant une implémentation des deux standards.

Un tel travail a été initié en 2003 par Boris Mutafelija et Harvey Stromberg [81]. Notre "mapping" est basé sur l'interprétation des recommandations des deux standards faite par le laboratoire et Sylis (présenté plus en détail dans la section 11.2. Le résultat de notre "mapping" est présenté en détail dans l'annexe C.

9.2.3.6 La validation du mapping ISO 9001v2000 - CMMI

Le résultat de notre mapping a été validé sur deux niveaux :

- Validation par rapport aux résultats des travaux de Boris Mutafelija et Harvey Stromberg : En comparant le résultat de notre "mapping" et celui de Boris Mutafelija et Harvey Stromberg, la différence majeure concerne les pratiques génériques (GP) de CMMI. En effet, ces dernières fournissent une base générique pour l'institutionnalisation et l'implémentation des domaines de processus. De ce fait, Boris Mutafelija et Harvey Stromberg ont choisi de réaliser un "mapping" indépendamment du domaine de processus auquel sont appliquées ces pratiques génériques. Dans notre travail, nous avons choisi de traiter ces pratiques génériques au niveau de chaque domaine de processus car le contexte d'application, c'est à dire l'application à un domaine de processus, peut faire apparaître des relations de "mapping" différentes du contexte général énoncé par la pratique générique. À part cette différence, la plus grande partie des résultats coïncident. Le peu de différences qui existent correspondent à une compréhension différente des pratiques concernées ;
- Validation opérationnelle : le résultat de notre mapping a été validé par les managers de l'entreprise intéressée par les travaux. Ces personnes ont une grande expérience des bonnes pratiques informatiques et des standards qualité. Cette validation se repose sur la compréhension que les managers ont des pratiques et la façon de les implémenter.

9.2.4 L'interopérabilité des processus

La figure 9.10 représente une vue macroscopique de l'activité d'une entreprise. Cette vue montre bien au niveau métier les interactions qui apparaissent lors des projets entre l'organisation (équipes projet) et les clients et/ou partenaires. Ces interactions sont plus visibles dans les figures 9.11 et 9.12 qui montrent des cycles projet simplifiés avec et sans partenaire.

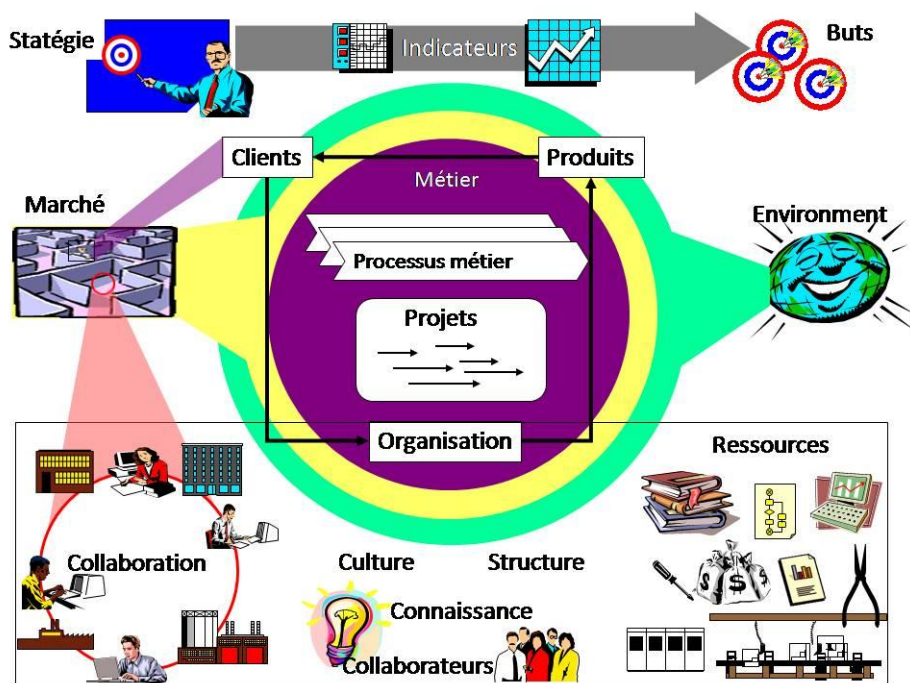


Figure 9.10 – Vue macroscopique de l'activité d'une entreprise

Ces figures montrent bien que toute entreprise qui travaille en interaction avec d'autres compagnies a tout intérêt à implémenter une démarche d'interopérabilité pouvant la rendre plus flexible et plus réactive vis-à-vis des demandes et des exigences de ses clients ou de ses partenaires.

L'activité de SYLIS fait apparaître deux types d'interaction :

- avec les clients ;
- avec d'éventuels partenaires ou sous-traitants.

La solution que nous proposons consiste à définir une approche d'interopérabilité spécifique à chacune de ces interactions.

9.2.4.1 La solution d'interopérabilité orientée clients

Ce mode d'interaction est basé sur un échange des documents relatifs au projet. La figure 9.13 montre la solution retenue pour assurer l'interopérabilité entre une entreprise et ses clients. Cette solution consiste à standardiser tous les documents échangés entre les deux parties.

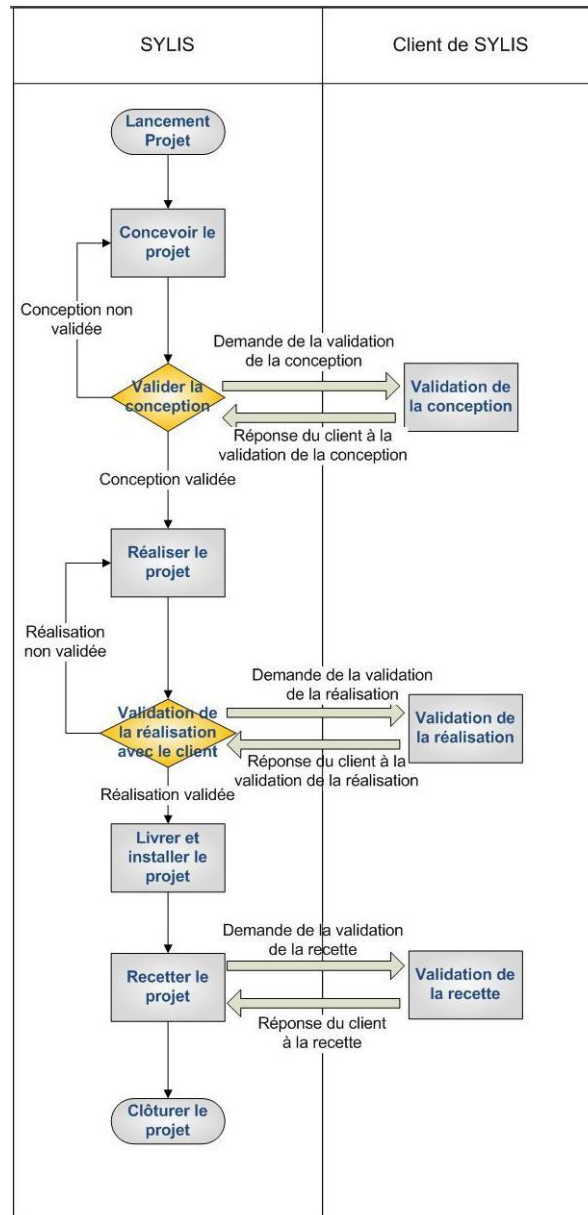


Figure 9.11 – Modèle simplifié d'un projet

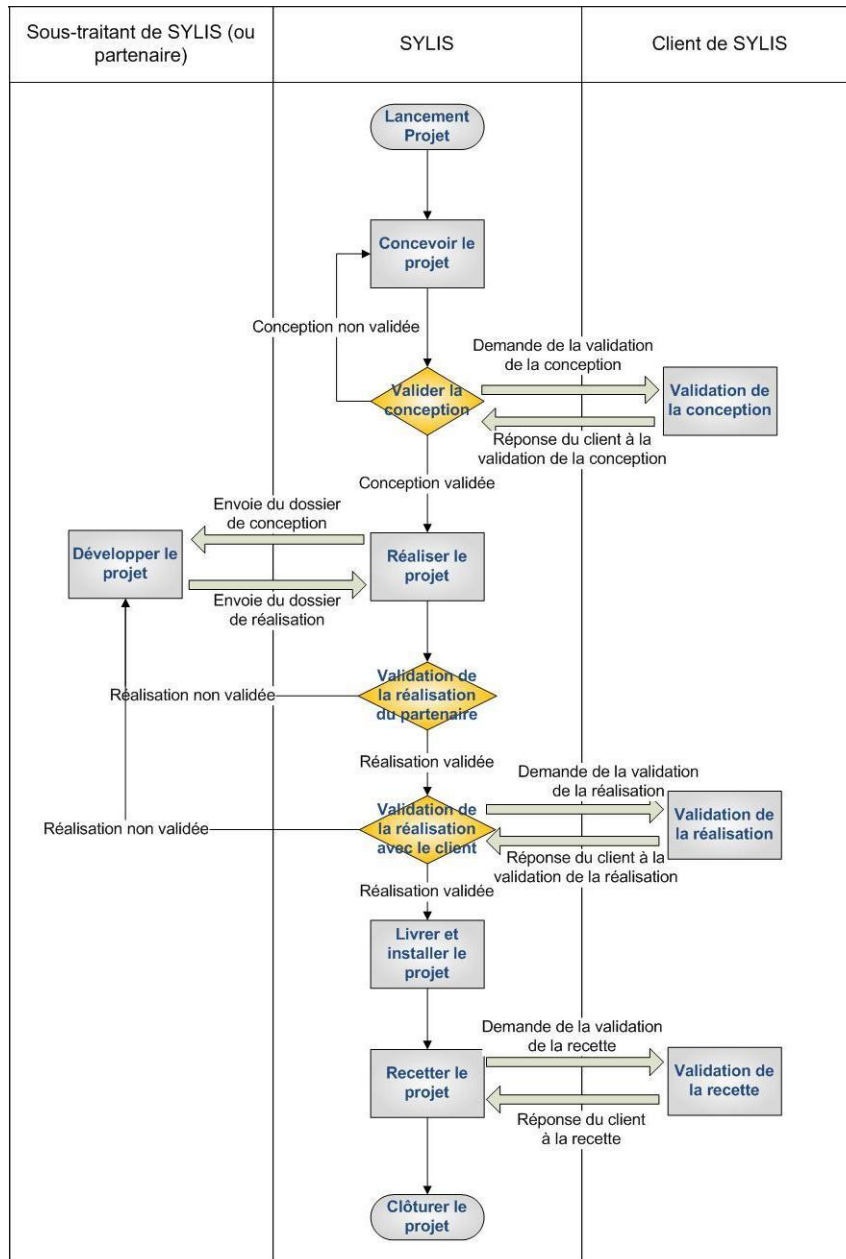


Figure 9.12 – Modèle simplifié d’un projet impliquant un partenaire

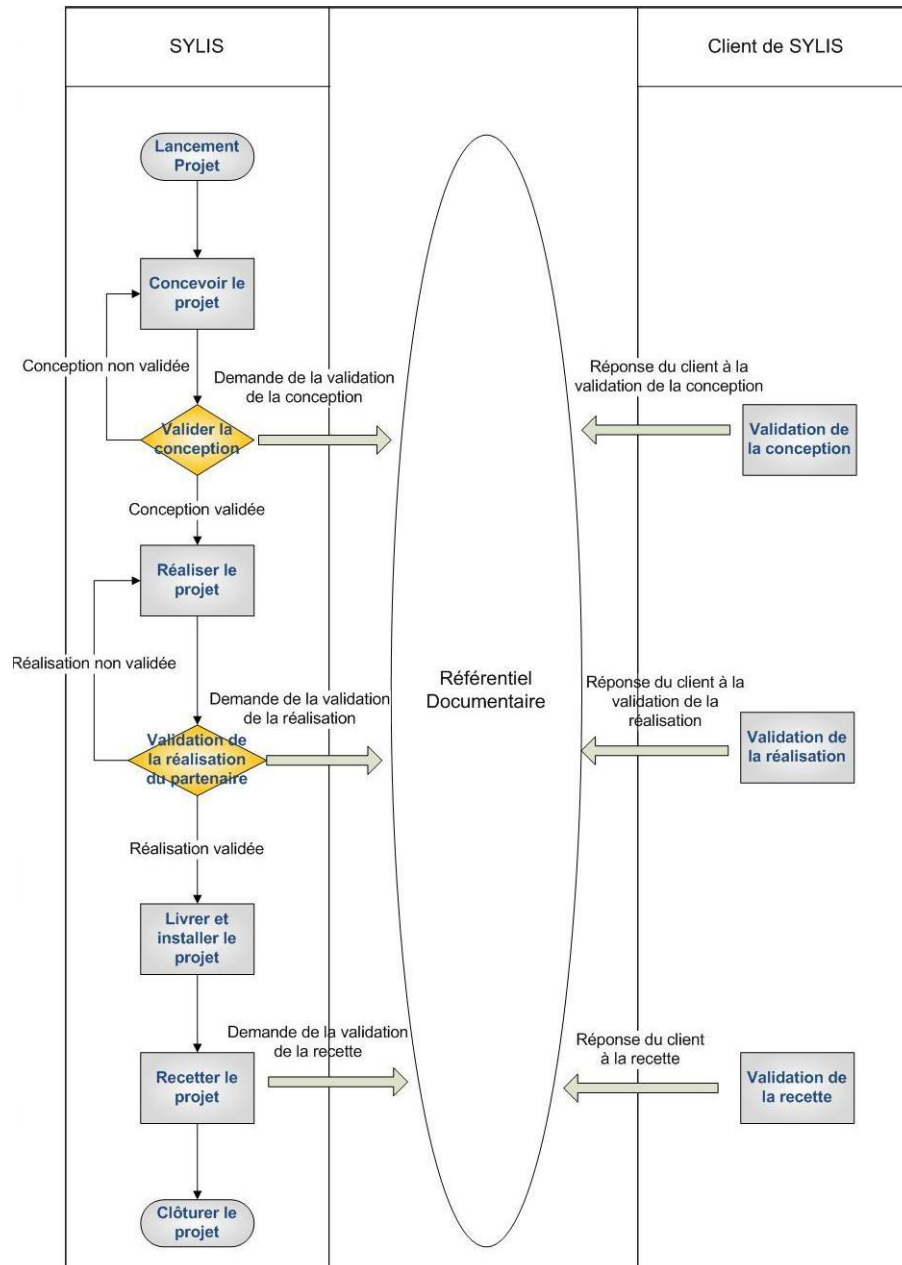


Figure 9.13 – Solution d’interopérabilité entre une entreprise et ses clients

Cet ensemble de documents standardisés destinés à être échangés avec le client constitue le "Référentiel Documentaire". Ce référentiel a pour but de créer une structure type de répertoire des projets. Cette structure va contenir tous les modèles de documents utilisables tout le long du projet. Ce répertoire type sera dupliqué pour les projets.

Cette solution prévoit, au cas où le client utilise des documents différents de ceux existants au sein de notre référentiel, une table de correspondance. Cette table permet de naviguer et de retrouver tous les documents du client relatifs à un client présent dans le référentiel documentaire. Cette table existe dans l'entête de chacun des documents du référentiel.

L'implémentation de ce référentiel documentaire va être présentée plus en détail dans la section 11.6.

9.2.4.2 La solution d'interopérabilité orientée partenaires

A partir de la cartographie des processus d'entreprise et du mapping entre CMMI et ISO 9001v2000, nous avons pu analyser le domaine d'intervention de CMMI au sein du référentiel entreprise et les améliorations que le modèle pouvait apporter. On voit ainsi l'apparition d'un nouveau processus au sein de la cartographie : "Gérer les ententes avec les fournisseurs Offshore".

Comme le montre la figure 9.14, le processus "interface" regroupe toutes les interactions des processus internes de SYLIS avec le partenaire. Ainsi, à chaque fois qu'un processus interne a besoin d'interagir avec un partenaire, c'est ce processus "interface" qui prend la main. Par exemple, dans la phase d'avant-vente, pour le choix du ou des partenaire(s) qui vont participer au projet, les processus "PROCES-REAL-001 Processus Vendre avec engagement de moyens" et "PROCES-REAL-002 Processus Vendre avec engagement de résultats" passent par ce processus pour la sélection des partenaires. Cette sélection se fait en consultant l'historique des partenaires ou même en les contactant pour avoir des informations clés sur leurs activités.

La figure 9.15 montre un exemple de processus en interaction avec les partenaires et qui passent toujours notre processus "interface".

La figure 9.16 montre ce processus "interface" ("Gérer les ententes avec les fournisseurs Offshore") qui permet de rendre les processus internes interopérables avec les processus d'un partenaire.

Ce processus regroupe neuf procédures qui représentent les principales interactions pouvant avoir lieu lors d'un projet impliquant un partenaire :

- "Sélectionner les partenaires répondant aux profils" : Elle a pour objet de décrire les

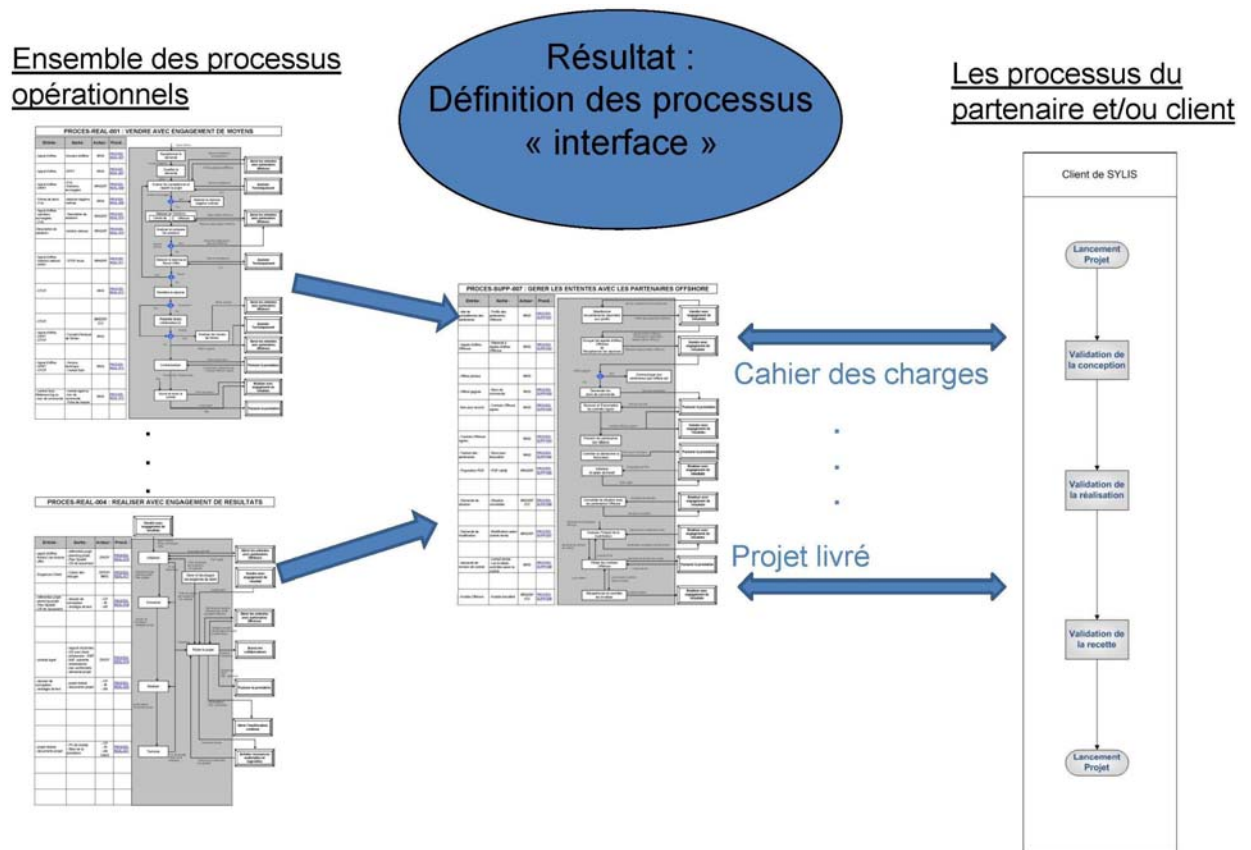


Figure 9.14 – Processus "interface"

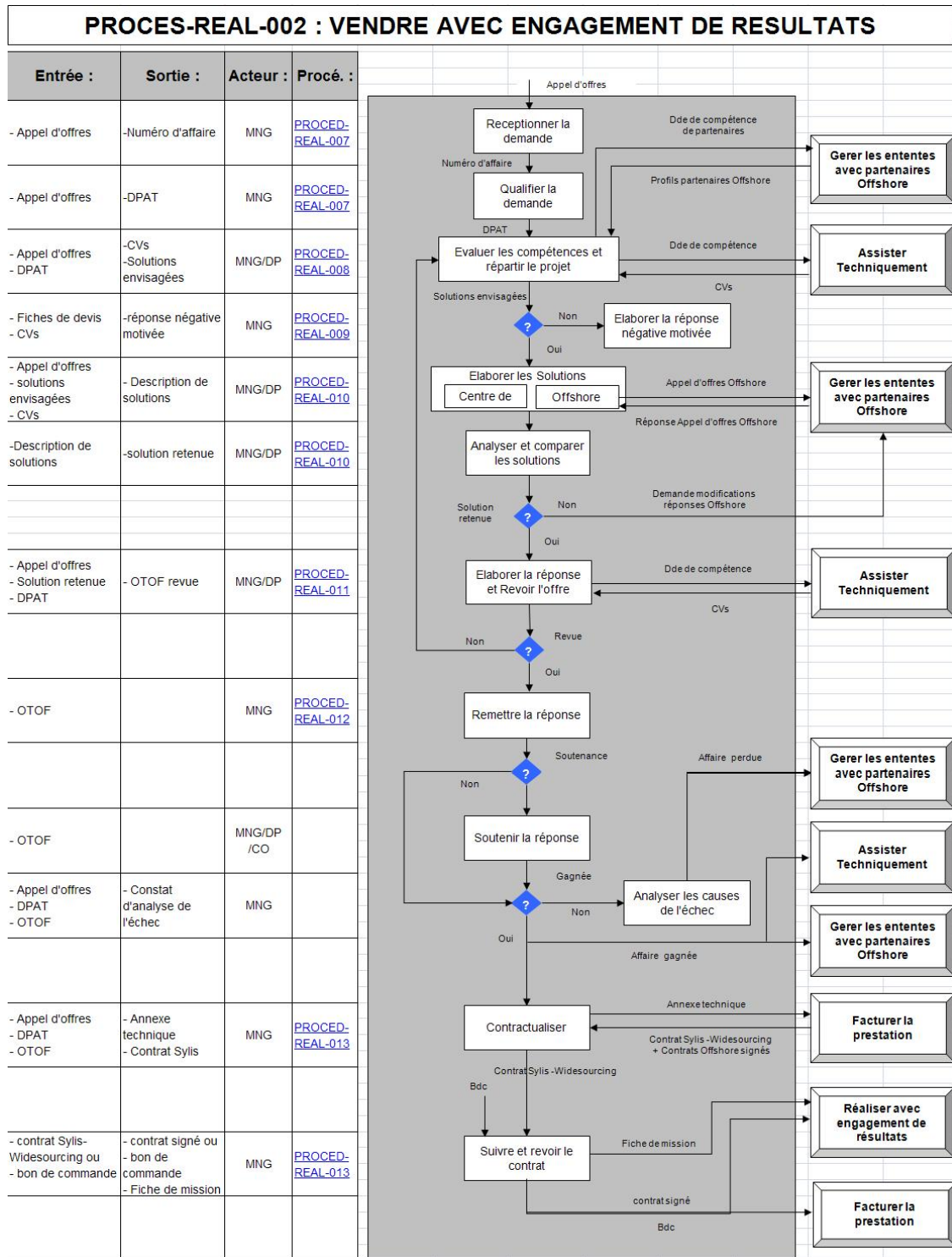


Figure 9.15 – Le processus "PROCES-REAL-002 : VENDRE AVEC ENGAGEMENT DE RESULTATS"

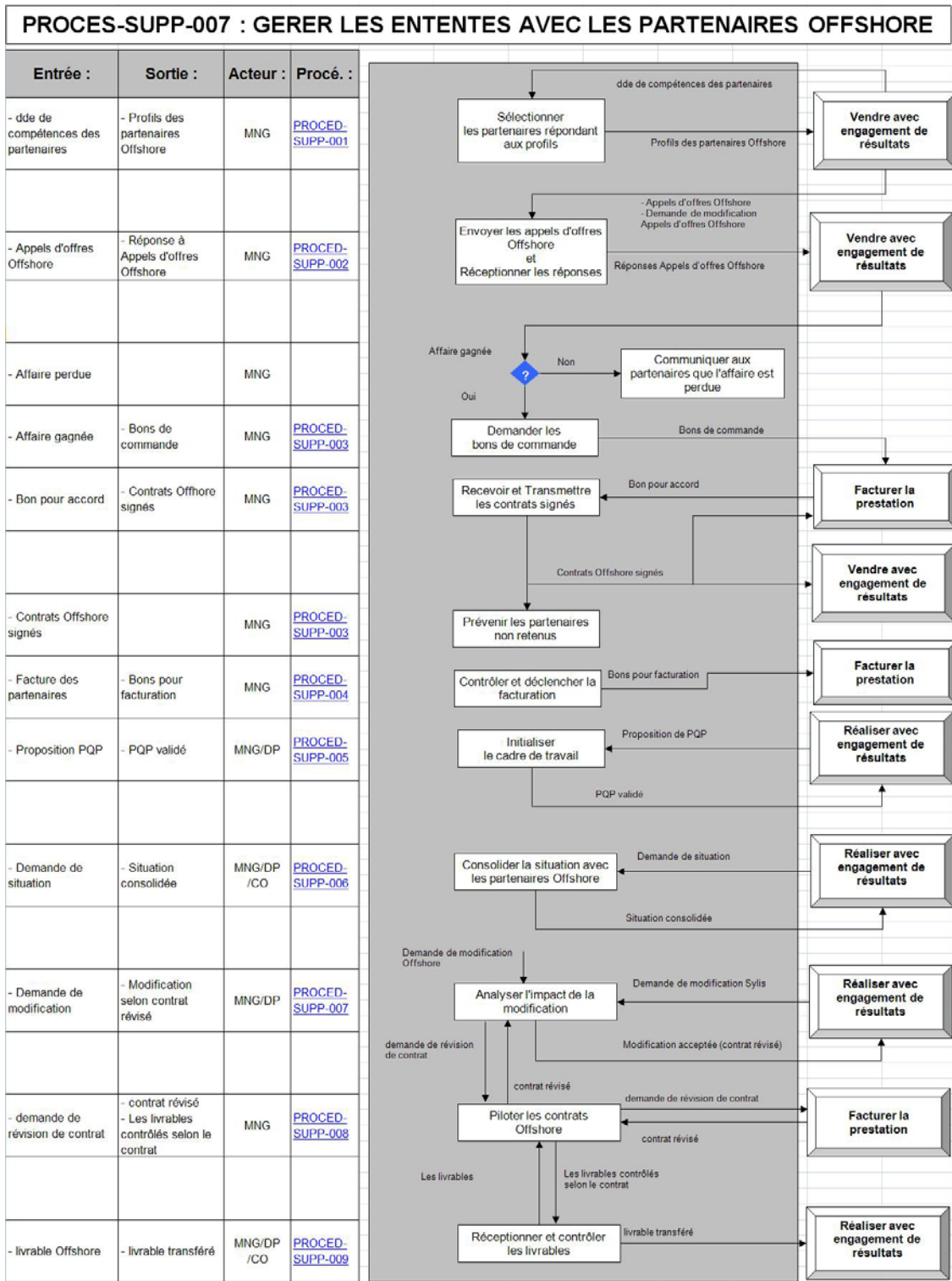


Figure 9.16 – Le processus "PROCES-SUPP-007 : GERER LES ENTENTES AVEC LES PARTENAIRES OFFSHORE"

étapes à suivre par les responsables de tâches afin de retrouver les profils des partenaires susceptibles de fournir les ressources requises pour le projet ;

- "Envoyer les appels d'offres Offshore et Réceptionner les réponses" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin de remettre un appel d'offres aux partenaires Offshore et recevoir leurs réponses ;
- "Demander les bons de commande", "Recevoir et Transmettre les contrats signés" et "Prévenir les partenaires non retenus" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin de rédiger les contrats Offshore, les faire signer par les partenaires et prévenir ceux qui n'ont pas été retenus ;
- "Contrôler et déclencher la facturation" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin de contrôler et déclencher la facturation Offshore ;
- "Initialiser le cadre de travail" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin d'étudier le PQP et le valider avec le partenaire ;
- "Consolider la situation avec les partenaires Offshore" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin de consolider la situation avec les partenaires Offshore et demander éventuellement des actions correctives ;
- "Analyser l'impact de la modification" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin d'analyser l'impact d'une modification de contrat et de diffuser le contrat révisé auprès du partenaire Offshore et du responsable de la réalisation du projet ;
- "Piloter les contrats Offshore" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin de piloter de gérer les contrats Offshore ;
- "Réceptionner et contrôler les livrables" : Elle a pour objet de décrire les étapes à suivre par les responsables de tâches afin de réceptionner les livrables Offshore, demander le contrôle par rapport au contrat et les transférer pour intégration ou livraison au client.

La figure 9.17 montre la procédure "Analyser l'impact de la modification". Toutes les procédures définies dans le processus "PROCES-SUPP-007 Processus Gérer les ententes avec les partenaires Offshore" ont été définies dans le détail afin de les rendre opérationnelles.

9.3 Conclusion

Dans ce chapitre, nous avons présenté notre modèle permettant d'intégrer les recommandations de deux standards qualité différents afin d'organiser les processus d'entreprise. Ce modèle a été appliqué pour créer un référentiel qualité unique multi-vues implémentant les recommandations d'ISO 9001v2000 et CMMI. Dans la deuxième partie de ce chapitre, nous avons présenté une application de notre référentiel afin d'assurer l'interopérabilité de nos pro-

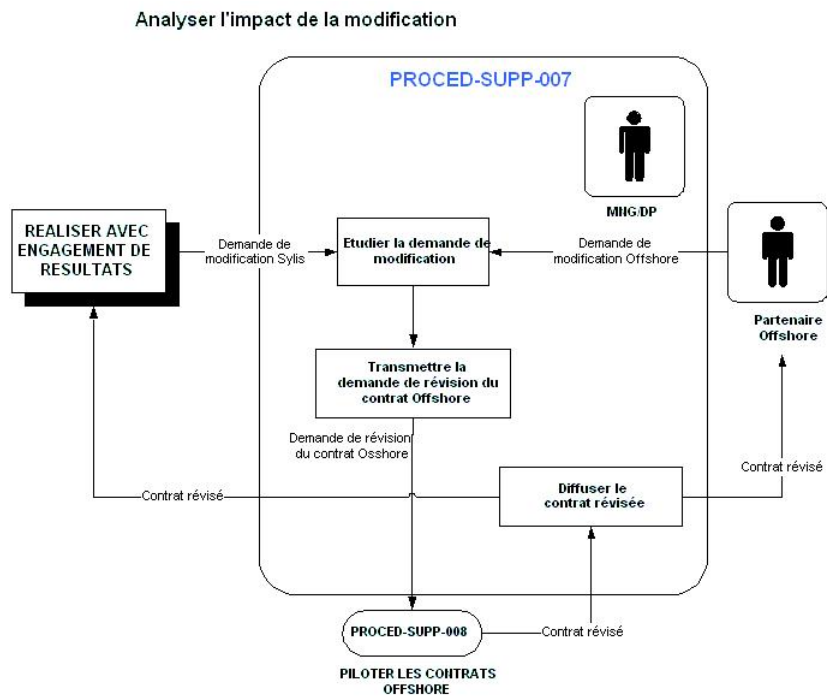


Figure 9.17 – La procédure "Analyser l'impact de la modification"

cessus avec ceux des clients et/ou partenaires. Dans le chapitre suivant, nous nous focalisons sur un processus critique. Nous définissons une méthodologie permettant l'estimation des délais et efforts des projets informatiques.

Focalisation sur un processus critique : estimation des délais et efforts des projets

10.1 Introduction

Dans nos travaux, nous avons intégré les recommandations d'ISO 9001v2000 et CMMI afin de créer un référentiel qualité multi-vues. L'étape suivante a été de mettre en place des méthodologies pour développer les processus vitaux de l'entreprise. Dans ce chapitre, nous nous focalisons sur un de ces processus vitaux pour l'organisation d'entreprise, à savoir l'estimation des délais et des charges des projets. Cette estimation est une tâche très complexe dans la gestion d'un projet logiciel. En effet, sans cette estimation, les gestionnaires et les autres responsables d'un projet logiciel ne peuvent assigner les budgets, affecter les ressources nécessaires et/ou établir les plans adéquats pour mener à bien le projet logiciel. C'est donc pour rationaliser le processus d'estimation du coût d'un projet logiciel que nous proposons le modèle suivant.

L'état de l'art des méthodes d'estimation des délais et des charges est présentée dans l'annexe D.

10.2 Modèle proposé

Le figure 10.1 montre le modèle d'estimation retenu et qui se base principalement sur COCOMO II, un modèle mis en place et validé par Barry W. Boehm en 2000 [15] au sein de du laboratoire *Center for Software Engineering (CSE)* de l'Université de Californie du Sud (*University of Southern California-USC*).

Dans la suite, nous allons détailler toute la démarche d'estimation en nous basant sur ce modèle et en partant des caractéristiques d'un projet, recueillies à partir du cahier des charges supposé complet [48] [47].

Comme le montre la figure 10.1, il y a un choix à faire dès le départ entre l'utilisation des lignes de code ou les points de fonction. Dans les sections qui suivent, nous allons présenter les deux mesures de taille qui sont mentionnées et un ensemble de recommandations pour bien choisir la mesure à utiliser.

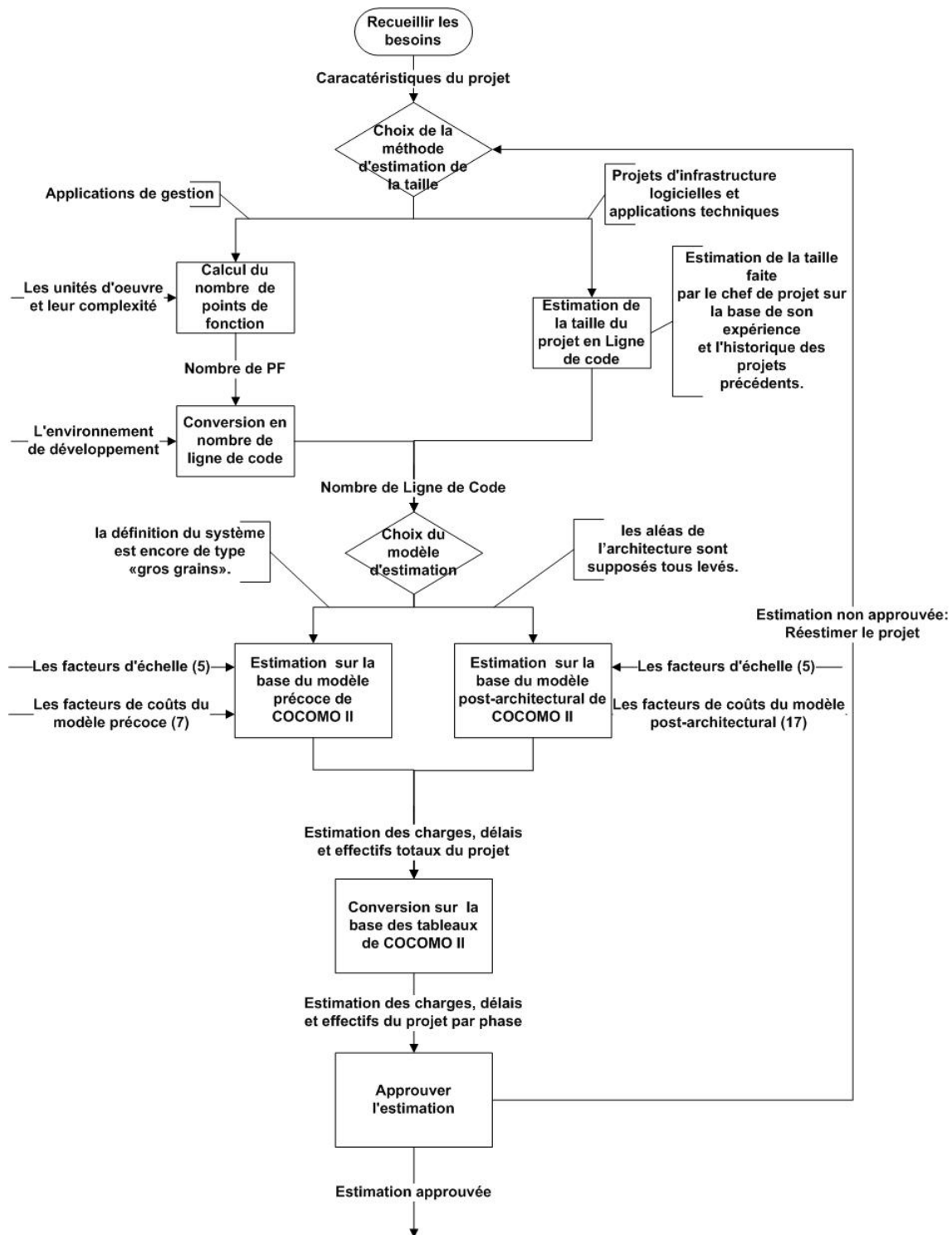


Figure 10.1 – Modèle proposé

10.3 Présentation de la ligne de code

Les lignes de code, souvent mentionnées comme Source Ligne Of Code (SLOC), sont des métriques utilisées pour mesurer la quantité de code dans un logiciel. Nous considérons la définition des sources de ligne logique retenue par The Software Engineering Institute (SEI) sur la base des principes suivants :

- les seules lignes de code qui sont livrées comme partie du produit sont prises en compte, les tests et autres supports logiciels sont exclus ;
- le code généré par les générateurs d'application est exclu ;
- une ligne de code est une instruction logique (Logical Statement au sens de SEI) ;
- les déclarations sont comptées ;
- les commentaires ne sont pas comptés ;
- on ne tient pas compte du logiciel de support qui n'est pas livré même si son développement a pu nécessiter beaucoup de ressources ;
- on compte tout le reste : déclaration des données, instructions de format . . .
- tout le développement après la définition du Cahier de Charge, test compris, est couvert
- toutes les activités de gestion de projet, conception, documentation sont couvertes mais non la formation des utilisateurs. On ne prend pas en compte les opérateurs de l'ordinateur.

10.4 Présentation des Points de Fonction (FP)

La méthode des points de Fonction est une métrique qui a été initiée par Albrecht et Gaffney en 1983 au sein d'IBM. Cette métrique a été reprise par le laboratoire *International Function Point Users Group (IFPUG)* qui publie régulièrement des mises à jour du manuel de bonnes pratiques. L'analyse par Point de Fonction est une technique objective et structurée pour mesurer la taille de logiciel en évaluant quantitativement les fonctionnalités fournies à l'utilisateur, en se basant sur les spécifications et la conception logique. Cette technique divise le système en des composants plus petits et donc plus facile à comprendre et analyser. On distingue 5 composants principaux d'Analyse de Point de Fonction : Entrées Externes (EI), Sorties Externes (EO), Requêtes Externes (EQ), Fichiers Logiques Internes (ILF) et Fichiers d'Interface Externes (EIF). Les trois premiers sont traités comme des Types de Fonction Transactionnels et les deux derniers sont appelés des Types de Fonction de Données. Les composants de la techniques des Points de Fonction sont présentés en détail dans l'annexe E.

10.4.1 Des Points de Fonction vers les Lignes de Code

À la fin de l'étape précédente, on a comptabilisé le poids de l'application en Points de Fonction. La méthodologie retenue permet ensuite de convertir les Points de Fonction vers le nombre de ligne de code correspondant en fonction du langage de développement utilisé en se basant sur des études statistiques faites sur la base de projets logiciels finis. C'est le *Software Productivity Research* qui gère les études sur les métriques des logiciels et étudie en partie les conversions possibles entre toutes ces métriques. Cet organisme publie régulièrement des versions mises à jour des tables de conversions. Nous avons retenu la dernière version publiée et validée.

Le tableau 10.1 montre la table de correspondance entre les Points de Fonction et le nombre de lignes en fonction du langage.

Langage	SLOC/FP	Langage	SLOC/FP
Access	38	Langage de haut niveau	64
Ada 83	71	HTML 3.0	15
Ada 95	49	JAVA	53
AI Shell	49	Jovial	107
APL	32	Lisp	64
Assembleur-Basic	320	Code Machine	640
Assembleur-Macro	213	Modula 2	80
Basic-ANSI	64	Pascal	91
Basic-Compilé	49	PERL	21
Basic-Visual	32	PowerBuilder	16
C	128	Prolog	64
C++	53	Langage de requête-Défaut	13
Cobol(ANSI 85)	91	Générateur de Report	80
Base de Données	40	L2G	107
L5G	5	Simulation-Défaut	46
L1G	320	L3G	80
Forth	49	Unix-Script Shell	21
Fortran 77	107	Visual Basic 5.0	29
Fortran 95	71	Visual C++	34
L4G	20		

Tableau 10.1 – Tableau de conversion des FP en SLOC.

10.5 Recommandations

Étant donné que nous avons présenté deux méthodes différentes pour mesurer la taille du projet (section 10.4 et 10.3), nous exposons dans cette partie un ensemble de recommandations permettant de choisir la mesure la plus appropriée en fonction de la catégorie à laquelle appartient l'application :

- le modèle d'estimation de la taille par Points de Fonction est bien adapté aux applications de gestion, car la modélisation en modèle de données et modèle des traitements se prête bien à l'analyse par Points de Fonction, et ceci dès la phase de conception générale, alors que peu d'information sur la taille des programmes à écrire est disponible. Ces applications de gestion intègrent des modules applicatifs qui sont soit transactionnels, soit de type infocentre (utilisation fréquente de générateurs d'états standardisés), soit batch ;
- les projets d'infrastructures logicielles et applications techniques correspondent aux développements faits chez les constructeurs de matériel (développements de tout ou partie d'un système d'exploitation) et chez les éditeurs de logiciels génériques (systèmes de gestion de base de données, compilateurs, moniteurs et autres middlewares, bibliothèques de composants logiciels. . .), les applications ou progiciels "techniques" non liés à la gestion (CAO, DAO. . .) sont en général aussi rattachés à ce type de logiciel. Dans ce cas, l'estimation en Points de Fonction est le plus souvent inadaptée et on a recourt alors à une estimation basée sur la quantité et la complexité des lignes de code à produire ;
- les projets d'intégration de systèmes sont les plus grands projets complexes que l'on peut trouver. Dans la démarche d'intégration de systèmes, les projets cumulent la contrainte d'adéquation à des besoins spécifiques "métiers" des applications de gestion, avec la complexité d'architecture des infrastructures logicielles. En effet, la spécification des besoins peut demander des fonctions non fournies par l'infrastructure de la plate-forme d'exécution, ou nécessiter des développements de pilotes périphériques, d'adaptations d'interfaces hétérogènes, de protocoles d'interopérabilité non standard. . . Dans ce cas, l'opération d'estimation utilise les Points de Fonction pour les modules le justifiant et les lignes de code pour les autres modules. Ceci implique une analyse détaillée de l'architecture fonctionnelle et du découpage en modules pour capturer toutes les fonctions à développer et leur complexité.

10.6 Choix du modèle d'estimation

Après avoir estimé la taille du projet, l'étape suivante consiste à choisir le modèle d'estimation à utiliser. La méthode retenue offre deux modèles d'estimation qui correspondent à deux cadres d'utilisation bien précis :

le modèle précoce (The Early Design Model) : C'est un modèle de haut niveau qui est utilisé pour explorer les alternatives architecturales et/ou les stratégies de développement. La phase de définition est celle où toutes les incertitudes concernant l'architecture à mettre en place doivent être levées et où la définition du système est encore de type "gros grains". Ce niveau de détail correspond au niveau de détail des informations disponibles et au niveau d'exactitude de l'évaluation désiré. Ce modèle se base sur l'évaluation de 7 facteurs de coûts. Ces facteurs sont présentés en détail dans l'annexe E ;

le modèle post-architecture (The post-architecture Model) : L'utilisation de ce modèle implique que le cycle de développement du système a été défini et validé. Ce modèle est appelé post-architecture pour bien indiquer que les aléas de l'architecture sont supposés tous résolus, bien que cela ne présuppose pas que tout le travail architectural soit effectivement réalisé. Pendant cette phase, on a une meilleure vision du projet ce qui permet une estimation plus précise qui se base sur l'évaluation de 17 facteurs de coûts. Ces facteurs sont présentés en détail dans l'annexe E.

10.7 Calcul de l'effort et du temps de développement

Les formules retenues sont celles définies dans le modèle COCOMO II. Ainsi pour calculer l'effort :

$$\begin{aligned} \text{Effort} &= \prod_{i=1}^n (\text{EM}_i) \times A \times \text{Taille}^B \\ B &= 0.91 + 0.01 \times \sum_{i=1}^5 (\text{SF}_i) \\ A &= 2.94 \end{aligned} \tag{10.1}$$

où Effort est exprimé en homme-mois (h.m), EM_i sont les facteurs de coûts ($n = 7$ pour le modèle précoce et $n = 17$ pour le modèle post-architecture) et SF_i sont les facteurs d'échelle.

$$\begin{aligned}
 \text{TDEV} &= \left[3.67 \times \overline{\text{PM}}^C \right] \times \frac{\text{SCED}\%}{100} \\
 C &= 0.28 + 0.2 \times (B - 0.91) \\
 B &= 0.91 + 0.01 \times \sum_{i=1}^5 (\text{SF}_i)
 \end{aligned}
 \tag{10.2}$$

où TDEV est la durée du projet exprimée en mois, $\overline{\text{PM}}$ est l'effort estimé sans tenir compte du facteur d'échelle SCED et SCED% est la valeur du facteur d'échelle SCED.

10.8 La répartition par phase

À la fin de l'étape précédente, nous avons une estimation globale des charges et délais du projet. Le modèle permet ensuite de répartir les charges et les délais totaux calculés par phase du projet, en adoptant la répartition décrite par les tableaux 10.2 et 10.3. Pour les valeurs des tailles qui n'existent pas dans les tableaux, on approche les valeurs des pourcentages par interpolation.

	Petits 2 KSLOC	Intermédiaires 8 KSLOC	Moyens 32 KSLOC	Grands 128 KSLOC	Très grands 512 KSLOC
Spécifications fonctionnelles	7%	7%	7%	7%	7%
Conception Générale	17%	17%	17%	17%	17%
Programmation (Codage et tests unitaires)	64%	61%	58%	55%	52%
Intégration et tests	19%	22%	25%	28%	31%

Figure 10.2 – La répartition des charges totales par phase

10.9 Le calibrage du modèle

Le calibrage du modèle constitue la deuxième étape proposée et consiste en la mise en place du modèle qui peut être considéré générique dans une organisation particulière. Le calibrage et la mise en place du modèle dans une organisation particulière comprend :

- le calibrage du modèle sur la base des projets existants ;

	Petits 2 KSLOC	Intermédiaires 8 KSLOC	Moyens 32 KSLOC	Grands 128 KSLOC	Très grands 512 KSLOC
Spécifications fonctionnelles	16%	18%	20%	22%	24%
Conception Générale	24%	25%	26%	27%	28%
Programmation (Codage et tests unitaires)	56%	52%	48%	44%	40%
Intégration et tests	20%	23%	26%	29%	32%

Figure 10.3 – La répartition des délais totaux par phase

- la consolidation ou élimination des paramètres redondants ;
- l'ajout de nouveaux facteurs de coûts qui ne sont pas explicités dans le modèle.

Dans notre travail, nous avons considéré seulement la première approche. Ce choix est dicté par le fait que les deux autres approches nécessitent un nombre important de projets pour l'étalonnage à une entreprise donnée. Par exemple, pour la mise en place des paramètres du modèle COCOMO II, ces deux approches ont nécessité 161 projets.

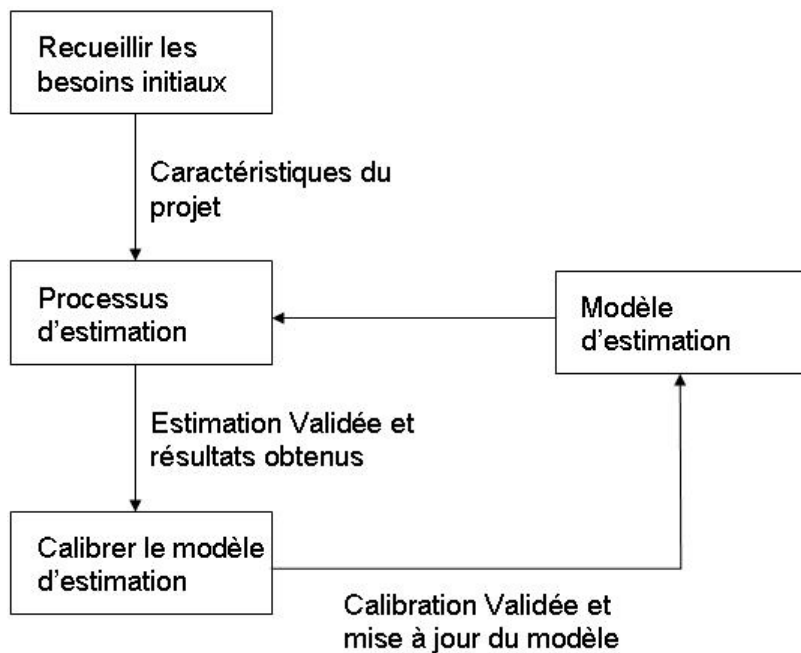


Figure 10.4 – Le processus d'estimation incluant le calibrage.

Le figure 10.4 montre l'approche retenue qui consiste à recueillir les résultats des projets et à les injecter dans le modèle afin de calibrer ses facteurs et pour que le nouveau modèle calibré soit disponible pour une nouvelle estimation.

Nous rappelons que la formule pour calculer l'Effort est la suivante :

$$\begin{aligned} \text{Effort} &= \prod_{i=1}^{17} (\text{EM}_i) \times A \times \text{Taille}^B \\ B &= B_1 + 0.01 \times \sum_{i=1}^5 (\text{SF}_i) \\ B_1 &= 0.91 \\ A &= 2.94 \end{aligned} \tag{10.3}$$

Le modèle permet de calibrer la constante A et le facteur B_1 mais il est recommandé d'avoir au moins les résultats de 5 projets pour calibrer le premier facteur et 10 pour B_1 . Nous utilisons une approche qui se base sur la régression linéaire (expliquée dans l'annexe D.5.1). La transformation logarithmique donne :

$$\begin{aligned} \ln(\text{Effort}) &= \beta_0 + \beta_1 \times \ln(\text{Taille}) + \beta_2 \times \text{SF}_1 \times \ln(\text{Taille}) \\ &+ \dots + \beta_6 \times \text{SF}_5 \times \ln(\text{Taille}) + \beta_7 \times \ln(\text{EM}_1) \\ &+ \dots + \beta_{22} \times \ln(\text{EM}_{16}) + \beta_{23} \times \ln(\text{EM}_{17}) \end{aligned} \tag{10.4}$$

où $A = e^{\beta_0}$ et $B_1 = \beta_1$

Il reste à injecter les valeurs de l'Effort des projets finis dans l'équation et dérouler la régression linéaire.

10.10 Conclusion

Dans ce chapitre, nous avons développé un modèle d'estimation des charges et des délais des développements logiciels. Ce modèle est basé :

- les Points de Fonction, définie par *Dr. Allan Albrecht* au sein d'IBM et repris par le laboratoire *International Function Point Users Group (IFPUG)*, pour le calcul de la taille du projet ;
- la méthode *COCOMO II*, définie par *Dr. Barry Boehm* au sein du laboratoire *Center for Software Engineering (CSE)* de l'Université de Californie du Sud (*University of Southern California-USC*), pour l'estimation des coûts et des charges.

La mise en place de ce modèle au sein d'une organisation particulière prend en compte l'aspect humain et culturel de l'entreprise pour palier le problème d'acceptabilité, qui peut se manifester lors de la conduite du changement. Ainsi, nous avons tenu à adopter une stratégie de persuasion à travers des présentations de la méthode et de ses avantages.

Le processus d'estimation retenu comporte 3 étapes :

- 1) l'estimation de la taille du projets en Points de Fonction ou en nombre de lignes de code ;
- 2) le choix du modèle d'estimation :
 - **le modèle précoce (The Early Design Model)** qui est un modèle de haut niveau utilisé tôt dans le cycle de développement ;
 - **le modèle post-architecture (The post-architecture Model)** qui est un modèle permettant une estimation plus précise des attributs du projet.
- 3) le calibrage des facteurs du modèle sur la base des coûts et des charges réels des projets déjà achevés.

CINQUIÈME PARTIE

CAS D'ÉTUDE

Application à Sylis

11.1 Introduction

Le chapitre précédent nous a permis de présenter la démarche générale permettant d'intégrer deux standards qualité afin de créer un référentiel qualité multi-vues. Cette démarche est ensuite appliquée à l'intégration d'ISO 9001v2000 et de CMMI. Le présent chapitre nous permet de présenter l'implémentation de notre démarche, appliqué à ISO 9001v2000 et CMMI, au sein de Sylis. La mise en place du référentiel qualité est accompagnée de la mise en place d'un référentiel documentaire, d'une méthodologie d'audit projet et d'un référentiel de compétences garants de la bonne implémentation du référentiel qualité au sein d'une entreprise.

11.2 Présentation de Sylis

Sylis est une une Société de Services en Ingénierie Informatique (SSII) créée en 1984, cotée au second marché de la bourse de Paris depuis 1997. La vocation du Groupe SYLIS est d'offrir à ses clients européens une gamme complète de services informatiques performants, innovants et à forte valeur ajoutée. La stratégie de SYLIS vise non seulement la couverture nationale, mais aussi celle de l'espace européen, par extension des zones de chalandise de ses établissements frontaliers.

Le Groupe Sylis se positionne comme partenaire, conseiller et intégrateur, dans le cadre de l'urbanisme, de l'architecture, de la conception, de l'implémentation, du déploiement et du maintien en condition opérationnelle des systèmes d'information stratégiques.

Chiffres d'affaires	138 Millions D'euros
Effectifs du groupe	1727 Collaborateurs

Tableau 11.1 – Les chiffres clés de Sylis en 2007

Le Groupe Sylis est structuré en fonction des métiers et des secteurs d'activité de ses clients (comme le montre la figure 11.1). Ainsi, les ingénieurs du Groupe, organisés en secteurs verticaux, sont à même de comprendre les problématiques "business" particulières à chacun de leurs clients. En terme de résultat opérationnel, Sylis est organisé en centres de profit géographiquement autonomes. Les systèmes d'information analytiques sont calqués sur cette organisation. Aussi, la ventilation des marges est organisée et agrégée par pays dans le cadre de l'information sectorielle.

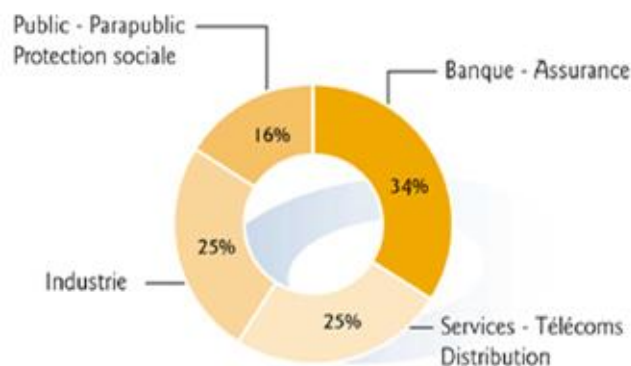


Figure 11.1 – Répartition du chiffre d'affaire par métier client

11.2.1 Les métiers de Sylis

Comme le montre la figure 11.2, l'offre de Sylis prend en compte la globalité des besoins des entreprises en matière de système d'information stratégique et couvre tous les aspects de l'ingénierie informatique.

Cette offre s'articule autour de quatre lignes produits :

- conseil, expertise et solutions ;
- gestion des systèmes d'information ;
- exploitation des systèmes d'information ;
- gestion des infrastructures.

Pour chacune de ces lignes, Sylis a capitalisé un savoir-faire lui permettant de proposer à sa clientèle une offre avec engagement de résultats et assurance Qualité.

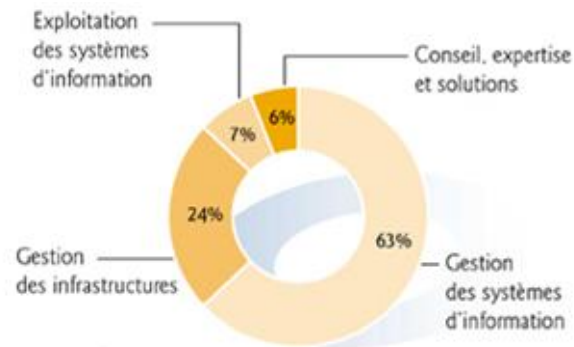


Figure 11.2 – Répartition des métiers de Syllis

Les solutions de Syllis touchent un large marché. Après avoir principalement œuvré pour les grands comptes de l'économie européenne, Syllis s'adresse aujourd'hui également au "midsize market" : le Groupe a notamment conclu un partenariat avec Microsoft qui lui confère un savoir-faire reconnu quant aux produits applicatifs des PME.

Syllis a développé des solutions packagées pour un service à plus forte valeur ajoutée. Des offres ingénierie (Tierce maintenance applicative, Migration Bascule, Tierce recette applicative...), conseil (Accompagnement au changement, Organisation, Formation...) et solutions (Axapta, Portail, Décisionnel...) sont ainsi proposées à la clientèle.

Conseil, expertise et solutions

Les consultants interviennent sur des aspects tant organisationnels, fonctionnels que technologiques afin d'apporter des solutions pragmatiques, technologiquement viables avec retour sur investissement tangible pour le client.

Les axes stratégiques majeurs concernent :

- le décisionnel ;
- la relation client ;
- les ERPs ;
- le Knowledge Management ;
- le training et l'accompagnement à la conduite du changement ;
- le SCM (Supply Chain Management) ;
- la technologie RFID (Radio Frequency IDentification).

Gestion des systèmes d'information

Sylis apporte à ses clients un soutien dans la gestion opérationnelle des systèmes mis en place, ces derniers étant directement liés aux axes stratégiques précédemment cités (cf Conseil et expertise).

- la maintenance d'applicatifs opérationnels ;
- l'accompagnement aux nouvelles technologies ;
- le développement de projets nouveaux sur mesure ;
- la tierce maintenance applicative ;
- la tierce recette applicative ;
- l'intégration de systèmes.

Ces prestations sont réalisées sur les sites clients ou sur la plate-forme délocalisée du Groupe.

Exploitation des systèmes d'information

Sylis prend en charge tout ou partie de l'informatique de production de ses clients et leur permet de bénéficier de l'ensemble des savoir-faire du Groupe :

- pilotage de la production ;
- suivi et administration ;
- support technique et help-desk ;
- infogérance sur site ou sur nos plates-formes délocalisées.

Gestion des infrastructures

Intervenant sur tous types d'environnements (systèmes centraux, middleware et poste de travail...), Sylis se positionne comme partenaire de proximité dans le cadre de la maîtrise et l'optimisation des systèmes et réseaux de l'entreprise.

Ainsi, Sylis assiste en permanence les équipes informatiques internes aux entreprises dans les domaines suivants :

- évolutions d'architecture et tests techniques ;
- déploiement et maintien en condition opérationnelle :
 - du middleware ;
 - des serveurs ;
 - des postes de travail ;
 - des logiciels.
- sécurité des réseaux et systèmes au travers des firewall, anti-virus et anti-spam...

11.2.2 Les axes d'intervention de Sylis

Les quatre lignes métiers décrites précédemment se déclinent de manière transverse suivants les axes : conseil et expertise, Ingénierie en Hautes Technologies, Intégration de Systèmes et Infogérance applicative. La définition de ces axes est la suivante :

Conseil, expertise et solutions

décrit précédemment dans la partie 11.2.1.

Ingénierie en Hautes Technologies

Sylis prend en charge tout ou partie du projet pour le compte de ses clients avec des ingénieurs spécialistes en technologies de l'information ayant des connaissances approfondies des métiers et du domaine fonctionnel de leur intervention (services, banque, assurance, industrie, administrations...). Leurs prestations sont réalisées selon des méthodes rigoureuses permettant un contrôle permanent de la qualité et de la valeur ajoutée du service rendu, Sylis garantissant ainsi sa capacité à comprendre et analyser les besoins du client.

Intégration de Systèmes

Par le biais de l'activité intégration de systèmes, le Groupe Sylis est l'ensemblier de solutions globales d'informatisation et en assure la maîtrise d'œuvre complète dans le cadre de développements spécifiques ; la spécificité de l'offre Sylis sur ce type de contrats est de prendre des engagements de résultats, via les méthodes du marché ISO, CMMI, ITIL. À ce titre, les licences et matériels vendus dans le cadre de ces solutions globales sont inclus dans cette activité.

Infogérance applicative

Les solutions sur mesure fournies par le Groupe Sylis sont basées sur une forte expertise dans le domaine de l'assistance et de la maintenance des systèmes d'information. Intervenant sur tous types d'environnements (systèmes centraux, client-serveur, poste de travail), Sylis prend en charge tout ou partie de la gestion d'un système (Tierce- Maintenance Applicative, Help Desk, Facility Management). Les équipes Sylis interviennent à distance ou sur le site

même de l'entreprise.

11.3 Description de la démarche globale

Avec la direction de Sylis, nous avons déterminé le cadre du projet :

Objectifs stratégiques de la direction

Ce projet répond aux objectifs stratégiques de la filiale SDIE du groupe Sylis de développer une industrie logicielle, délivrant une approche et une solution :

- industrielle ;
- qualitative ;
- pour les projets externalisés des clients ;
- avec une souplesse des moyens et des coûts associés.

Afin de répondre dans les meilleures conditions à ces objectifs, SDIE développe une démarche qualité mixte suivant :

- la norme ISO 9001v2000 déjà en place chez Sylis Nord ;
- le modèle CMMI, "guide de bonnes pratiques", développé par le Software Engineering Institute (SEI).

Afin de fournir la preuve de son engagement au développement et à la mise en œuvre du système de management de la qualité ainsi qu'à l'amélioration continue de son efficacité, la direction s'est engagée à :

- communiquer au sein de l'organisme l'importance de satisfaire les exigences des clients ainsi que les exigences réglementaires et légales ;
- établir la politique qualité ;
- assurer que des objectifs qualité sont établis ;
- mener des revues de direction ;
- assurer la disponibilité des ressources.

3 axes de la politique qualité :

- communiquer sur le savoir faire de SDIE avec les agences commerciales de Sylis et ses clients ;
- mieux produire en industrialisant ;
- appliquer une gestion adaptée aux collaborateurs SDIE.

Objectifs spécifiques du projet :

- 1) mettre en place un système d'information liant :
 - données de travail des collaborateurs (modèles types des documents à utiliser, les outils. . .) ;
 - référentiel qualité de l'entreprise ;
 - Données d'évaluation ISO 9001v2000 et CMMI.
- 2) définir le référentiel qualité de l'entreprise en accord avec les objectifs stratégiques et les axes de la politique qualité définis par la direction.

Cahier des charges du référentiel entreprise :

Il doit :

- être utilisable et donc proche des collaborateurs, à leur écoute ;
- être diffusé, connu et utilisé par les collaborateurs ;
- intégrer le modèle ISO 9001v2000 et le concept d'amélioration continue ;
- intégrer le modèle CMMI et le concept d'amélioration continue ;
- être facilement évaluable au sens ISO 9001v2000 ;
- être facilement évaluable au sens CMMI.

Pour répondre à ces objectifs et au cahier des charges, deux solutions sont envisageables :
On peut définir plusieurs référentiels entreprise pour les différents utilisateurs, collaborateurs, évaluateur ISO et évaluateur CMMI ou définir un seul référentiel répondant aux deux modèles.

La seconde solution a été retenue. Elle présente deux principaux avantages :

- une vision unique pour tous les acteurs collaborateurs et évaluateurs (pas de risques de confusion entre les frameworks de CMMI et ISO 9001v2000) ;
- un seul référentiel à maintenir.

11.4 Référentiel qualité

Afin de répondre aux règles établies lors de la réalisation du référentiel qualité deux notions de clients ont été distinguées :

- on appellera **Client Commercial**, les agences commerciales Sylis avec lesquelles nous adoptons une relation d'ordre fournisseur-client (contrat. . .) ;

- on appellera **Client Produit**, les clients des agences commerciales de Syllis qui leur ont commandé une prestation SDIE.

SDIE WideSourcing a pour vocation principale de **vendre et réaliser avec engagement de résultats (Forfait)** cependant elle a la possibilité de manière très ponctuelle de **vendre et réaliser avec engagement de moyens (Régie)** (ex : Location de ressources WideSourcing aux clients commerciaux ou produit) d'où sa petite taille sur la cartographie.

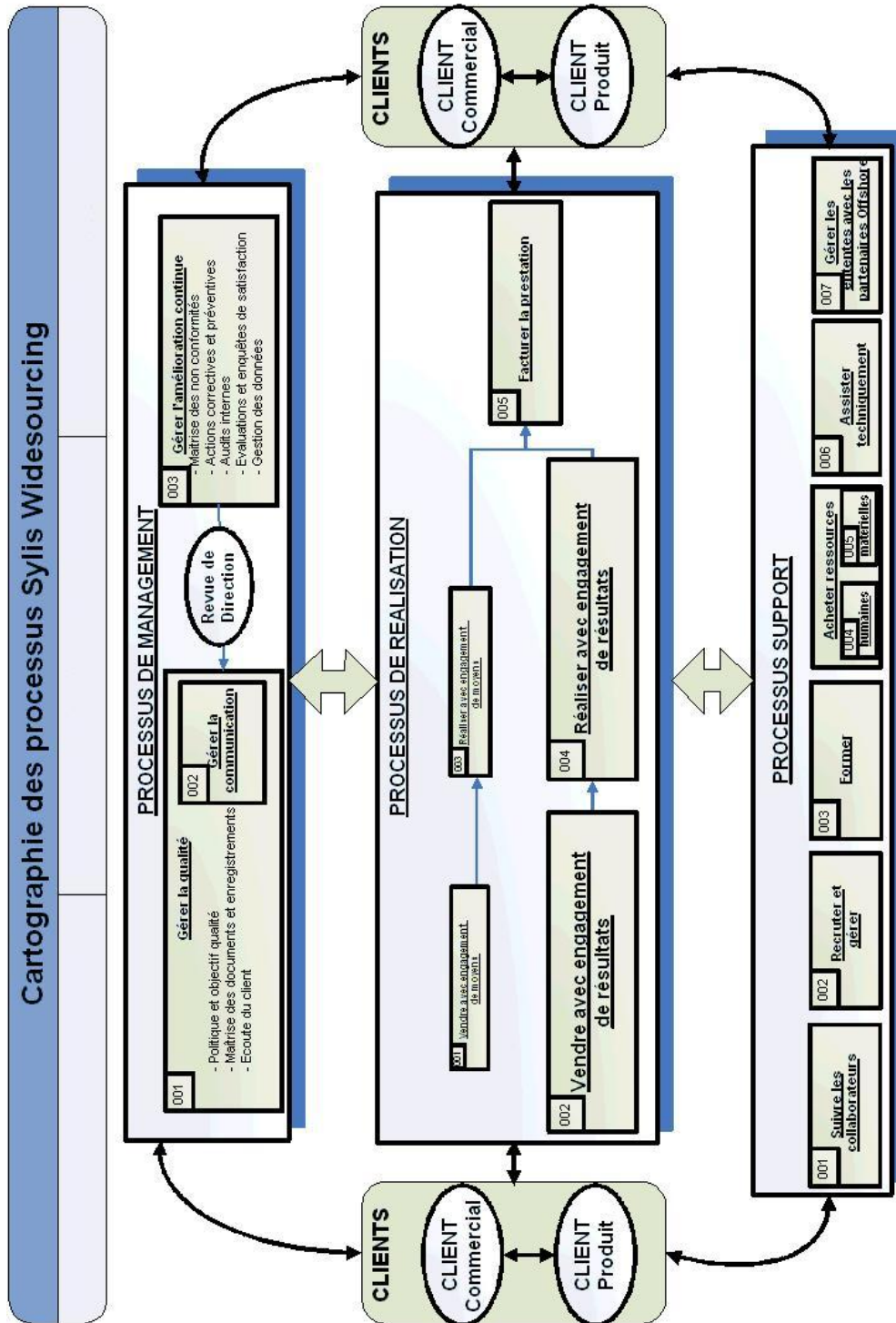


Figure 11.3 – La cartographie des processus de SDIE-Sylis

11.4.1 Le mapping opérationnel ISO 9001v2000 - CMMI

En se basant sur les résultats du mapping, présentés dans la section 9.2.3.5 et plus en détail dans l'annexe C, nous avons mis en place une application permettant la navigation entre les deux standards ISO 9001v2000 et CMMI afin de permettre de trouver la correspondance entre les pratiques des deux normes.

Les figures 11.4 et 11.5 montrent que l'application permet :

- pour chaque pratique de CMMI de trouver les différents chapitres d'ISO 9001v2000 qui la couvrent ;
- pour chaque chapitre d'ISO 9001v2000 de trouver les différentes pratiques de CMMI qui le couvrent.

11.4.2 Une cartographie unique des processus

À partir de la cartographie des processus (présentée dans la section 11.4) et du mapping entre ISO 9001v2000 et CMMI (présenté dans la section 11.4.1), on a pu analyser le domaine d'intervention de CMMI au sein du référentiel entreprise et les améliorations que le modèle pouvait apporter.

La figure 11.6 montre une cartographie unique des processus d'entreprise. Cette figure présente le niveau d'intégration des domaines de processus de CMMI dans une cartographie ISO 9001v2000. Ainsi on observe que les domaines de processus :

- DAR : Résolution et Analyse pour Décision, niveau 3 ;
- IT : Équipe intégrée, niveau 3 ;
- OEI : Environnement organisationnel en vue de l'intégration, niveau 3 ;
- ISM : Gestion de fournisseur intégrée, niveau 3 ;
- RSKM : Gestion des Risques, niveau 3.

ne sont pas pris en compte par ISO 9001v2000. Ces domaines de processus sont à implémenter en intégralité pour être conforme à CMMI. Le reste des domaines de processus de CMMI, et de manière localisée par rapport aux processus ISO 9001v2000, est traité en partie ou en totalité. Pour implémenter ces derniers domaines de processus, il faut se référer au mapping présenté dans l'annexe C pour intégrer les pratiques non couvertes par ISO 9001v2000.

Mapping chapitre ISO9001v2000 vers pratiques CMMI

SYLIS
IT & Business solutions

Mapping ISO9001v2000-CMMI

Sélectionnez un chapitre ISO 9001 v2000 : 7.4.1

résultat du Mapping :

ML	PA	Pratique	Correlateur	Remarques
ML 2	SAM	GP2.6	S	
ML 2	SAM	GP2.9	S	
ML 2	SAM	SP1.1	S	
ML 2	SAM	SP1.2	S	
ML 2	SAM	SP2.2	M	CMMI est plus faible.
ML 2	SAM	SP2.3	S	
ML 3	PI	SP3.1	S	
ML 3	TS	SP1.1	S	
ML 3	TS	SP1.2	S	
ML 3	TS	SP1.3	S	
ML 3	TS	SP2.4	S	

Figure 11.4 – Le résultat du mapping d’ISO 9001v2000 vers CMMI

Mapping pratique CMMI vers chapitres ISO9001 v2000

Sélectionnez un niveau de maturité : **ML 2**

Sélectionnez un domaine de processus : **CM**

Sélectionnez une pratique : **SP11**

Intitulé du domaine de processus : Gestion de configuration

Titre de la pratique : Identifier les articles de configuration.

Description de la pratique : Identifier les articles de configuration, les composants et les produits associés qui seront gérés en configuration.

Résultat du Mapping :

Chapitre ISC	Titre du chapitre	Corrélateur	Remarques
4.2.3	Maîtrise des documents	S	
4.2.4	Maîtrise des enregistrements	S	
7.3.7	Maîtrise des modifications de la conception et du développement	S	
7.5.3	Identification et traçabilité	S	
8.3	Maîtrise du produit non conforme	M	Pas explicitement exigé par CMMI.

Figure 11.5 – Le résultat du mapping de CMMI vers ISO 9001v2000

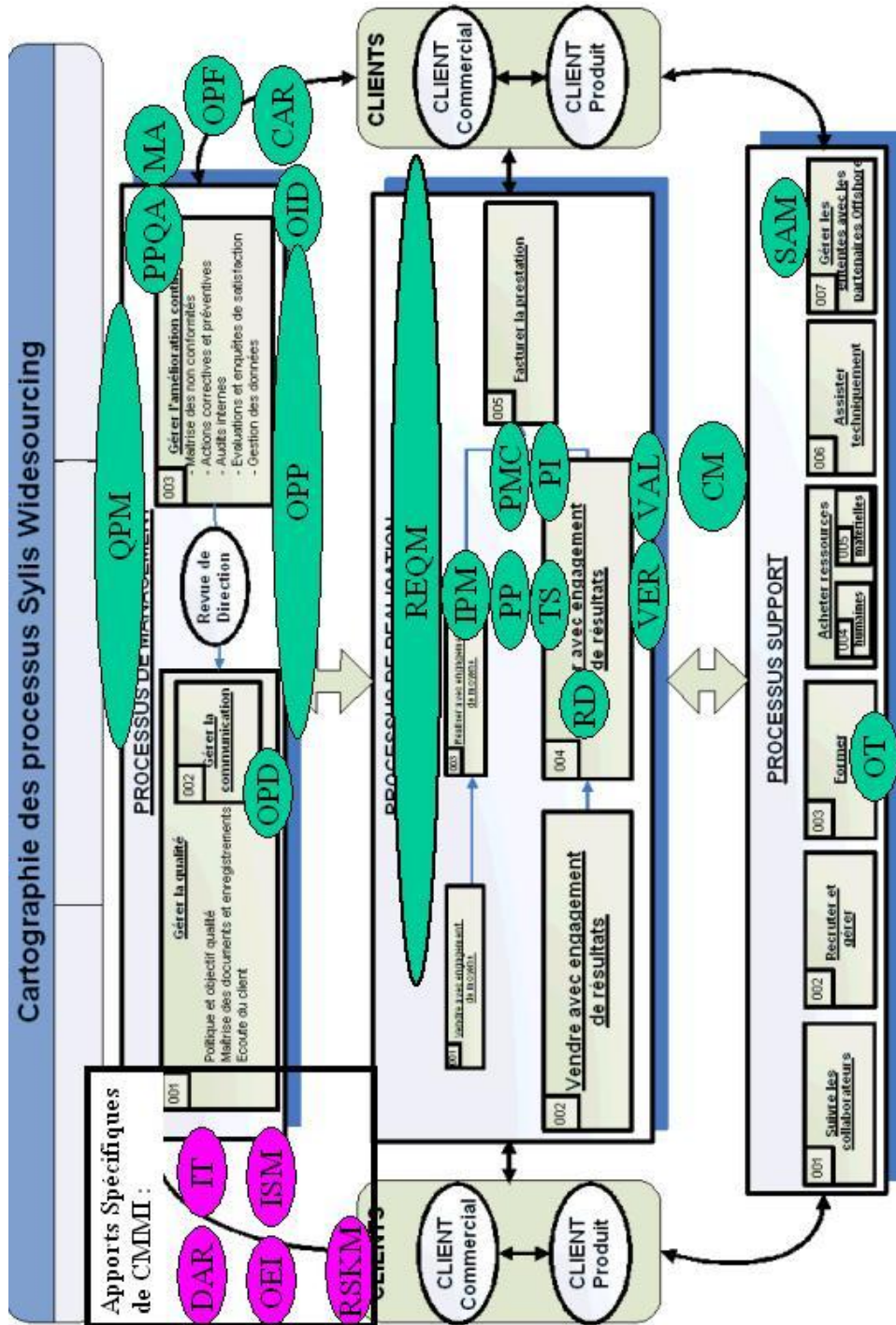


Figure 11.6 – Le niveau d'intégration des domaines de processus de CMMI dans une cartographie de processus ISO 9001v2000

Concernant les processus ISO de réalisation : Le processus ISO "réaliser avec engagement de résultat" traite 7 PAs :

- TS, Solution Technique, niveau 3 ;
- PI, Intégration du produit, niveau 3 ;
- VER, Vérification, niveau 3 ;
- VAL, Validation, niveau 3 ;
- PP, Planification de Projet, niveau 2 ;
- PMC, Maîtrise et suivi de projet, niveau 2 ;
- IPM, Gestion de Projet Intégré, niveau 3.

Ces PAs sont répartis parmi les différentes procédures du processus ISO "réaliser avec engagement de résultat", une procédure pouvant traiter de plusieurs PAs.

REQM, Gestion des exigences et RD, Développement des exigences répondent directement de la volonté d'ISO 9001v2000 de satisfaire le client et donc de répondre à ses exigences. Ces 2 PAs sont traités principalement par le chapitre 7 : les processus de réalisation ISO.

Concernant les processus ISO de management : QPM, Gestion Quantitative de la Qualité et des Processus (niveau 4) et OPP Performances des Processus de l'Organisation (niveau 4) sont traités conjointement par les processus ISO "gérer la qualité" et "gérer l'amélioration continue".

Le processus ISO "Gérer la Qualité" traite aussi du PA OPD, Définition des Processus de l'Organisation, niveau 3.

En plus de OPP et QPM, le processus ISO "Gérer l'amélioration continue" regroupe :

- MA, Mesure et Analyse, niveau 2 ;
- PPQA, Assurance qualité produit et processus, niveau 2 ;
- OPF, Focalisation de l'Organisation sur les Processus, niveau 3 ;
- OID, Déploiement et Amélioration des Processus Innovation de la technologie des processus de l'Organisation, niveau 5 ;
- CAR, Résolution et Analyse Causale, niveau 5.

Concernant les processus ISO de support : Le processus ISO "Former" correspond au PA OT, Formation dans l'Organisation, niveau 3.

Les processus ISO "Acheter des ressources humaines" et "gérer les ententes avec les fournisseurs Offshore" traitent du PA SAM, Gestion des ententes avec les fournisseurs, niveau 2.

11.4.3 CMMI et les processus implémentés

La figure 11.7 montre le niveau d'intégration des recommandations de CMMI dans les processus mis en place. Cette figure montre aussi un exemple où on voit apparaître les domaines

de processus de CMMI dans la définition d'un processus donné.

11.4.4 Un exemple de processus mis en place : PROCES-REAL-001 Processus Vendre avec engagement de moyens

Finalité du processus

Ce processus, déclenché par la réception d'un appel d'offres du client, a pour finalité de positionner les moyens humains en adéquation avec les demandes des compétences des clients, organiser la réponse et sa contractualisation.

Risques du processus

- la demande n'est pas enregistrée ;
- l'appel d'offres est mal qualifié ;
- la réponse n'est pas en adéquation avec la demande ;
- la fiche de mission n'est pas rédigée ;
- le contrat n'est pas signé.

Indicateurs de performance du processus

- améliorer la réactivité et les délais de réponses : Nombre de réponses dans les délais demandés / Nombre de réponses totales ;
- améliorer le taux de réponses : Nombre de réponses / Nombre de demandes reçues ;
- améliorer les propositions faites au client : Nombre d'entretiens / Nombre de demandes reçues (pour une demande client reçue, présenter n collaborateurs équivaut à un entretien) ;
- améliorer la réussite : Nombre d'affaires gagnées / Nombre de demandes reçues.

Ressources humaines concernées

- manager (MNG) ;
- directeur / Chef de projet (DP) ;
- collaborateur (CO).

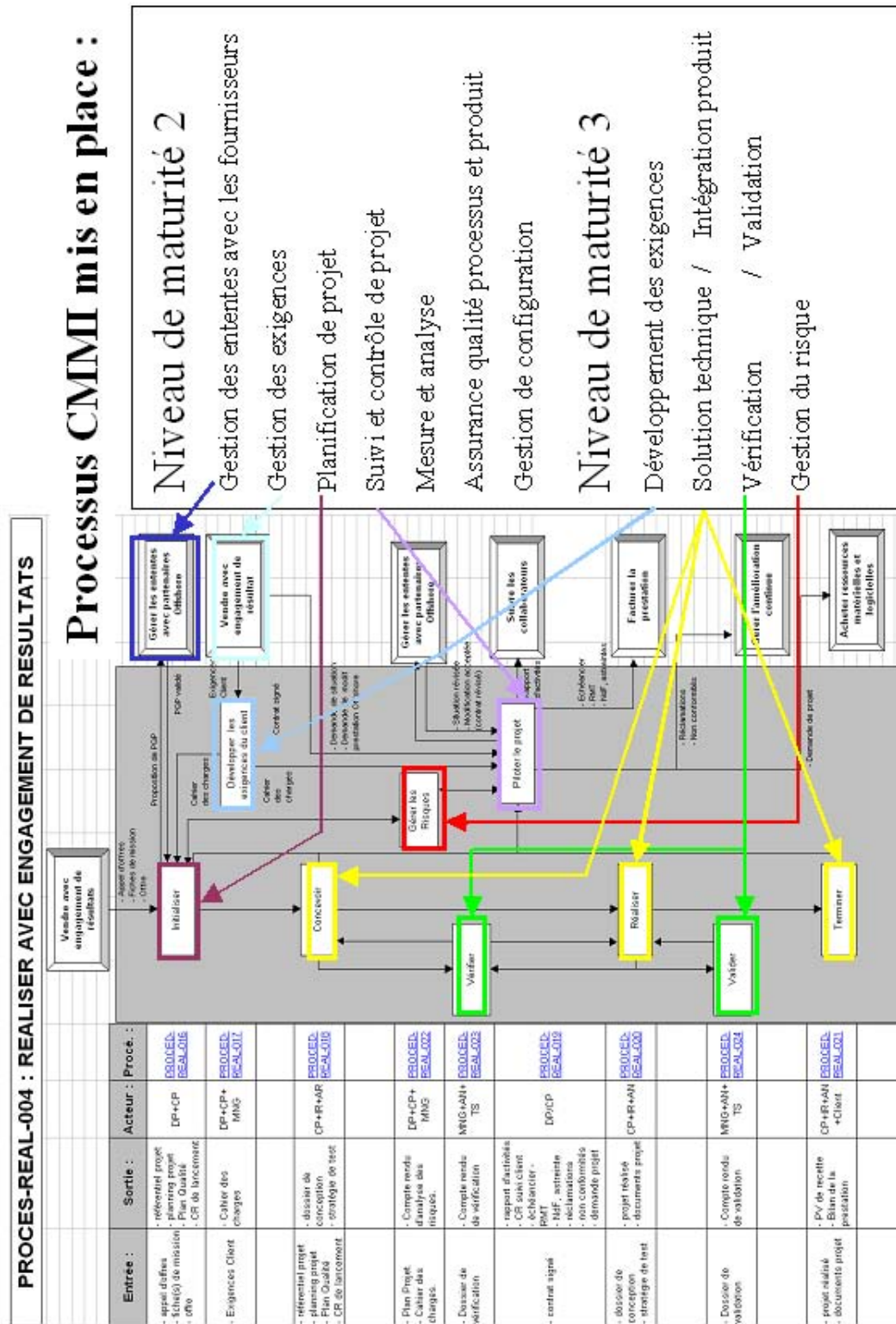


Figure 11.7 – Le niveau d’intégration des domaines de processus de CMMI dans un processus

Autres processus utilisés

- assister techniquement ;
- gérer les ententes avec les partenaires Offshore ;
- facturer la prestation ;
- réaliser avec engagement de moyens ;
- acheter les ressources humaines.

Déroulement du processus

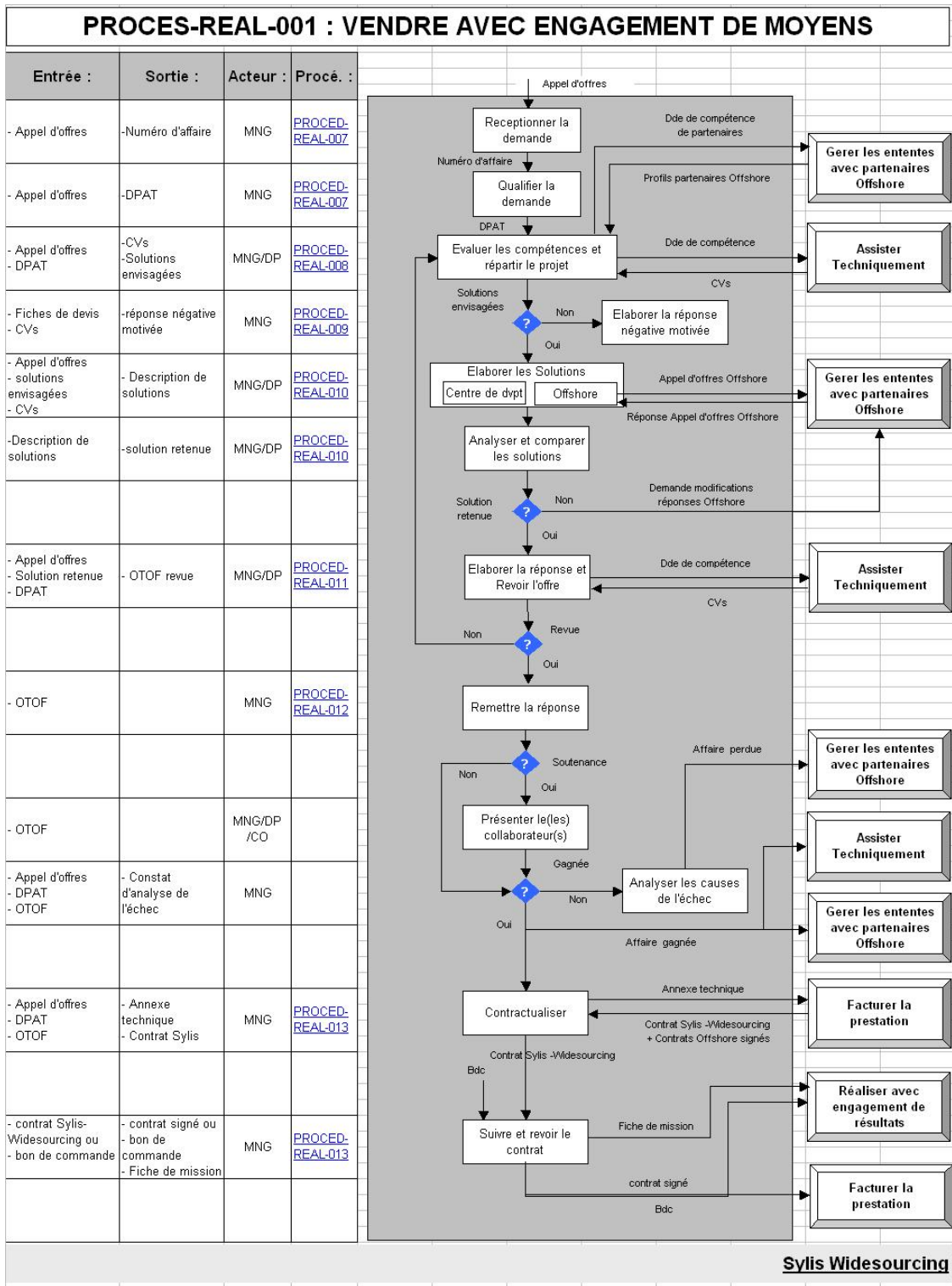


Figure 11.8 – PROCES-REAL-001 Processus Vendre avec engagement de moyens.

Les PAs de CMMI pris en compte

Niveau de maturité	Sigle	PA	Processus et Procédures concernés
ML 2	REQM	Gestion des exigences	
	PP	Planification de projet	PROCED-REAL-008 Evaluer les compétences et répartir le projet, PROCED-REAL-010 Elaborer les solutions, PROCED-REAL-011 Elaborer la réponse et revoir l'offre, PROCED-REAL-013 Contractualiser et revoir contrat.
	PMC	Maîtrise et suivi de projet	
	SAM	Gestion des ententes avec les fournisseurs	
	CM	Gestion de la configuration	
	PPQA	Assurance qualité produit et processus	PROCES-REAL-001 Processus Vendre avec engagement de moyens.
	MA	Analyses et mesures	PROCES-REAL-001 Processus Vendre avec engagement de moyens.
	RD	Développement des exigences	
	TS	Solution technique	
	PI	Intégration de produit	
	VER	Vérification	
	VAL	Validation	
	OPF	Focalisation de l'organisation sur les processus	PROCES-REAL-001 Processus Vendre avec engagement de moyens.
ML 3	OT	Formation dans l'organisation	
	OPD	Définition des processus de l'organisation	Référentiel Qualité WideSourcing.
	ISM	Gestion de fournisseur intégrée	PROCED-REAL-008 Evaluer les compétences et répartir le projet, PROCED-REAL-010 Elaborer les solutions.
	IPM	Gestion de projet intégré	PROCED-REAL-010 Elaborer les solutions, PROCED-REAL-011 Elaborer la réponse et revoir l'offre.
	IT	Équipe intégrée	PROCED-REAL-008 Evaluer les compétences et répartir le projet, PROCED-REAL-010 Elaborer les solutions, PROCED-REAL-011 Elaborer la réponse et revoir l'offre.
	RSKM	Gestion des risques	
	DAR	Résolution et analyse pour décision	PROCED-REAL-008 Evaluer les compétences et répartir le projet, PROCED-REAL-010 Elaborer les solutions, PROCED-REAL-011 Elaborer la réponse et revoir l'offre.
	OEI	Environnement organisationnel en vue de l'intégration	PROCED-REAL-011 Elaborer la réponse et revoir l'offre.

Les procédures associées

Les procédures associées au processus "PROCES-REAL-001 Processus Vendre avec engagement de moyens" sont :

- PROCED-REAL-007.01 Réceptionner et qualifier la demande : Cette procédure a pour objet de décrire les étapes à suivre par les responsables de tâches afin de mener à bien l'opération de réception et de qualification d'une demande d'un client ;
- PROCED-REAL-008.01 Evaluer les compétences et répartir le projet : Cette procédure a pour objet de décrire les opérations à effectuer par les responsables de tâches afin de mener à bien l'évaluation du besoin lié à la demande de moyens et répartir le projet entre Widesourcing et les partenaires Nearshore/Offshore ;
- PROCED-REAL-009.01 Elaborer la réponse négative motivée : Cette procédure a pour objet de décrire les étapes à suivre et les responsables des tâches pour mener à bien l'opération d'élaboration d'une "Réponse Négative Motivée" qui fait suite à la procédure "Evaluer les compétences et répartir le projet" ;
- PROCED-REAL-010.01 Elaborer les solutions : Cette procédure a pour objet de décrire les opérations à réaliser par les responsables de tâches afin d'élaborer les solutions Centre de développement et Offshore envisagées ;
- PROCED-REAL-011.01 Elaborer la réponse et revoir l'offre : Cette procédure a pour objet de décrire les opérations à réaliser par les responsables de tâches afin de mener à bien l'élaboration de l'offre, sa revue pour envoi au Client ;
- PROCED-REAL-012.01 Remettre la réponse : Cette procédure a pour objet de remettre l'OTOF revue au client et d'informer les partenaires non retenus ;
- PROCED-REAL-013.01 Contractualiser et revoir contrat : Cette procédure a pour objet de décrire les opérations à réaliser par les responsables de tâches afin de mener à bien la contractualisation, le suivi et la revue de contrat.

11.5 Plan d'amélioration du référentiel Qualité

En appliquant un modèle de support d'amélioration continue IDEALSM (relatif à la mise en place de CMMI), on essaie à travers cette étape de valider les processus et procédures spécifiques à l'activité SDIE et à organiser les prochaines étapes.

11.5.1 Examen du système qualité

11.5.1.1 Point sur les décisions prises

Proposition d'amélioration : Établir les processus spécifiques à l'activité de SDIE :

- PROCES-REAL-001 Processus Vendre avec engagement de moyens ;
- PROCES-REAL-002 Processus Vendre avec engagement de résultats ;
- PROCES-REAL-003 Processus Réaliser avec engagement de moyens ;
- PROCES-REAL-004 Processus Réaliser avec engagement de résultats ;
- PROCES-SUPP-007 Processus Gérer les ententes avec les partenaires Off-shore.

Décision : Obtenir une première validation sur l'adéquation des processus avec le fonctionnement réel et opérationnel auprès du Manager. La validation finale sera réalisée auprès de la direction.

Résultat : À ce jour, la réalisation des processus, la validation opérationnelle et la validation par la direction ont été achevées.

11.5.1.2 Point de fonctionnement du Système de Management de la Qualité ISO 9001v2000 - CMMI

- Faire apparaître les recommandations CMMI prises en compte dans le référentiel qualité et les processus de WideSourcing.

Propositions d'amélioration :

- pour les recommandations CMMI prises en compte : Noter dans quels processus on les voit apparaître et à quels degrés ils sont couverts ;
- pour les recommandations CMMI du niveau 2 et 3 non encore prises en compte : Faire des propositions pour les intégrer dans l'activité du WideSourcing.

- Faire apparaître les outils, documents et méthodes utilisés au sein du WideSourcing.

Propositions d'amélioration :

- faire le lien entre le système documentaire mis en place (l'ensemble des outils, documents types et méthodes) et les processus ;
- faire apparaître à quel(s) moment(s) les outils, documents types et méthodes de WideSourcing sont utilisés au niveau des processus et des procédures.

- Processus "Vendre avec engagement de moyens" et Processus "Réaliser avec engagement de moyens" : Les processus sont appliqués par l'ensemble des équipes de SDIE. Ils correspondent à un fonctionnement réel et opérationnel.

Pour la vente avec engagement de moyens, il faut aller plus dans le détail pour montrer la différence avec la vente avec engagement de résultats.

Pour la réalisation avec engagement de moyens, il faudrait :

- ajouter un indicateur pour la satisfaction du client ;
- montrer la tâche de préparation de l'environnement et celle d'échange avec le client sur les outils à utiliser. (Ceci pour le démarrage de la mission).

Propositions d'amélioration :

- montrer la différence entre les ventes avec engagement de moyens et résultats ;
- ajouter un indicateur de mesure de la satisfaction du client ;
- retravailler la procédure de démarrage de la réalisation avec engagement de moyens.

- Processus "Vendre avec engagement de résultats" et Processus "Réaliser avec engagement de résultats" : Les processus sont appliqués par l'ensemble des équipes WideSourcing. Ils correspondent à un fonctionnement réel et opérationnel.

Pour la vente avec engagement de résultats, il faut retravailler la préparation et la soutenance de la réponse afin d'intégrer les attentes du client et les réponses de Syllis.

Pour la réalisation avec engagement de résultats, il faut plus de précision pour la procédure d'initialisation du projet et intégrer les différents types de PQP (PQP SDIE-Partenaires Offshore, PQP SDIE-client Produit et PQP client produit-client commercial).

Propositions d'amélioration :

- intégrer les attentes du client lors de la soutenance de la réponse WideSourcing ;
- intégrer les différents types de PQP lors de l'initialisation de la réalisation du projet avec engagement de résultats.

- Processus "Gérer les ententes avec les partenaires Offshore" : Le processus est appliqué par l'ensemble des équipes de SDIE. Il correspond à un fonctionnement réel et opérationnel.

11.5.2 Synthèse d'appréciation

Les améliorations identifiées lors de cette revue tendent à montrer l'industrialisation de l'activité de SDIE avec des processus, des procédures et des outils bien identifiés.

Le système correspond au fonctionnement de SDIE. La prochaine revue doit permettre de valider les améliorations et d'étudier les résultats des différents indicateurs.

11.6 Le référentiel documentaire

Le référentiel projet a pour but de créer une structure type de répertoire des projets. Cette structure contient tous les modèles de documents utilisables tout le long du projet. Ce

répertoire type est dupliqué pour chaque projet. Les objectifs du "référentiel documentaire" sont :

- créer une structure unique des répertoires projet ;
- créer un ensemble de modèles de documents utilisable dans tous les projets ;
- standardiser les documents et les méthodes utilisés ;
- accélérer la phase de production des documents ;
- partager les connaissances et les références ;
- s'assurer du respect du référentiel qualité ;
- préparer et faciliter la localisation des preuves d'évaluation et de certification des normes qualité prises en compte dans le référentiel qualité.

La figure 11.9 montre un aperçu du référentiel documentaire mis en place. Ce référentiel répond au besoin d'organiser l'entreprise et surtout d'industrialiser son activité. Il correspond à la description des activités faite dans le référentiel qualité et couvre toutes les étapes d'un projet :

- 1) étude d'opportunité ;
- 2) définition des besoins ;
- 3) définition de l'architecture ;
- 4) spécification fonctionnelle ;
- 5) spécification technique ;
- 6) réalisation ;
- 7) recette ;
- 8) déploiement - mise en production ;
- 9) vérification en service régulier ;
- 10) maintien en service régulier.

À chacune de ces étapes, correspond un certain nombre de documents standards qui doivent être reproduits dans chaque projet. La figure 11.10 montre l'ensemble des documents relatif à la phase de recette.

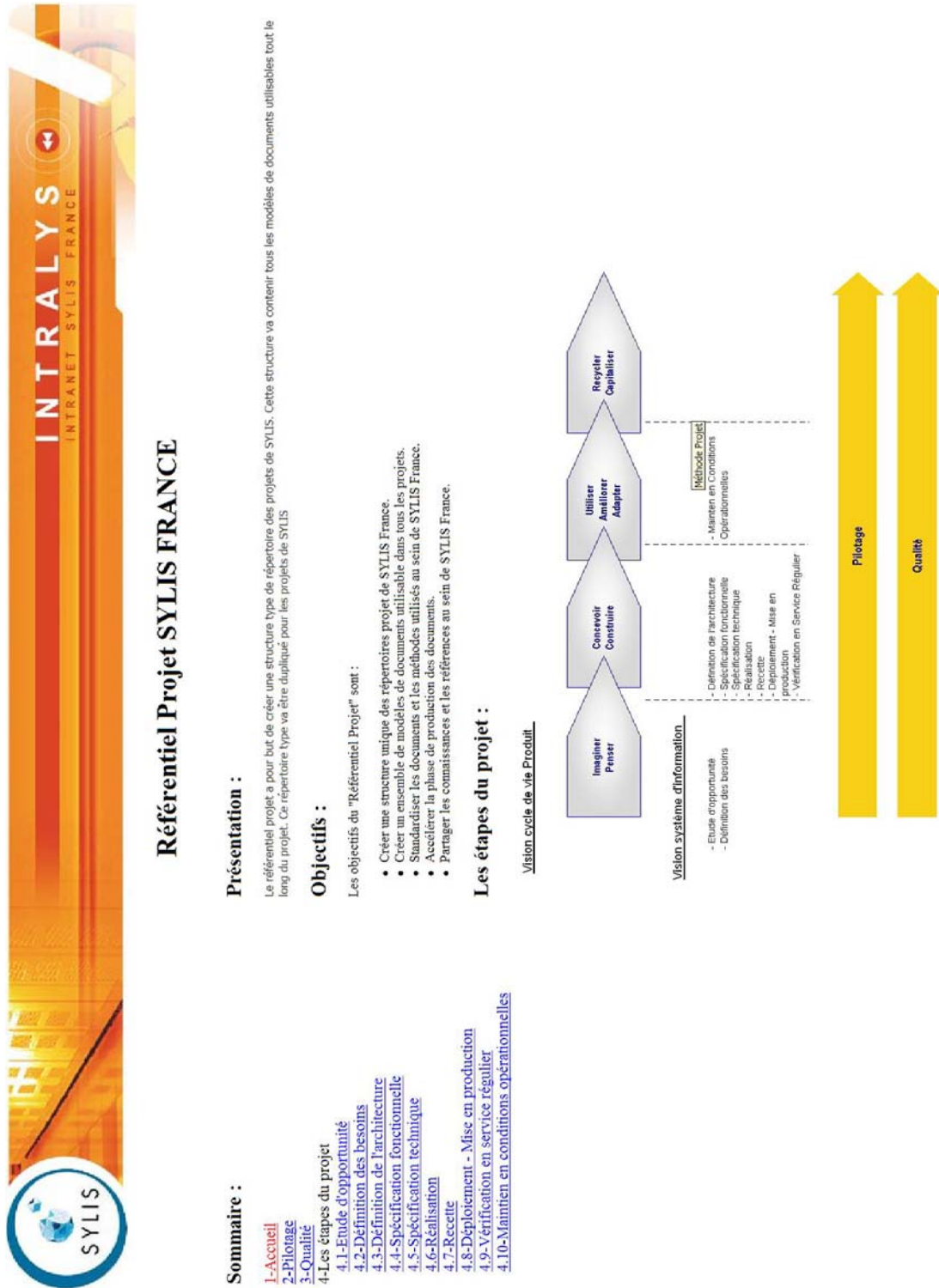



Figure 11.9 – Le référentiel documentaire implémenté au sein de Syllis



Sommaire :

- 1-Accueil
- 2-Pilotage
- 3-Qualité
- 4-Les étapes du projet
 - 4.1-Etude d'opportunité
 - 4.2-Définition des besoins
 - 4.3-Définition de l'architecture
 - 4.4-Spécification fonctionnelle
 - 4.5-Spécification technique
 - 4.6-Réalisation
 - 4.7-Recette
 - 4.8-Déploiement - Mise en production
 - 4.9-Vérification en service régulier
 - 4.10-Maintien en conditions opérationnelles

L'objectif :

L'objectif de la recette est de permettre au client et à SYLIS de vérifier la correspondance entre les spécifications fonctionnelles et techniques et ce qui a été effectivement produit. Il s'agit donc de planifier et d'exécuter des ensembles de tests qui, finalement garantisent au client que le produit qui lui est livré est un produit fiable, robuste, et qui surtout fonctionne conformément à ses attentes.

Les tâches :

- La recette comprend :
 - La recette fonctionnelle
 - La recette d'intégration
- Définir le périmètre de test. Il s'agit de savoir globalement sur quoi vont porter les tests. Ce périmètre est défini en amont, et porte sur les grandes parties d'un système ou d'une application.
- La définition des lots doit être faite dans le plan des recettes. Un lot est une partie plus ou moins grande du système final, livrée par SYLIS pour être testée.
- Rédiger le plan de recette, qui a une valeur contractuelle (entre SYLIS et le client), de tous les tests qui seront passés, dans quel ordre, ce sur quoi ils porteront, de quel façon ils seront effectués et avec quels moyens. Le plan de test doit recenser avec une grande précision les scénarii de test qui seront effectués sur chaque lot. Il doit s'agir d'un recensement exhaustif.
- Définir le protocole de la recette. Il a pour objet la définition des modalités d'organisation de la recette. Il précise les responsabilités des différents acteurs du projet et les livrables attendus de chacun, les interactions entre ces acteurs, les procédures ou règles à respecter au cours de la phase de recette. Les principaux points définis dans le protocole de recette:
 - Le pilotage et le suivi de la recette: Il faut définir les différents types de comités qui seront tenus, leur fréquence, les participants et les points qui doivent y être vus systématiquement.
 - Les rôles et les responsabilités des acteurs : Les rôles et les responsabilités sont clairement attribués. Il doit être clair pour tous les intervenants de ce qui incombe de leur responsabilité, et l'intégralité des tâches qu'on attend d'eux.

Les documents :

- plan des recettes.doc
- anomalie projet.doc
- registre anomalies projet.xls
- procès verbal de livraison.doc

Figure 11.10 – L'étape de recette du référentiel documentaire

11.7 Audit projet

L'audit projet permet de vérifier la bonne implémentation du référentiel qualité au sein des projets. Comme le montre la figure 11.11, on définit un certain nombre de points de contrôle tout le long du projet. La méthodologie d'audit est structurée de la façon suivante :

les audits : Ce sont des constats de faits qui permettent d'évaluer les risques et de les anticiper. Ils permettent de s'assurer au cours des phases du cycle que les éléments de décision pour le passage à la phase suivante sont requis. Les auditeurs ne sont pas forcément des experts techniques mais surtout des collaborateurs expérimentés et sensibilisés aux bonnes pratiques :

- 1) AU 0 : Audit de proposition commerciale ;
- 2) AU 1 : Audit de Lancement de projet ;
- 3) AU 2 : Audit de Spécification ;
- 4) AU 3 : Audit de Conception ;
- 5) AU 4 : Audit de Réalisation/Tests Unitaires ;
- 6) AU 5 : Audit d'Intégration/Validation ;
- 7) AU 6 : Audit de Livraison/Installation ;
- 8) AU 7 : Audit de Recette ;
- 9) AU 8 : Audit de clôture de projet ;
- 10) AU PIL : Audit de pilotage.

les revues : Elles permettent de s'assurer au cours des phases du cycle que les éléments produits le sont en respect du besoin client, de l'expertise attendue et du formalisme souhaité. Les responsables des revues sont les pairs ont déjà une expérience techniques.

- 1) RE SC = Revue de Spécification/Conception ;
- 2) RE RTU = Revue de Réalisation/Tests Unitaires ;
- 3) RE IV = Revue d'Intégration/Validation ;
- 4) RE LI = Revue de Livraison/Installation.

11.8 Le référentiel des compétences

Pour réaliser notre référentiel de compétences, nous nous sommes appuyés sur les travaux publiés par l'École de Management des Systèmes d'Information de Grenoble en 2007 [39].

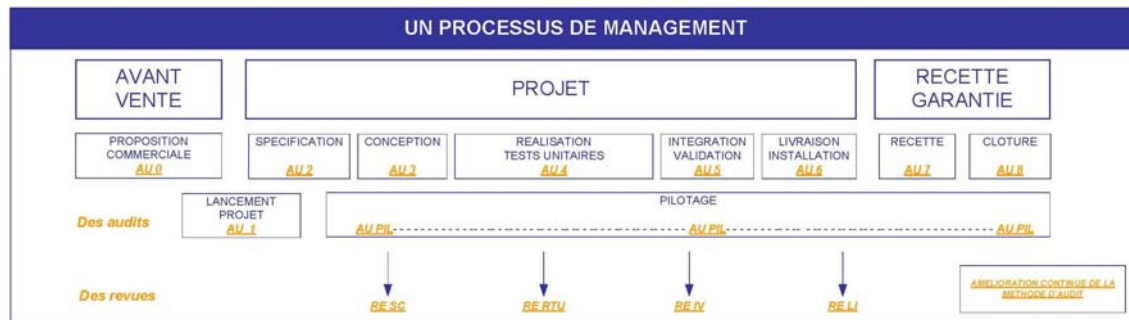


Figure 11.11 – Les étapes de la méthodologie d’audit projet

Les domaines de compétences

On peut regrouper l’ensemble des compétences nécessaires à l’exercice des métiers des systèmes d’information (SI) dans quatre grands domaines de compétences :

- infrastructure et architecture ;
- conception et exploitation ;
- management et pilotage ;
- marché et gestion des fournisseurs.

Ces cinq domaines permettent de couvrir l’ensemble des missions dédiés aux métiers des SI, à savoir les missions liées à la stratégie métier de l’entreprise, aux services et support aux métiers, aux projets et leur management et aux aspects techniques et technologiques.

Compétences métier

Les domaines de compétences offrant une vision trop macroscopique, chacun d’entre eux a été décliné en un ensemble de compétences-métier dont 14 ont été identifiés comme essentielles pour l’exercice de l’activité d’une entreprise :

- infrastructure et architecture :
 - architecture applicative ;
 - architecture technique ;
 - architecture fonctionnelle/processus métier ;
 - télécoms et réseaux.
- conception et exploitation :
 - conception développement ;
 - gestion des données/bases de données ;
 - production/exploitation ;

- sécurité informatique.
- management et pilotage :
 - management et conduite de projets ;
 - pilotage et gouvernance ;
 - audit/qualité/méthode.
- marché et gestion des fournisseurs :
 - gestion des fournisseurs et achat ;
 - connaissance du marché informatique et télécoms.

Comme le montre la figure 11.12, chaque compétence métier est un ensemble de connaissance, savoir-faire, capacités relationnelles et cognitives, nécessaires pour exercer une "facette" d'un métier. Chaque élément d'une compétence-métier est appelé une "unité de compétence". Il y a quatre unités de compétences :

- les connaissances et savoirs théoriques ou pratiques qu'il est nécessaire de posséder ;
- les savoir-faire ou pratiques à mettre en œuvre pour l'exercice d'une compétence métier ;
- les capacités relationnelles, c'est-à-dire l'ensemble des capacités à utiliser dans la relation avec les autres et notamment dans une situation de management ;
- les capacités cognitives, c'est-à-dire l'ensemble des capacités personnelles ou dispositions utiles pour la réalisation d'une compétence métier.

Chaque unité de compétence est déclinée sous 5 niveaux définis comme suit :

notion : c'est le niveau de base qui permet de comprendre une compétence métier ;

application : la compétence métier peut être exercée dans un périmètre réduit et maîtrisé ;

maîtrise : la compétence métier est parfaitement maîtrisée dans un périmètre restreint mais avec une liberté d'action ;

spécialiste : la compétence métier est parfaitement maîtrisée mais sur l'ensemble de l'entreprise.

expert : le domaine de la compétence métier s'étend au-delà de l'entreprise. Il y a reconnaissance extérieure de cette compétence-clé.

La figure 11.13 montre l'exemple de la compétence métier audit/qualité/méthode, implémenté et conforme au tableau de référence (voir la figure 11.12).

11.8. LE RÉFÉRENTIEL DES COMPÉTENCES

Niveaux de compétences (tableau de référence)					
	1	2	3	4	5
	Notion	Notions + ↓ = Application	Application + ↓ = Maîtrise	Maîtrise + ↓ = Spécialiste	spécialiste + ↓ = Expert
Connaissances savoirs théoriques ou pratiques qu'il est nécessaire de posséder pour réaliser une compétence-clé	Connaissances de base	Connaissances du sujet limitées aux processus, méthodes ou outils principaux le concernant	Bonne connaissances de toutes les facettes du sujet de ses processus, méthodes ou outils le concernant	Haut niveau de connaissances dans tous les aspects du sujet, de tous ses processus, méthodes ou outils	Connaissances approfondies du sujet sur tous les aspects, internes ou externes à l'entreprise. Capacité à contribuer efficacement dans l'amélioration des standards utilisés
Savoir-faire Pratiques à mettre en œuvre pour l'exercice d'un savoir	Savoir-faire généraux, pas de savoir-faire particulier sur le sujet	Savoir-faire suffisants pour traiter les activités récurrentes du sujet, appliquer une procédure sans initiative	Savoir-faire suffisants pour traiter les aspects imprévus du sujet et les cas d'exception	Grande expérience sur le sujet. Est considéré dans l'entreprise comme un spécialiste du sujet	Fait preuve de savoir-faire de haut niveau reconnu au sein mais aussi en dehors de l'entreprise
Sociales / relationnelles Ensemble de capacités d'action à utiliser dans la relation avec les autres et notamment dans une situation de management	Pas de capacité et responsabilité particulière	Responsabilités limitées aux instructions basiques sur un périmètre connu avec des équipes débutantes sur le sujet	Donne des instructions dans des procédures et processus maîtrisés. Est capable de former à minima aux concepts de base	Est en mesure de développer et mettre en œuvre des contenus et des méthodes pour des actions de coaching, de conseil ou de management en interne à l'entreprise	Est en mesure de définir et appliquer des contenus et des méthodes, notamment en management, qui ont des impacts au-delà de l'entreprise
Cognitives Ensemble de capacités personnelles et dispositions utiles pour la réalisation une compétence-clé	Tout ce qui permet d'accomplir des tâches simples sous contrôle	Tout ce qui permet d'accomplir des tâches standard sous contrôle léger sans sortir du cadre défini	Tout ce qui permet d'accomplir des tâches de manière autonome pour un travail prédefini. Peut sortir du cadre lorsque c'est nécessaire	Tout ce qui permet d'accomplir des tâches dans des situations complexes, d'extrapoler, de les faire évoluer, voire de les modifier profondément dans le cadre de l'entreprise	Tout ce qui permet d'accomplir des tâches dans des situations complexes, d'anticiper et d'apporter des changements, y compris hors du cadre de l'entreprise

Figure 11.12 – Le tableau de référence des compétences métier [39]

Audit / Qualité / méthode									
1		2		3		4		5	
<p>Connaissances</p> <ul style="list-style-type: none"> - Connaissance théorique des certifications, des normes et des procédures qualité mises en œuvre dans l'entreprise 		<ul style="list-style-type: none"> - Sur un périmètre restreint. Connaît les normes et procédures qualité en vigueur - Connaissance des méthodes et normes de développement des langages les plus courants 		<ul style="list-style-type: none"> - Bonne connaissance du plan d'action qualité mis en œuvre sur son domaine - Connaissance des certifications qualité - Connaissance des principaux systèmes d'exploitation 		<ul style="list-style-type: none"> - Très bonne connaissance des clients de la DSI et de son environnement - Connaissance des architectures de SI - Connaissance des techniques de conduite de projet 		<ul style="list-style-type: none"> - Connaissance de la stratégie de l'entreprise et de la politique qualité de l'entreprise - Connaissance approfondie de l'évolution des normes et certifications - Connaît la limite des méthodes 	
<p>Savoir-faire</p> <p>Non qualifiable</p>		<ul style="list-style-type: none"> - Sait tester et mettre en place une norme ou méthode adoptée 		<ul style="list-style-type: none"> - Sait réaliser des programmes de test pour valider l'application des procédures - Analyse les dysfonctionnements et les problèmes de traçabilité - Sait rédiger des référentiels qualité ou supports correspondants 		<ul style="list-style-type: none"> - Sait réaliser des audits des organisations et des méthodes de gestion de projet - Sait analyser une situation pour améliorer les processus - Définit, met en place et suit un tableau de bord qualité - Peut manager une équipe de qualifiants - Peut prendre en charge les problématiques de sécurité 		<ul style="list-style-type: none"> - Définit le schéma d'organisation, des méthodes de travail, des normes et procédures qualité - Évalue la rentabilité des techniques mises en place - Adapte les méthodes aux évolutions de l'entreprise et procédures - Assure une veille dans l'environnement des normes - Participe à des colloques, forums, groupes de travail à l'extérieur de l'entreprise 	
<p>Sociales / relationnelles</p> <p>Non qualifiable</p>		<p>Non qualifiable</p>		<ul style="list-style-type: none"> - Sensibilise les utilisateurs aux problèmes qualité - Forme les équipes SI sur les référentiels 		<ul style="list-style-type: none"> - Fournit assistance et conseil aux acteurs dans le choix des méthodes - Entretien une relation permanente avec les utilisateurs - Alerte en cas d'écart 		<ul style="list-style-type: none"> - Informe les différents acteurs de l'entreprise sur les évolutions - Interface reconnue des experts externes à l'entreprise ou des autres domaines de compétence - Publie sur le domaine, en interne ou externe à l'entreprise 	
<p>Cognitives</p> <p>Non qualifiable</p>		<ul style="list-style-type: none"> - Rigueur et méthodologie dans l'application des procédures - Adaptabilité 		<ul style="list-style-type: none"> - Aisance rédactionnelle - Qualité d'analyse, notamment pour trouver des pistes d'amélioration - Qualités pédagogiques pour expliquer les procédures mises en place 		<ul style="list-style-type: none"> - Persévérance et opiniâtreté pour arriver au résultat escompté - Qualité d'écoute et de communication vis-à-vis des acteurs de l'entreprise 		<ul style="list-style-type: none"> - Esprit de conviction - Ouverture d'esprit - Très grande clarté dans le discours 	

Figure 11.13 – La compétence métier audit/qualité/méthode

11.9 L'aspect humain

Étant donné que l'entreprise possède ses propres référentiels et méthodes, on craignait la non-utilisation de la démarche retenue. Et comme la mise en place du modèle s'apparente à une conduite du changement, nous avons commencé par traiter le problème d'acceptabilité que nous pouvons rencontrer. Ainsi, nous avons classé tout le personnel de l'entreprise en deux catégories :

Les alliés : Ce sont les personnes qui approuvent le travail et qui sont convaincues de son utilité et de la nécessité de le mettre en place au sein de l'entreprise. Cette catégorie regroupe surtout la direction de l'entreprise.

Les récalcitrants : Ce sont les personnes qui sont un petit peu hostile à la mise en place de la méthodologie parce qu'elle risque de bousculer leurs habitudes. Cette hostilité est due à deux raisons :

- les personnes n'utilisent ni de méthodes ni de référentiels ou ;
- les personnes ont leurs propres méthodologies et référentiels.

Nous avons concentré nos efforts sur cette deuxième partie du personnel, et comme le montre la figure 11.14, nous avons choisi la stratégie de la persuasion à travers des exposés qui présentent les avantages de la démarche retenue par rapport à l'ensemble des méthodologies existantes et en discutant avec eux pour savoir quels sont leurs attentes et s'il y a des points à modifier. Dès lors, l'attitude de cette catégorie a changé et ils se sont plus impliqués dans la mise en place du projet.

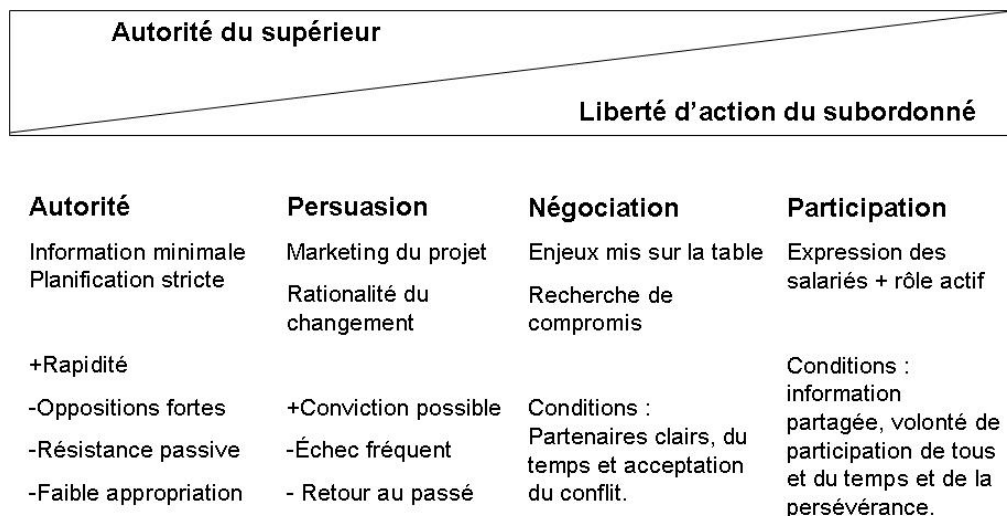


Figure 11.14 – Les différentes stratégies de conduite du changement

11.10 Conclusion

Ce chapitre nous a permis de présenter une application concrète de notre démarche d'intégration des standards qualité. Notre cas d'étude consistait en la création d'un référentiel qualité unique permettant d'intégrer les recommandations d'ISO 9001v2000 et CMMI. La mise en place du référentiel qualité est accompagné de la mise en place d'un référentiel documentaire, une méthodologie d'audit projet et d'un référentiel de compétences garants de la bonne implémentation du référentiel qualité au sein d'une entreprise.

SIXIÈME PARTIE

ÉPILOGUE

Conclusion générale et Perspectives

12.1 Rappel du cadre des travaux

Le cadre général de nos travaux porte sur l'intégration des processus métier et l'organisation d'entreprise.

Le croisement de concepts issus de la gestion des processus métier, des normes et standards qualité et de l'interopérabilité nous a permis de nous intéresser à l'organisation et l'intégration des processus métier d'entreprise, pour proposer une démarche de mise en place d'un référentiel qualité multi-vues.

Le but de notre travail est de montrer comment intégrer les processus métier d'une entreprise à l'aide d'un référentiel commun offrant différents points de vue. Cette démarche généralisable est appliquée à l'intégration de deux standards de qualité, ISO 9001v2000 et Capability Maturity Model Integration (CMMI), afin de générer un référentiel qualité multi-vues permettant une certification relative aux deux normes. Ce référentiel prend en compte les chapitres imposés par ISO et les recommandations de CMMI. Dans le cadre de l'implémentation du référentiel, nous nous sommes intéressés à la définition d'une méthodologie d'estimation des délais et charges des projets informatiques afin de rationaliser ce processus critique pour l'entreprise. La mise en place de ce référentiel qualité s'accompagne de la définition d'une démarche assurant l'interopérabilité des processus définis avec ceux des clients et/ou partenaires.

Pour répondre à la question d'assurer le respect du référentiel qualité au sein des projets, une méthodologie d'audit projet, un référentiel documentaire et un référentiel des compétences viennent compléter le travail déjà réalisé.

12.2 Synthèse sur les apports de ces travaux

Mise en place d'un référentiel qualité multi-vues

Dans cette thèse, nous avons proposé une méthodologie permettant d'intégrer des standards qualité afin de mettre en place un référentiel qualité multi-vues. Cette méthodologie a été appliquée afin d'étudier la faisabilité d'intégration et la réalisation d'un mapping des deux standards qualité, ISO 9001v2000 et Capability Maturity Model Integration (CMMI). Le référentiel qualité ainsi généré permet d'éviter le dédoublement du travail pour la mise en place ou encore la mise à jour de ce référentiel. En cas d'évaluation, il permet aussi de fournir les preuves à partir d'un même référentiel.

Solutions d'interopérabilité orientée clients et partenaires

Le travail réalisé consiste à définir une démarche d'interopérabilité des processus d'entreprise, définis dans le référentiel qualité, avec ceux de ses clients et partenaires. Notre proposition consiste à définir deux solutions d'interopérabilité, chaque solution est relative à un type d'interaction différent :

- les interactions avec le client : dans ce cas, la solution consiste à définir un "Référentiel documentaire" afin de standardiser les échanges de documents avec le client. Ce référentiel contient tous les modèles de documents utilisables tout le long du projet et échangeables avec le client ;
- Les interactions avec un partenaire ou un sous-traitant : dans ce cas de figure, un processus appelé "PROCES-SUPP-007 Processus Gérer les ententes avec les partenaires Offshore" a été créé afin de permettre le dialogue et l'interopérabilité avec les processus du partenaire. Tous les processus internes délèguent cette tâche à ce processus. Cette solution présente l'avantage que toutes les opérations concernant le partenaire soient concentrées dans un seul processus.

La méthodologie d'estimation des délais et durée des projets

Dans nos travaux, nous avons intégré les recommandations d'ISO 9001v2000 et CMMI afin de créer un référentiel qualité multi-vues. L'étape suivante a été de mettre en place des méthodologies pour développer les processus vitaux de l'entreprise. Dans cette partie, nous nous sommes focalisés sur un de ces processus vitaux pour l'organisation d'entreprise, à savoir l'estimation des délais et des charges des projets. Le modèle proposé est basé sur :

- les Points de Fonction, définie par *Dr. Allan Albrecht* au sein d'IBM et repris par le laboratoire *International Function Point Users Group (IFPUG)*, pour le calcul de la taille du projet ;
- la méthode *COCOMO II*, définie par *Dr. Barry Boehm* au sein du laboratoire *Center for Software Engineering (CSE)* de l'Université de Californie du Sud (*University of Southern California-USC*), pour l'estimation des coûts et des charges.

La mise en place de ce modèle au sein d'une organisation particulière prend en compte l'aspect humain et culturel de l'entreprise pour pallier le problème d'acceptabilité, qui peut se manifester lors de la conduite du changement. Ainsi, nous avons tenu à adopter une stratégie de persuasion à travers des présentations de la méthodes et de ses avantages.

Le processus d'estimation retenu comporte 3 étapes :

- 1) l'estimation de la taille du projets en Points de Fonction ou en nombre de lignes de code ;
- 2) le choix du modèle d'estimation :
 - le modèle précoce (The Early Design Model) qui est un modèle de haut niveau utilisé tôt dans le cycle de développement ;
 - le modèle post-architecture (The post-architecture Model) qui est un modèle permettant une estimation plus précise des attributs du projet.
- 3) le calibrage des facteurs du modèle sur la base des coûts et des charges réels des projets déjà achevés.

Mise en place d'une méthodologie d'audit projet, un référentiel documentaire et un référentiel des compétences

La mise en place du référentiel qualité est accompagnée par l'implémentation d'une méthodologie d'audit projet, un référentiel documentaire et un référentiel des compétences afin d'assurer le respect du référentiel qualité au sein des projets :

- la définition d'une méthodologie d'audit projet : elle assure la mise en place des bonnes pratiques, mentionnées dans le référentiel qualité, au niveau des projets. L'audit projet permet d'évaluer les écarts du projet en cours par rapport aux bonnes pratiques définies dans le référentiel qualité et de remédier à ces écarts, le cas échéant ;
- la mise en place d'un référentiel documentaire : le référentiel projet a pour but de créer une structure type de répertoire des projets. Cette structure contient tous les modèles de documents utilisables tout le long du projet. Ce répertoire type est dupliqué pour chaque projet. Les objectifs principaux du "référentiel documentaire" sont accélérer la phase de production des documents et préparer et faciliter la localisation des preuves

- d'évaluation et de certification des normes qualité prises en compte dans le référentiel qualité ;
- la définition d'un référentiel des compétences : le référentiel de compétences répond à un besoin de cohérence sur les qualificatifs, la définition des postes et le rôle de chaque intervenant au niveau d'un projet. Il permet :
 - d'évaluer objectivement les collaborateurs et leur évolution ;
 - de faire correspondre les compétences des collaborateurs avec leurs rôles sur les projets.

12.3 Limitations et perspectives

Le référentiel qualité multi-vues

L'approche retenue peut être étendue sur trois plans :

- la mise en place d'un modèle de support d'amélioration continue IDEALSM (relatif à l'implémentation de CMMI). L'objectif est d'améliorer les processus et procédures déjà mis en place à travers des revues de direction et le retour des collaborateurs. Cette boucle d'amélioration continue se base sur les indicateurs mis en place et le retour d'expérience ;
- l'étude de la possibilité d'intégrer un troisième standard ou norme qualité qui prend en considération un champ non couvert par les deux standards retenus. Étant donné qu'ISO 9001v2000 est une norme qualité généraliste et qui traite la mise en place d'un référentiel qualité et la satisfaction du client et que CMMI fournit les bonnes pratiques pour le développement informatique, il serait intéressant d'intégrer un standard qui couvre la qualité de service comme ITIL.
- l'étude de la transformation des processus métier modélisés dans le référentiel qualité en un système d'information pour la gouvernance de l'entreprise. Cette approche est parfaitement compatible avec BPM et consisterait à faire la transformation du formalisme simplifié que nous avons utilisé pour la modélisation du référentiel qualité vers un langage graphique tel que BPMN ou UML. L'étape suivante consisterait, à partir de cette représentation graphique, à générer une représentation exécutable utilisant un langage tel que BPEL ou XPD.

La méthodologie d'estimation des délais et durée des projets

La méthode retenue peut être prolongée sur deux plans :

- nous avons essayé lors du choix des facteurs d'échelle et de coût de prendre des facteurs quantifiables, donc plus facile à évaluer. Mais il y a des facteurs indispensables à l'évaluation mais qui sont non quantifiables et qui dépendent de l'appréciation de l'évaluateur (comme la maturité des programmeurs et des développeurs). Une perspective intéressante de recherche consisterait à essayer de quantifier ces facteurs ou bien les remplacer par un ensemble de facteurs quantifiables et représentatifs du facteur à remplacer. Il faut aussi étudier l'impact de ces facteurs sur l'exactitude des résultats du modèle ;
- une seconde piste à explorer concerne le calibrage et la mise en place du modèle dans une organisation particulière. Cette partie du travail comprend :
 - le calibrage du modèle sur la base des projets existants ;
 - consolider ou éliminer les paramètres redondants ;
 - ajouter de nouveaux facteurs de coûts qui ne sont pas explicités dans le modèle.

Dans notre travail, nous avons considéré seulement la première approche vu le nombre de projets disponibles. Il serait intéressant d'étudier les deux dernières approches et de les appliquer sur le modèle. Enfin, il semble intéressant de prendre en compte les projets finis. Dans notre travail, nous avons choisi la régression linéaire mais il serait intéressant de mener une étude sur d'autres méthodes de calibrage envisageables telles que les réseaux de neurones.

ANNEXES

ANNEXE A

Explication des Process Areas de CMMI

Cet annexe permet d'expliquer succinctement les Process Areas de CMMI et leurs objectifs à travers les tableaux A.1 et A.2.

ANNEXE A. EXPLICATION DES PROCESS AREAS DE CMMI

Niveau de maturité	Sigle	PA	Objectifs
ML 2	REQM	Gestion des exigences	Contrôler les exigences des produits du projet et de ses composants et identifier les contradictions entre ces conditions et les plans du projet et produits du travail.
	PP	Planification de projet	Établir et maintenir les plans qui définissent les activités du projet.
	PMC	Maîtrise et suivi de projet	Fournir une analyse du progrès du projet de sorte que des modalités de reprise appropriées puissent être prises quand l'exécution du projet dévie de manière significative du plan.
	SAM	Gestion des ententes avec les fournisseurs	Contrôler l'acquisition des produits des fournisseurs avec lesquels il existe un accord formel.
	CM	Gestion de la configuration	Établir et maintenir l'intégrité des produits de travail avec l'identification et la commande de configuration, la comptabilité de statut de configuration et les audits de configuration.
	PPQA	Assurance qualité produit et processus	Fournir au personnel et aux managers une vision objective des processus et produits de travail associés.
	MA	Analyses et mesures	Développer des possibilités de mesure employées pour soutenir les besoins de gestion de l'information.
	RD	Développement des exigences	Analyser et produire les exigences client, et celles induites sur le produit et ses composants.
	TS	Solution technique	Concevoir et développer la solution.
	PI	Intégration de produit	Intégrer les composants produits, tester et livrer la solution.
ML 3	VER	Vérification	Vérifier que les produits élémentaires sélectionnés sont conformes à leurs spécifications.
	VAL	Validation	Démontrer que le produit ou composant, placé dans son environnement cible, répond aux exigences.
	OPF	Focalisation de l'organisation sur les processus	Définir et mettre en œuvre les améliorations du processus de l'organisation à partir de l'analyse des forces et des faiblesses des processus de l'organisation et de leur capitalisation.
	OT	Formation dans l'organisation	Développer les compétences des personnes pour leur permettre d'exercer leur mission avec efficacité.
	OPD	Définition des processus de l'organisation	Définir et maintenir le référentiel de l'organisation.
	ISM	Gestion de fournirier intégrée	Identifier proactivement des sources des produits qui peuvent être employés pour répondre aux exigences du projet et pour contrôler les fournisseurs choisis tout en maintenant un rapport coopératif de projet-fournisseur.
	IPM	Gestion de projet intégré	Définir et gérer le projet et les engagements de ses acteurs selon un processus (PDP) dérivé des processus de l'organisation.
	IT	Équipe intégrée	Former et soutenir une équipe intégrée pour le développement des produits de travail.
	RSKM	Gestion des risques	Identifier les problèmes potentiels avant qu'ils ne surviennent et prendre les actions préventives appropriées.
	DAR	Résolution et analyse pour décision	Étudier, selon des critères définis, plusieurs solutions en vue d'une prise de décision.
OEI	Environnement organisationnel en vue de l'intégration	Fournir une infrastructure du développement de produit et de processus intégré (IPPD) et de contrôle des personnes pour l'intégration.	

Tableau A.1 – Explication des Process Areas de CMMI (ML 2 et 3)

Niveau de maturité	Sigle	PA	Objectifs
ML 4	OPPD	Performances des Processus de l'organisation	Mesurer les performances des processus de l'organisation et les analyser par rapports aux objectifs.
	QPM	Gestion Quantitative de la Qualité et des processus	Gérer quantitativement le Project's Defined Process de façon à atteindre les objectifs définis de qualité et les objectifs de performance.
ML 5	OID	Déploiement et amélioration des processus innovation de la technologie des processus de l'organisation	Sélectionner et mettre en œuvre les améliorations et innovations qui impactent de façon mesurable sur les processus et technologies et permettent à l'organisation d'atteindre ses objectifs de qualité et de performance.
	CAR	Résolution et analyse causale	Identifier et supprimer les causes des défauts et problèmes pour éviter leur réapparition dans le futur.

Tableau A.2 – Explication des Process Areas de CMMI (ML 4 et 5)

SCAMPI (Standard CMMI Appraisal Method for Process Improvement) class A

Les figures B.1 et B.2 montrent en détail les étapes de l'évaluation SCAMPI A.

B.1 Planification et préparation de l'évaluation

La première phase du processus d'évaluation SCAMPI consiste à préparer les activités sur site. La figure B.3 montre l'organigramme de l'équipe d'évaluation.

Le plus tôt possible, le Appraisal Team Leader collabore avec le coordinateur interne pour sélectionner les membres de l'équipe d'évaluation (4 à 8 personnes selon le périmètre) qui devront impérativement être préalablement formés :

- Formation d'introduction à CMMI;
- Formation des équipes d'évaluation SCAMPI.

Après ces formations, l'équipe d'évaluation peut prendre en charge le développement opérationnel des PIID (Practice Implementation Indicator Descriptions) et participer à la revue de préparation et à la période d'évaluation sur site.

Les membres de l'équipe d'évaluation devront également signer un engagement de confidentialité stipulant que toutes les informations collectées (sur les projets et auprès des intervenants) demeureront confidentielles au processus d'évaluation. Ce contrat favorise des discussions ou-

ANNEXE B. SCAMPI (STANDARD CMMI APPRAISAL METHOD FOR PROCESS IMPROVEMENT) CLASS A

Phase :	Processus :	Objectif :	Activités :
1 Planifier et préparer l'évaluation	1.1 Analyser les exigences	Comprendre les besoins business de l'organizational unit pour laquelle l'évaluation est demandée. Le appraisal team leader rassemblera l'information et aidera le sponsor de l'évaluation à faire correspondre les objectifs d'évaluation avec les objectifs business.	<p>1.1.1 Déterminer les objectifs</p> <p>1.1.2 Déterminer les contraintes d'évaluation</p> <p>1.1.3 Déterminer le périmètre d'évaluation</p> <p>1.1.4 Déterminer les éléments de sorties</p> <p>1.1.5 Obtenir l'engagement d'obtention des éléments d'entrée</p>
	1.2 Développer le plan d'évaluation	Documenter les résultats de la planification d'évaluation comprenant les exigences, les accords, les estimations, les risques, la mise en oeuvre de la méthode et des considérations pratiques (par exemple, calendrier, logistique, et informations contextuelles sur l'organisation) liées à l'évaluation. Obtenir et enregistrer l'approbation du sponsor sur le plan d'évaluation.	<p>1.2.1 Mettre sur pied la méthode</p> <p>1.2.2 Identifier les ressources nécessaires</p> <p>1.2.3 Déterminer le coût et le plan du programme</p> <p>1.2.4 Planifier et gérer la logistique</p> <p>1.2.5 Documenter et gérer les risques</p> <p>1.2.6 Obtenir l'engagement au plan d'évaluation</p>
	1.3 Sélectionner et préparer l'équipe	S'assurer de la disponibilité d'une équipe expérimentée, exercée, convenablement qualifiée et préparée pour mettre en oeuvre le processus d'évaluation.	<p>1.3.1 Identifier le Team Leader</p> <p>1.3.2 Sélectionner les membres de l'équipe</p> <p>1.3.3 Préparer l'équipe</p>
	1.4 Obtenir et analyser les preuves objectives initiales	Obtenir l'information qui facilite la préparation spécifique au site et une compréhension commune de la mise en oeuvre des pratiques modèles à travers l'organizational unit. Identifiez les problèmes, lacunes, ou risques potentiels pour aider à affiner le plan. Renforcer la compréhension des opérations et des processus de l'organisation.	<p>1.4.1 Préparer les participants</p> <p>1.4.2 Administrer les instruments</p> <p>1.4.3 Obtenir les preuves objectives initiales</p> <p>1.4.4 Répéter les preuves objectives</p>
	1.5 Préparation de la collecte des preuves objectives	Planifier et documenter la stratégie de collecte des données spécifiques incluant : <ul style="list-style-type: none"> • les sources d'information • les outils et techniques utilisés • l'éventualité de devoir faire face à des données insuffisantes 	<p>1.5.1 Réaliser une revue rapide</p> <p>1.5.2 Préparer la planification de la collecte de donnée</p> <p>1.5.3 Replanifier la collecte de donnée</p>

Figure B.1 – Les phases d'évaluation SCAMPI A

B.1. PLANIFICATION ET PRÉPARATION DE L'ÉVALUATION

Phase :	Processus :	Objetif :	Activités :
2 Conduite de l'évaluation	2.1 Examiner les preuves objectives	Collecter les informations concernant les pratiques mises en oeuvre dans l'organisation et relier les données résultantes au modèle référent. Réaliser les activités en accord avec le plan de collecte de donnée. Prendre des actions correctives et revoir le plan de collecte de donnée si nécessaire.	<p>2.1.1 Examiner les preuves objectives provenant des Instruments</p> <p>2.1.2 Examiner les preuves objectives provenant des Présentations</p> <p>2.1.3 Examiner les preuves objectives provenant des Documents</p> <p>2.1.4 Examiner les preuves objectives provenant des Interviews</p>
	2.2 Vérifier et valider les preuves objectives	Vérifier pour chaque instantiation la mise en oeuvre des pratiques au sein de l'organisation, et valider les résultats préliminaires, décrivant les écarts avec celle des pratiques du référentiel entreprise. Chaque mise en oeuvre de chaque pratique est vérifiée de sorte que l'on puisse comparer aux pratiques du modèle CMMI et que l'équipe caractérise l'étendue de la mise en oeuvre des pratiques au sein du référentiel. Les écarts dans la mise en oeuvre des pratiques sont enregistrés et validés avec les membres de l'organisation. Les mises en oeuvre exemplaires des pratiques au sein de l'entreprise peuvent être mises en évidence en tant que forces afin d'être incluses dans le rapport d'évaluation	<p>2.2.1 Vérifier les preuves objectives</p> <p>2.2.2 Caractériser la mise en oeuvre des pratiques au sein du référentiel entreprise</p> <p>2.2.3 Valider les écarts de la mise en oeuvre des pratiques</p>
3 Rapports des résultats	2.3 Documenter les preuves objectives	Rédiger des rapports sur l'information recueillie en identifiant et consolidant ensuite des notes, transformant les données en rapports qui documentent les forces et faiblesses de la mise en oeuvre des pratiques.	<p>2.3.1 Prendre/passser en revue les notes sur preuves objectives</p> <p>2.3.2 Enregistrer la présence/absence de preuves objectives</p>
	2.4 Générer les résultats de l'évaluation	Évaluer le niveau de satisfaction des <i>Goals</i> à partir de niveau de mise en oeuvre des pratiques au sein de l' <i>organizational unit</i> . Le niveau de mise en oeuvre des pratiques est noté à partir de la validité des données (artefact direct, indirect et affirmation) collectées à partir de l'échantillon représentatif de l' <i>organizational unit</i> . Le <i>rating des Capability Level</i> et/ou <i>Maturity Level</i> dépend du niveau de satisfaction des <i>Goals</i> .	<p>2.4.1 Déterminer les résultats et note des <i>Goals</i></p> <p>2.4.2a Déterminer les <i>Capability Level</i> des <i>Process Areas (continue)</i></p> <p>2.4.3a Déterminer le <i>Capability Profile (continue)</i></p> <p>2.4.2b Déterminer le niveau de satisfaction des <i>Process Areas (étagée)</i></p>
	3.1 Livrer les conclusions de l'évaluation	Fournir des résultats crédibles de l'évaluation qui peuvent être utilisés pour guider des actions. Représenter force et faiblesses des processus dans leur utilisation actuelle. Fournir le <i>Capability Level</i> ou <i>Maturity Level</i> .	<p>3.1.1 Présenter les résultats finaux</p> <p>3.1.2 Revoir avec le sponsor les conclusions</p> <p>3.1.3 Planifier les prochaines étapes</p>
	3.2 Synthétiser et archiver les résultats de l'évaluation	Préserver les données importantes et rapports d'évaluation, et s'assurer de la confidentialité des données (pas de projet ni d'individus identifiables).	<p>3.2.1 Collecter les leçons apprises</p> <p>3.2.2 Générer le rapport final d'évaluation</p> <p>3.2.3 Fournir un rapport de documentation de l'évaluation au SEI</p>

Figure B.2 – Les phases d'évaluation SCAMPI A (suite)

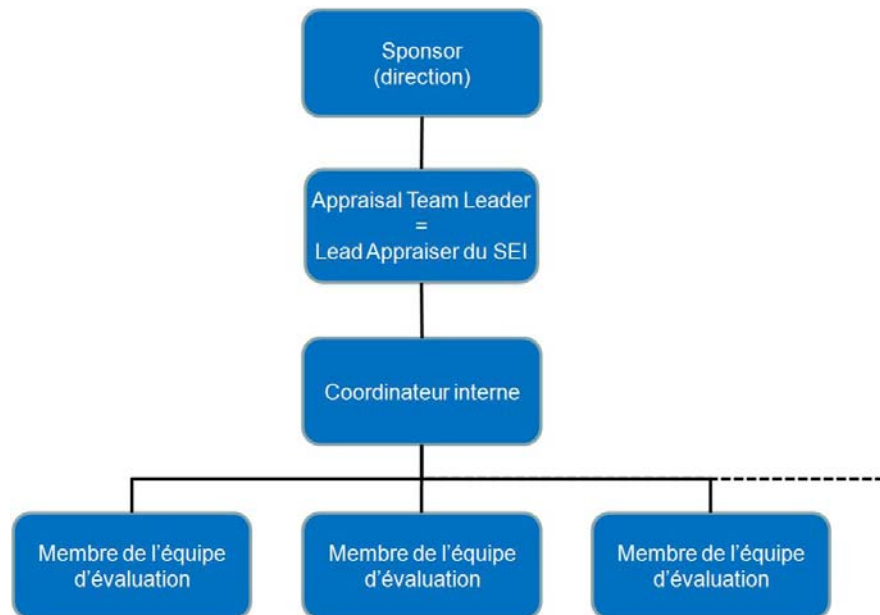


Figure B.3 – Organigramme de l'évaluation SCAMPI Class A

vertes avec les interviewés et entre membres de l'équipe. Avec l'assistance du coordinateur interne, l'évaluateur développe également le plan d'évaluation qui définit les points suivants :

- Périmètre finalisé d'évaluation (organisation, projets couverts, nombre de jours) ;
- Projets concernés (sélectionnés par l'évaluateur) ;
- Fonctions et intervenants à interviewer ;
- Membres de l'équipe d'évaluation ;
- Lieu(x) de réalisation de l'évaluation ;
- Planning détaillé des activités sur site (y compris des interviews).

Les indicateurs d'implémentation de pratiques représentent les preuves objectives qu'une activité (ou pratique) a été implémentée par les projets sélectionnés ; ces indicateurs fournissent les bases de vérification lors de l'évaluation sur site et pour le consensus d'équipe lors de l'évaluation de maturité. Le Directeur d'évaluation collabore avec le coordinateur interne pour sélectionner quatre projets à analyser.

Pour optimiser l'évaluation sur site, l'équipe doit préparer les descriptions des indicateurs d'implémentation PIID pour les projets sélectionnés. Les PIID constituent une structure d'identification des signes objectifs à considérer lors de l'évaluation.

La dernière étape de la phase de préparation réalisée par le Appraisal Team Leader et l'équipe d'évaluation est la revue de préparation et des PIID dont l'objectif premier est de déterminer si la mission d'évaluation sur site peut être réalisée. Cette revue ne vise pas à garantir

que l'entreprise obtiendra la notation désirée, mais à vérifier que les éléments clés sont en place pour conduire la méthode SCAMPI. A l'issue de la revue de préparation, l'évaluateur et l'équipe décident de réaliser l'évaluation sur site ou de la reporter.

B.2 Conduite de l'évaluation

L'évaluation sur site débute par une présentation inaugurale du Appraisal Team Leader présentant les objectifs, le calendrier, les besoins et les processus à tous les interviewés et aux principaux dirigeants.

Après cette phase, l'évaluation se poursuit par l'examen des preuves, la validation et la vérification (avec les PIID pour trame d'analyse). L'équipe d'évaluation conduit les interviews et la revue de documentation par échantillonnage des participants et artefacts des projets sélectionnés. L'équipe d'évaluation enregistre ses observations pour en tirer des conclusions au regard des pratiques CMMI.

Les " découvertes " identifient les forces et les domaines d'amélioration de la conformité des processus d'entreprise avec les pratiques CMMI. L'équipe d'évaluation analyse l'ensemble brut des découvertes avec les interviewés pour les mettre à jour avec d'éventuelles informations additionnelles.

En fonction du jeu résultant d'observations et de découvertes, l'équipe d'évaluation note les objectifs, domaines de processus et le niveau de maturité des processus de l'entreprise et rassemble ses conclusions dans une note finale d'évaluation donnant lieu à une présentation.

B.3 Le rapport des résultats

Le Appraisal Team Leader présente les conclusions finales au Sponsor (direction générale), interviewés et autres intervenants concernés le dernier jour de l'intervention sur site. L'entreprise reçoit alors sa notation de maturité ainsi que le détail de ses forces et domaines d'amélioration.

Le Appraisal Team Leader archive le rapport final d'évaluation auprès du SEI - comme requis par la procédure SCAMPI. Le SEI utilise les données collectées uniquement afin d'identifier et de rapporter des tendances générales - toutes les informations d'identification restant strictement confidentielles.

Le Appraisal Team Leader adresse également un rapport final à l'entreprise reprenant les points clés de la mission (plans, présentations, conclusions, etc.). Toutes les informations fournies par les interviews et revues de projets sont traitées confidentiellement ; le rapport

synthétise les résultats afin qu'aucun projet ni individu ne puissent être identifiés comme source des données.

B.4 Qualification des Appraisal Team Members (ATM)

L'équipe d'évaluation doit :

- Avoir un total de 25 ans d'expérience du terrain avec une moyenne de 6 ans ;
- Avoir un total de 10 ans d'expérience de gestion de projet et au moins un ATM avec au moins 6 ans d'expérience en tant que directeur ou chef de projet ;
- Avoir de l'expérience dans les cycles de vie utilisé par l'Organizational Unit (au moins deux ATMs devraient avoir de l'expérience en tant que opérationnel).

Les ATMs devraient :

- ne pas être directeurs ou chef de l'un des projets choisis ;
- ne pas être le supérieur direct des interviewés ;
- avoir les bonnes qualifications écrites et orales de communication ;
- avoir la capacité de faciliter le flux de communication ;
- avoir participé (au moins 50% d'entre eux) à une évaluation précédente ;
- être perçu par le sponsor d'évaluation comme crédible.

Le résultat du mapping ISO 9001v2000 - CMMI

Cet annexe permet de présenter le résultat du mapping entre les chapitres d'ISO 9001v2000 et les pratiques de CMMI.

C.1 Mapping ISO 9001v2000 vers CMMI

ISO 9001v2000	CMMI PA ¹	CMMI pratiques	Force ²	Commentaires
4.1 - Système de management de la qualité : Exigences générales	Tout	GP 2.1, 2.2, 2.3, 2.6, 2.8, 2.9	S	
	OPD	SP 1.1	S	
	OPF	SP 2.2	S	
	SAM	SP 1.3	M	CMMI est plus faible
	SAM	SP 2.2.1	S	
4.2.1 - Exigences relatives à la documentation : Généralités	Tout	GP 2.1	M	CMMI satisfait généralement les clauses
				<i>Suite page suivante ...</i>

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA¹	CMMI pratiques	Force²	Commentaires
4.2.1 - Exigences relatives à la documentation : Généralités	OPD	SP 1.1	M	CMMI satisfait généralement les clauses
4.2.2 - Manuel qualité	Tout	GP 2.2	S	
	OPD	SP 1.1	S	
4.2.3 - Maîtrise des documents	Tout	GP 2.6	S	
	CM	Tout	S	
4.2.4 - Maîtrise des enregistrements	Tout	GP 2.6	S	
	CM	Tout	S	
5.1 - Engagement de la direction	Tout	GP 2.1, 2.3, 2.10	M	CMMI est plus faible
5.2 - Écoute client	Tout	GP 2.7	M	CMMI ne recherche pas explicitement la satisfaction du client
	RD	SP 1.1, 1.2, 2.1, 3.3, 3.4, 3.5	M	CMMI ne recherche pas explicitement la satisfaction du client
5.3 - Politique Qualité	Tout	GP 2.1	S	CMMI est plus spécifique sur le contenu de la politique
	OPF	SP 1.1	S	CMMI est plus spécifique sur le contenu de la politique
5.4.1 - Objectifs qualité	OPF	SP 1.1	S	
	OPP	SP 1.3	S	
	QPM	SP 1.1, 1.2, 1.3	S	
5.4.2 - Planification du système de management de la qualité	OPD	Tout	S	
	Tout	GP 2.2, 3.1	S	
<i>Suite page suivante ...</i>				

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA¹	CMMI pratiques	Force²	Commentaires
5.5.1 - Responsabilités et autorités	Tout	GP 2.4	S	
5.5.2 - Représentant de la direction	OPF	GP 2.4	W	Concept non explicite dans CMMI
5.5.3 - Communication interne	OPD	GP 2.1	W	CMMI est plus faible. Pas d'exigences pour communiquer efficacement
	OPF	GP 2.10, SP 1.1	W	CMMI est plus faible. Pas d'exigences pour communiquer efficacement
5.6.1 - Revue de direction : Généralités	Tout	GP 2.6, 2.10	S	
	OPF	SP 1.2, 1.3	S	
5.6.2 - Éléments d'entrée de la revue	Tout	GP 2.10	S	
	PMC	SP 1.6, 1.7, 2.1, 2.2, 2.3	S	
5.6.3 - Éléments de sortie de la revue	Tout	GP 2.10	S	
	PMC	SP 1.6, 1.7, 2.1, 2.2, 2.3	S	
6.1 - Mise à disposition des ressources	Tout	GP 2.3	S	
	PP	SP 2.4	S	
6.2.1 - Ressources humaines : Généralités	Tout	GP 2.5	S	
6.2.2 - Compétences, sensibilisation et formation	Tout	GP 2.5	S	
	OT	Tout	S	
	OEI	SP 1.3	S	
<i>Suite page suivante ...</i>				

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA ¹	CMMI pratiques	Force ²	Commentaires
6.3 - Infrastructure	Tout	GP 2.3	S	
	OEI	SP 1.2	S	
6.4 - Environnement de travail	Tout	GP 2.3	S	Pas explicitement couvert par CMMI
	PP	SP 2.4	M	Pas explicitement couvert par CMMI
	OEI	SP 1.2	M	Pas explicitement couvert par CMMI
7.1 - Planification de la réalisation du produit	Tout	GP 2.2, 3.1	S	
	IPM	SP 1.1, 1.3, 1.4	S	
	OPD	SP 1.1, 1.2, 1.3	S	
	PP	Tout	S	
	QPM	SP 1.1, 1.2	S	
7.2.1 - Détermination des exigences relatives au produit	RD	SP 1.1, 1.2, 2.1, 2.2, 2.3, 3.1, 3.2	S	
	REQM	SP 1.1	S	
	TS	SP 1.2	S	
7.2.2 - Revue des exigences relatives au produit	REQM	GP 2.6, 2.7. SP 1.1, 1.2, 1.3, 1.5	S	ISO est moins spécifique que CMMI
	RD	GP 2.7, 2.10. SP 1.1, 2.1, 3.5	S	ISO est moins spécifique que CMMI
	VER	GP 2.7. SP 2.2	S	
7.2.3 - Communication avec le client	IPM	SP 2.1, 2.2, 2.3	M	CMMI est plus faible
	MA	SP 2.4	M	CMMI est plus faible
<i>Suite page suivante ...</i>				

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA¹	CMMI pratiques	Force²	Commentaires
7.2.3 - Communication avec le client	REQM	SP 1.2	M	CMMI est plus faible
	RD	GP 2.7	M	CMMI est plus faible
7.3.1 - Planification de la conception et du développement	IPM	SP 1.1, 1.3, 1.4, 2.1, 2.2, 2.3, 3.1, 3.2, 4.1, 4.2, 4.3	S	
	TS	SP 1.1, 1.2, 1.3, 2.1, 2.2, 2.3, 2.4, 3.1, 3.2	S	
	PI	SP 1.1, 1.2, 1.3, 2.1, 2.2	S	
	PP	Tout	S	
	VAL	SP 1.1	S	
	VER	SP 1.1	S	
7.3.2 - Éléments d'entrée de la conception et du développement	RD	GP 2.7, 2.10. SP 1.1, 1.2, 2.1, 3.2, 3.3, 3.4, 3.5	S	
7.3.3 - Éléments de sortie de la conception et du développement	TS	Tout	S	
	IPM	SP 1.1	S	
7.3.4 - Revue de la conception et du développement	PMC	GP 2.7. SP 1.6, 1.7	S	
7.3.5 - Vérification de la conception et du développement	VER	GP 2.6. SP 1.1, 1.2, 1.3, 2.1, 2.2, 2.3, 3.1, 3.2	S	
7.3.6 - Validation de la conception et du développement	VAL	GP 2.6. SP 1.1, 1.2, 1.3, 2.1, 2.2	S	
	PI	GP 2.6	S	
<i>Suite page suivante ...</i>				

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA ¹	CMMI pratiques	Force ²	Commentaires
7.3.7 - Maîtrise des modifications de la conception et du développement	TS et PI	GP 2.6	S	
	CM	Tout	S	
7.4.1 - Processus d'achat	SAM	GP 2.6, 2.9. SP 1.1, 2.3	S	
	SAM	SP 1.2, 2.2	M	CMMI est plus faible
	PI	SP 3.1	S	
7.4.1 - Processus d'achat	TS	SP 1.1, 1.2, 1.3, 2.4	S	
7.4.2 - Informations relatives aux achats	SAM	SP 1.1, 1.3, 2.1	S	
7.4.3 - Vérification du produit acheté	SAM	SP 1.3, 2.2	S	
	SAM	SP 2.3	W	CMMI n'est pas explicite
	VER	SP 3.1	S	
7.5.1 - Maîtrise de la production et de la préparation de service	TS	GP 2.2, 2.3, 2.6, 2.8. SP 3.1, 3.2	M	CMMI est plus faible
	PI	SP 3.1, 3.2, 3.3, 3.4	M	CMMI est plus faible
7.5.2 - Validation des processus de production et de préparation de service	VAL	Tout	S	
	RD	SP 3.5	S	
7.5.3 - Identification et traçabilité	CM	SP 1.1, 1.3, 2.1, 2.2, 3.1	S	
	PI	SP 3.1, 3.2, 3.3	S	
	REQM	SP 1.4	S	
<i>Suite page suivante ...</i>				

C.1. MAPPING ISO 9001V2000 VERS CMMI

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA¹	CMMI pratiques	Force²	Commentaires
7.5.4 - Propriété du client				Pas explicitement couvert par CMMI
7.5.5 - Préservation du produit	PI	SP 3.4	M	CMMI est moins explicite
7.6 - Maîtrise des dispositifs de surveillance et de mesure	MA	GP 2.1, 2.2, 2.8, 2.9, 2.10	S	
	VAL	GP 2.8	S	
	VER	GP 2.8	S	
8.1 - Mesures, analyses et améliorations : Généralités	Tout	GP 2.2, 2.8	S	
	MA	SP 1.1, 1.2, 1.3, 1.4	S	
	QPM	SP 2.1, 2.2, 2.3, 2.4	S	
8.2.1 - Satisfaction du client	MA	SP 1.1, 1.2, 2.2	M	CMMI pas très fort
	PMC	SP 1.5	M	CMMI pas très fort
8.2.2 - Audit interne	MA	SP 2.4	S	Plan d'amélioration des processus de CMMI est plus explicite
	PPQA	Tout	S	CMMI est plus détaillé
	OPF	Tout	S	CMMI est plus détaillé
8.2.3 - Surveillance et mesure des processus	Tout	GP 2.8	S	
	MA	GP 2.2, SP 1.2, 1.3, 1.4	S	
	PMC	SP 2.1, 2.2, 2.3	S	
	QPM	SP 2.2, 2.3	S	
<i>Suite page suivante ...</i>				

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA¹	CMMI pratiques	Force²	Commentaires
8.2.4 - Surveillance et mesure du produit	VER	SP 1.1, 1.2, 2.1, 2.2, 2.3, 3.1, 3.2	S	
	CM	SP 3.2	M	CMMI est plus faible
	VAL	SP 1.3, 2.1, 2.2	S	
	REQM	SP 1.1	S	
	SAM	SP 1.3	S	
	PPQA	SP 1.2	S	
8.3 - Maîtrise du produit non conforme	CM	Tout	M	Pas explicitement exigé par CMMI
	PMC	SP 2.1, 2.2, 2.3	M	Pas explicitement exigé par CMMI
	VAL	GP 2.4, SP 1.3, 2.2	M	Pas explicitement exigé par CMMI
	VER	GP 2.4, SP 1.3, 2.2	M	Pas explicitement exigé par CMMI
8.4 - Analyse des données	Tout	GP 2.8, 3.2	S	
	MA	SP 2.3, 2.3, 2.4	S	
	OPF	SP 1.3	S	
	OPD	SP 1.1, 1.2, 1.3, 1.4, 1.5	S	
	RD	SP 1.1, 1.2, 2.1, 3.1, 3.2, 3.3, 3.4	M	CMMI ne recherche pas explicitement la satisfaction du client
	QPM	SP 1.4	M	CMMI ne recherche pas explicitement la satisfaction du client
	CAR	SP 1.1, 1.2	M	CMMI ne recherche pas explicitement la satisfaction du client
<i>Suite page suivante ...</i>				

<i>... Suite de la page précédente</i>				
ISO 9001v2000	CMMI PA ¹	CMMI pratiques	Force ²	Commentaires
8.4 - Analyse des données	SAM	SP 2.2	M	CMMI ne recherche pas explicitement la satisfaction du client
8.5.1 - Amélioration continue	MA	SP 1.1, 1.2, 1.4, 2.1, 2.2	S	
	OPF	SP 1.1, 1.3	S	
	OID	SP 1.1	S	
8.5.2 - Action corrective	PMC	SP 2.1, 2.2, 2.3	S	
	OPF	SP 2.1, 2.2, 2.3	S	
	CAR	Tout	S	
8.5.3 - Action préventive	CAR	SP 1.1, 1.2, 2.1, 2.2, 2.3	M	CMMI est plus faible, mais plus sophistiqué (CAR Processus de niveau 5)
	OPF	SP 2.4	S	

Tableau C.1 – Mapping ISO 9001v2000 vers CMMI

¹Process Area = Domaine de processus.

²Force = Niveau de couverture de la pratique CMMI de la recommandation ISO 9001v2000 (S :Fort, M :Moyen, W :Faible).

C.2 Mapping CMMI vers ISO 9001v2000

Le mapping des pratiques CMMI vers les chapitres ISO 9001v2000 a été réalisé en adéquation avec les résultats présentés dans le tableau C.1. Le tableau C.2 présente le mapping des pratiques de CMMI vers les chapitres d'ISO 9001v2000.

CMMI	Description	ISO 9001v2000	Force ¹
CM - Gestion de configuration			
GP 2.1	Établir une directive organisationnelle	4.1, 4.2.3, 5.3, 7.3.7	S
		4.2.1, 5.1, 8.3	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 4.2.3, 4.2.4, 5.4.2, 7.1, 7.3.7, 8.1	S
		8.3	M
GP 2.3	Fournir les ressources	4.1, 4.2.3, 6.1, 6.3, 7.3.7	S
		5.1, 6.4, 8.3	M
GP 2.4	Assigner les responsabilités	4.2.3, 5.5.1, 7.3.7	S
		8.3	M
GP 2.5	Former les personnes	4.2.3, 6.2.1, 6.2.2, 7.3.7	S
		8.3	M
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.3.7	S
		8.3	M
GP 2.7	Identifier et impliquer les parties prenantes concernées	4.2.3, 7.3.7	S
		5.2, 8.3	M
GP 2.8	Suivre et contrôler le processus	4.1, 4.2.3, 7.3.7, 8.1, 8.2.3, 8.4	S
		8.3	M
GP 2.9	Évaluer la conformité de manière objective	4.1, 4.2.3, 7.3.7	S
		8.3	M
<i>Suite page suivante . . .</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	4.2.3, 5.6.1, 5.6.2, 5.6.3, 7.3.7	S
		5.1, 8.3	M
GP 3.1	Établir un processus personnalisé	4.1, 4.2.3, 5.4.2, 7.1, 7.3.7	S
		8.3	M
GP 3.2	Collecter des informations sur l'amélioration	4.1, 4.2.3, 7.3.7, 8.4	S
		8.3	M
SP 1.1	Identifier les articles de configuration	4.2.3, 4.2.4, 7.3.7, 7.5.3	S
SP 1.1	Identifier les articles de configuration	8.3	M
SP 1.2	Établir un système de gestion de configuration	4.2.3, 4.2.4, 7.3.7	S
		8.3	M
SP 1.3	Créer ou mettre à jour des référentiels	4.2.3, 4.2.4, 7.3.7, 7.5.3	S
		8.3	M
SP 2.1	Suivre les demandes de changement	4.2.3, 4.2.4, 7.3.7, 7.5.3	S
		8.3	M
SP 2.2	Contrôler les articles de configuration	4.2.3, 4.2.4, 7.3.7, 7.5.3	S
		8.3	M
SP 3.1	Établir des enregistrements de gestion de configuration	4.2.3, 4.2.4, 7.3.7, 7.5.3	S
		8.3	M
SP 3.2	Mener des audits de configuration	4.2.3, 4.2.4, 7.3.7	S
		8.2.4, 8.3	M
MA - Analyses et mesures			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 7.6	S
		4.2.1, 5.1	M
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 7.6, 8.1, 8.3.2	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 7.6, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 7.6	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3, 7.6	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Établir des objectifs de mesure	8.1, 8.5.1	S
		8.2.1	M
SP 1.2	Spécifier des mesures	8.1, 8.2.3, 8.5.1	S
		8.2.1	M
SP 1.3	Spécifier des procédures de collecte et de stockage de données	8.1, 8.2.3	S
SP 1.4	Spécifier des procédures d'analyse	8.1, 8.2.3, 8.5.1	S
SP 2.1	Collecter des données de mesure	8.2.4, 8.5.1	S
SP 2.2	Analyser les données de mesure	8.4, 8.5.1	S
		8.2.1	M
SP 2.3	Stocker données et résultats	8.4	S
SP 2.4	Communiquer des résultats	8.2.2	S
		7.2.3, 8.4	M
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
PMC - Maîtrise et suivi de projet			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.3.4	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Suivre les paramètres de planification de projet		
SP 1.2	Suivre les engagements		
SP 1.3	Suivre les risques de projet		
SP 1.4	Suivre la gestion des données		
SP 1.5	Suivre l'implication des parties prenantes	8.2.1	M
SP 1.6	Mener des revues d'avancement	5.6.2, 5.6.3, 7.3.4	S
SP 1.7	Mener des revues de jalons	5.6.2, 5.6.3, 7.3.4	S
SP 2.1	Analyser les points à résoudre	5.6.2, 5.6.3, 8.2.3, 8.5.2	S
		8.3	M
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.2	Appliquer une action corrective	5.6.2, 5.6.3, 8.2.3, 8.5.2	S
		8.3	M
SP 2.3	Gérer une action corrective	5.6.2, 5.6.3, 8.2.3, 8.5.2	S
		8.3	M
PP - Planification de Projet			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 7.3.1	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 7.3.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2, 7.3.1	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.3.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.3.1	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 7.3.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 7.3.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3, 7.3.1	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1, 7.3.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 7.3.1, 8.4	S
SP 1.1	Faire l'estimation de la portée du projet	7.1, 7.3.1	S
SP 1.2	Établir les estimations des attributs des produits de sorties et des tâches	7.1, 7.3.1	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.3	Définir le cycle de vie du projet	7.1, 7.3.1	S
SP 1.4	Déterminer les estimations de charge et de coût	7.1, 7.3.1	S
SP 2.1	Établir le budget et le calendrier	7.1, 7.3.1	S
SP 2.2	Identifier les risques du projet	7.1, 7.3.1	S
SP 2.3	Prévoir la gestion des données	4.2.3, 7.1, 7.3.1	S
SP 2.4	Prévoir les ressources du projet	6.1, 7.1, 7.3.1	S
		6.4	M
SP 2.5	Prévoir les connaissances et aptitudes nécessaires	6.2.2, 7.1, 7.3.1	S
SP 2.6	Planifier l'implication des parties prenantes	7.1, 7.3.1	S
SP 2.7	Établir le plan de projet	7.1, 7.3.1	S
SP 3.1	Passer en revue les plans qui ont une incidence sur le projet	7.3.1	S
SP 3.2	Concilier le niveau de charge et de ressources	7.3.1	S
SP 3.3	Obtenir l'engagement au plan	7.3.1	S
PPQA - Assurance qualité produit et processus			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1, 8.2.2	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1, 8.2.2	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 8.2.2	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 8.2.2	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Évaluer objectivement des processus	8.2.2	S
SP 1.2	Évaluer objectivement des produits des sorties et des services	8.2.2, 8.2.4	S
SP 2.1	Communiquer et assurer la résolution des points non conformes	8.2.2	S
SP 2.2	Établir des enregistrements	4.2.4, 8.2.2	S
REQM - Gestion des exigences			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.2.2	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.2.2	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Obtenir une compréhension des exigences	7.2.1, 7.2.2, 8.2.4	S
SP 1.2	Obtenir l'engagement sur les exigences	7.2.2	S
		7.2.3	M
SP 1.3	Gérer les changements aux exigences	7.2.2	S
SP 1.4	Maintenir la traçabilité biunivoque des exigences	7.5.3	S
SP 1.5	Identifier les incohérences entre les produits du projet et les exigences	7.2.2	S
SAM - Gestion des ententes avec les fournisseurs			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.4.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	4.1	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 7.4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Déterminer un type d'acquisition	7.4.1, 7.4.2	S
SP 1.2	Choisir des fournisseurs	7.4.1	S
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.3	Établir des ententes avec le fournisseur	7.4.2, 7.4.3, 8.2.4	S
		4.1	M
SP 2.1	Passer en revue les produits sur étagère ("COTS")	7.4.2, 7.4.3	S
SP 2.2	Se conformer à l'entente avec le fournisseur	4.1, 7.4.3	S
		7.4.1, 8.4	M
SP 2.3	Accepter le produit acquis	7.4.1	S
		7.4.3	W
SP 2.4	Transférer les produits		
DAR - Résolution et Analyse pour Décision			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Établir des lignes directives pour l'analyse de décision		
SP 1.2	Établir et maintenir des critères d'évaluation		
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.3	Identifier des solutions possibles		
SP 1.4	Sélectionner des méthodes d'évaluation		
SP 1.5	Évaluer des solutions possibles		
SP 1.6	Sélectionner des solutions		
IPM - Gestion de Projet Intégrée			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Établir le processus personnalisé pour le projet	7.1, 7.3.1, 7.3.3	S
SP 1.2	Utiliser les actifs processus organisationnels pour les activités de planification du projet		
SP 1.3	Intégrer les plans	7.1, 7.3.1	S
SP 1.4	Gérer le projet en utilisant les plans intégrés	7.1, 7.3.1	S
SP 1.5	Contribuer aux actifs processus organisationnels		
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.1	Gérer l'implication des parties prenantes	7.3.1	S
		7.2.3	M
SP 2.2	Gérer les dépendances	7.3.1	S
		7.2.3	M
SP 2.3	Régler les problèmes de coordination	7.3.1	S
		7.2.3	M
SP 3.1	Définir le contexte pour la vision commune du projet	7.3.1	S
SP 3.2	Établir une vision commune du projet	7.3.1	S
SP 4.1	Déterminer la structure d'équipe intégrée pour le projet	7.3.1	S
SP 4.2	Développer une distribution préliminaire des exigences aux équipes intégrées	7.3.1	S
SP 4.3	Établir les équipes intégrées	7.3.1	S
ISM - Gestion de fournisseur intégrée			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Analyser les sources potentielles de produits		
SP 1.2	Évaluer et déterminer les sources de produits		
SP 2.1	Suivre des processus sélectionnés chez le fournisseur		
SP 2.2	Évaluer des produits de sortie sélectionnés chez le fournisseur		
SP 2.3	Revoir l'entente ou la relation avec le fournisseur		
IT - Équipe intégrée			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Identifier les tâches de l'équipe		
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.2	Identifier les connaissances et les aptitudes requises		
SP 1.3	Affecter les membres de l'équipe appropriés		
SP 2.1	Établir une vision commune		
SP 2.2	Établir une charte de l'équipe		
SP 2.3	Définir les rôles et les responsabilités		
SP 2.4	Établir les procédures de fonctionnement		
SP 2.5	Collaborer entre les équipes qui travaillent en commun		
OEI - Environnement organisationnel en vue de l'intégration			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Établir la vision partagée de l'organisation		
SP 1.2	Établir un environnement de travail intégré	6.3	S
		6.4	M
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.3	Identifier les exigences d'aptitudes spéciales pour IPPD	6.2.2	S
SP 2.1	Établir des mécanismes de leadership		
SP 2.2	Établir des incitatifs à l'intégration		
SP 2.3	Établir des mécanismes pour équilibrer responsabilités d'équipe et responsabilités d'unité d'origine		
OPD - Définition des Processus de l'Organisation			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 5.4.2	S
		4.2.1, 5.1	M
		5.5.3	W
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 5.4.2, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.4.2, 5.5.1	S
GP 2.5	Former les personnes	5.4.2, 6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.4.2, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.4.2	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 5.4.2, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 5.4.2	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.4.2, 5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 5.4.2, 8.4	S
SP 1.1	Établir les processus normalisés	4.1, 4.2.2, 5.4.2, 7.1	S
		4.2.1	M
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.2	Établir les descriptions des modèles de cycles de vie	5.4.2, 7.1	S
		4.2.1	M
SP 1.3	Établir les critères et les lignes directives de personnalisation	5.4.2, 7.1	S
		4.2.1	M
SP 1.4	Établir le répertoire de mesures de l'organisation	5.4.2	S
SP 1.5	Établir une bibliothèque des actifs processus de l'organisation	5.4.2	S
OPF - Focalisation de l'Organisation sur les Processus			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 8.2.2	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1, 8.2.2	S
GP 2.3	Fournir les ressources	4.1, 5.4.2, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1, 8.2.2	S
		5.2.2	W
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
		5.5.3	W
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.1	Établir les besoins du processus organisationnel	4.1, 5.3, 5.4.1, 8.2.2, 8.5.1	S
		5.5.3	W
SP 1.2	Évaluer les processus de l'organisation	5.6.1, 8.2.2, 8.4	S
SP 1.3	Identifier les améliorations au processus de l'organisation	5.6.1, 8.2.2, 8.4, 8.5.1	S
SP 2.1	Établir des plans d'action processus	8.2.2, 8.5.2	S
SP 2.2	Implémenter les plans d'action processus	4.1, 8.2.2, 8.5.2	S
SP 2.3	Déployer les actifs de processus organisationnels	8.5.2	S
SP 2.4	Incorporer les expériences de processus dans les actifs de processus organisationnels	8.5.3	S
OT - Formation dans l'Organisation			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Établir les besoins stratégiques de formation	6.2.2	S
SP 1.2	Déterminer quels besoins de formation seront du ressort de l'organisation	6.2.2	S
SP 1.3	Établir un plan de formation tactique organisationnel	6.2.2	S
SP 1.4	Établir une capacité de formation	6.2.2	S
SP 2.1	Dispenser la formation	6.2.2	S
SP 2.2	Établir des dossiers de formation	6.2.2	S
SP 2.3	Évaluer l'efficacité de la formation	6.2.2	S
PI - Intégration du Produit			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.3.6, 7.3.7	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Déterminer la séquence d'intégration	7.3.1	S
SP 1.2	Établir l'environnement d'intégration de produit	7.3.1	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.3	Établir les procédures et critères d'intégration de produit	7.3.1	S
SP 2.1	Passer en revue les descriptions d'interfaces pour s'assurer de leur exhaustivité	7.3.1	S
SP 2.2	Gérer les interfaces	7.3.1	S
SP 3.1	Confirmer que les composants de produit sont prêts à être intégré	7.4.1, 7.5.3	S
		7.5.1	M
SP 3.2	Assembler les composants de produit	7.5.3	S
		7.5.1	M
SP 3.3	Évaluer les composants de produits assemblés	7.5.3	S
		7.5.1	M
SP 3.4	Conditionner et livrer le produit ou le composant de produit	7.5.1, 7.5.5	M
RD - Développement Exigences			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.2.2, 7.3.2	S
		5.2, 7.2.3	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3, 7.2.2, 7.3.2	S
		5.1	M
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Expliciter les besoins	7.2.1, 7.2.2, 7.3.2	S
		5.2, 8.4	M
SP 1.2	Développer les exigences client	7.2.1, 7.3.2	S
		5.2, 8.4	M
SP 2.1	Établir les exigences produit et composants de produit	7.2.1, 7.2.2, 7.3.2	S
		5.2, 8.4	M
SP 2.2	Allouer les exigences composants de produit	7.2.1	S
SP 2.3	Identifier les exigences d'interface	7.2.1	S
SP 3.1	Établir des conceptions de fonctionnement et des scénarios	7.2.1	S
		8.4	M
SP 3.2	Établir une définition des fonctionnalités requises	7.2.1, 7.3.2	S
		8.4	M
SP 3.3	Analyser les exigences	7.3.2	S
		5.2, 8.4	M
SP 3.4	Analyser les exigences pour assurer l'équilibre	7.3.2	S
		5.2, 8.4	M
SP 3.5	Valider les exigences avec des méthodes exhaustives	7.2.2, 7.3.2, 7.5.2	S
		5.2	M
RSKM - Gestion des Risques			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Déterminer les sources et les catégories de risques		
SP 1.2	Définir les paramètres de risques		
SP 1.3	Établir la stratégie de gestion du risque		
SP 2.1	Identifier les risques		
SP 2.2	Évaluer les risques, les catégoriser et les ordonner par priorités		
SP 3.1	Développer les plans de réduction du risque		
SP 3.2	Implémenter les plans de réduction du risque		
TS - Solution Technique			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 7.3.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 7.3.3, 8.1	S
		7.5.1	M
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3, 7.3.3	S
		5.1, 6.4, 7.5.1	M
GP 2.4	Assigner les responsabilités	5.5.1, 7.3.3	S
GP 2.5	Former les personnes	6.2.1, 6.2.2, 7.3.3	S
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.3.3, 7.3.7	S
		7.5.1	M
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.3.3	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 7.3.3, 8.1, 8.2.3, 8.4	S
GP 2.8	Suivre et contrôler le processus	7.5.1	M
GP 2.9	Évaluer la conformité de manière objective	4.1, 7.3.3	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3, 7.3.3	S
			M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1, 7.3.3	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 7.3.3, 8.4	S
SP 1.1	Développer des alternatives de solutions détaillées ainsi que des critères de sélection	7.3.1, 7.3.3, 7.4.1	S
SP 1.2	Faire évoluer les conceptions opérationnelles et les scénarios	7.2.1, 7.3.1, 7.3.3, 7.4.1	S
SP 1.3	Choisir les solutions pour les composants de produit	7.3.1, 7.3.3, 7.4.1	S
SP 2.1	Faire la conception du produit ou du composant de produit	7.3.1, 7.3.3	S
SP 2.2	Établir un ensemble de données techniques	7.3.1, 7.3.3	S
SP 2.3	Faire la conception des interfaces en s'appuyant sur des critères	7.3.1, 7.3.3	S
SP 2.4	Réaliser les analyses permettant de déterminer si on va faire, acheter ou réutiliser	7.3.1, 7.3.3, 7.4.1	S
SP 3.1	Implémenter la conception	7.3.1, 7.3.3	S
		7.5.1	M
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 3.2	Développement de la documentation de soutien au produit	7.3.1, 7.3.3	S
		7.5.1	M
VAL - Validation			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 7.5.2	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 7.5.2, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3, 7.5.2	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1, 7.5.2	S
		8.3	M
GP 2.5	Former les personnes	6.2.1, 6.2.2, 7.5.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.3.6, 7.5.2	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.5.2	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 7.5.2, 7.6, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 7.5.2	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3, 7.5.2	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1, 7.5.2	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 7.5.2, 8.4	S
SP 1.1	Choisir les produits en vue de la validation	7.3.1, 7.3.6, 7.5.2	S
SP 1.2	Établir l'environnement de validation	7.3.6, 7.5.2	S
SP 1.3	Établir les procédures et les critères de validation	7.3.6, 7.5.2, 8.2.4	S
		8.3	M
SP 2.1	Réaliser la validation	7.3.6, 7.5.2, 8.2.4	S
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.2	Analyser les résultats de validation	7.3.6, 7.5.2, 8.2.4	S
		8.3	M
VER - Vérification			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
		8.3	M
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 7.3.5	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	7.2.2	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 7.6, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Choisir les produits de sortie à vérifier	7.3.1, 7.3.5, 8.2.4	S
SP 1.2	Établir l'environnement de vérification	7.3.5	S
SP 1.2	Établir les procédures et critères de vérification de produit	7.3.5, 8.2.4	S
		8.3	M
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.1	Se préparer pour les revues de pairs	7.3.5, 8.2.4	S
SP 2.2	Mener des revues de pairs	7.2.2, 7.3.5, 8.2.4	S
SP 2.3	Analyser des données de revues de pairs	7.3.5, 8.2.4	S
SP 3.1	Réaliser la vérification	7.3.5, 7.4.3, 8.2.4	S
SP 3.2	Analyser les résultats de vérification et identifier les actions correctives	7.3.5, 8.2.4	S
		8.3	M
OPP - Performances des Processus de l'Organisation			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Choisir les processus	8.4	S
SP 1.2	Établir les mesures de performance du processus	8.4	S
SP 1.3	Établir les objectifs de qualité et de performance de processus	5.4.1, 8.4	S
<i>Suite page suivante ...</i>			

ANNEXE C. LE RÉSULTAT DU MAPPING ISO 9001V2000 - CMMI

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 1.4	Établir les référentiels de performance de processus	8.4	S
SP 1.5	Établir les modèles de performance des processus	8.4	S
QPM - Gestion Quantitative de la Qualité et des Processus			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3, 7.1	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Établir les objectifs du projet	5.4.1, 7.1	S
SP 1.2	Composer le processus personnalisé	5.4.1, 7.1	S
SP 1.3	Choisir les sous-processus qui sont gérés statistiquement	5.4.1	S
SP 1.4	Gérer la performance du projet	8.4	M
SP 2.1	Choisir les mesures et les techniques d'analyse	8.1	S
SP 2.2	Appliquer des méthodes statistiques pour comprendre la variation	8.1, 8.2.3	S
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.3	Suivre la performance des sous-processus sélectionnés	8.1, 8.2.3	S
SP 2.4	Enregistrer les données de gestion statistique	8.1	S
CAR - Résolution et Analyse Causale			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3, 8.5.2	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1, 8.5.2	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3, 8.5.2	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1, 8.5.2	S
GP 2.5	Former les personnes	6.2.1, 6.2.2, 8.5.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1, 8.5.2	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	8.5.2	S
		5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4, 8.5.2	S
GP 2.9	Évaluer la conformité de manière objective	4.1, 8.5.2	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3, 8.5.2	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1, 8.5.2	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4, 8.5.2	S
SP 1.1	Choisir les données de défauts pour analyse	8.5.2	S
		8.4, 8.5.3	M
SP 1.2	Analyser les causes	8.5.2	S
		8.4, 8.5.3	M
SP 2.1	Implémenter les propositions d'actions	8.5.2	S
		8.5.3	M
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.2	Évaluer les retombées des changements	8.5.2	S
		8.5.3	M
SP 2.3	Enregistrer les données	8.5.2	S
		8.5.3	M
OID - Déploiement et Amélioration des Processus Innovation de la technologie des processus de l'Organisation			
GP 2.1	Établir une directive organisationnelle	4.1, 5.3	S
		4.2.1, 5.1	M
GP 2.2	Planifier le processus	4.1, 4.2.2, 5.4.2, 7.1, 8.1	S
GP 2.3	Fournir les ressources	4.1, 6.1, 6.3	S
		5.1, 6.4	M
GP 2.4	Assigner les responsabilités	5.5.1	S
GP 2.5	Former les personnes	6.2.1, 6.2.2	S
GP 2.6	Gérer en configuration	4.1, 4.2.3, 4.2.4, 5.6.1	S
GP 2.7	Identifier et impliquer les parties prenantes concernées	5.2	M
GP 2.8	Suivre et contrôler le processus	4.1, 8.1, 8.2.3, 8.4	S
GP 2.9	Évaluer la conformité de manière objective	4.1	S
GP 2.10	Passer le statut en revue avec les responsables de niveau supérieur	5.6.1, 5.6.2, 5.6.3	S
		5.1	M
GP 3.1	Établir un processus personnalisé	4.1, 5.4.2, 7.1	S
GP 3.2	Collecter des informations sur l'amélioration	4.1, 8.4	S
SP 1.1	Collecter et analyser les propositions d'amélioration	8.5.1	S
SP 1.2	Identifier et analyser les innovations		
SP 1.3	Faire des projets pilotes des améliorations		
SP 1.4	Sélectionner les améliorations pour déploiement		
SP 2.1	Planifier le déploiement		
<i>Suite page suivante ...</i>			

<i>... Suite de la page précédente</i>			
CMMI	Description	ISO 9001v2000	Force¹
SP 2.2	Gérer le déploiement		
SP 2.3	Mesurer les effets de l'amélioration		

Tableau C.2 – Mapping CMMI pratiques spécifiques vers ISO 9001v2000

¹Force = Niveau de couverture du chapitre ISO 9001v2000 de la pratique CMMI (S :Fort, M :Moyen, W :Faible).

Les techniques d'estimation des coûts et des charges

Dans cet annexe, nous proposons une revue de la littérature sur les différentes techniques d'estimation des coûts de développement de logiciels. Ainsi, nous présentons brièvement celles se basant sur l'expertise, l'analogie et la stratégie descendante ou ascendante. Dans ce chapitre, l'emphase est mise sur la technique d'estimation se basant sur la modélisation. Nous exposons, en particulier, deux approches pour la modélisation de la relation exprimant l'effort de développement en fonction des facteurs affectant le coût : l'approche *paramétrique* utilisant des techniques statistiques telles que la régression linéaire et l'analyse bayésienne et l'approche *non-paramétrique* utilisant des techniques de l'Intelligence Artificielle telles que les réseaux de neurones, les arbres de régression et les algorithmes génétiques. Pour chaque technique de modélisation, nous présentons les avantages et les inconvénients. Nous terminons ce chapitre par une synthèse sur les travaux de recherche ayant comme objectif la validation de ces techniques sur des bases de projets logiciels historiques.

D.1 Introduction

Le contrôle et la maîtrise du processus de développement d'un logiciel sont basés sur une estimation fiable de son coût. En effet, sans cette estimation, les gestionnaires et les autres responsables d'un projet logiciel ne pourront assigner les budgets et les compétences nécessaires afin de mener à bien le développement du projet. Ainsi, les gestionnaires, s'appuyant généralement sur les résultats des estimations, pourraient commettre des erreurs irréparables, en

prévoyant des plans inadéquats ou en assignant au projet des budgets insuffisants. Ces erreurs pourraient aussi se répercuter sur les utilisateurs. En effet, un allongement du délai de développement initialement prévu du logiciel entraînerait un retard de livraison et d'installation du logiciel. Les utilisateurs seraient donc confrontés à des problèmes d'aspects économique et organisationnel, surtout si, comme c'est le cas en général, ils avaient dressé leurs plannings et mis en place une nouvelle structure en affectant du personnel et/ou en aménageant les locaux par exemple.

Bien que des chiffres précis soient difficiles à établir, il est de plus en plus admis que les coûts de développement de logiciels représentent une part importante des coûts informatiques. Ainsi, dans beaucoup de projets logiciels, le coût de développement dépasse 80% du coût total initial du projet [14]. La figure D.1 résume la situation de l'évolution des coûts de développement par rapport à ceux d'achat de matériel dans un projet logiciel [14]. La croissance de ces coûts ainsi que les difficultés que l'on rencontre pour les prévoir et les contrôler constituent encore aujourd'hui une préoccupation des chercheurs et des responsables informatiques. En effet, ces responsables doivent faire face aux besoins sans cesse grandissants et diversifiés des utilisateurs mais aussi, aux différents problèmes rencontrés au cours du développement d'un logiciel (retard dans les délais, coût croissant, qualité faible, ...).

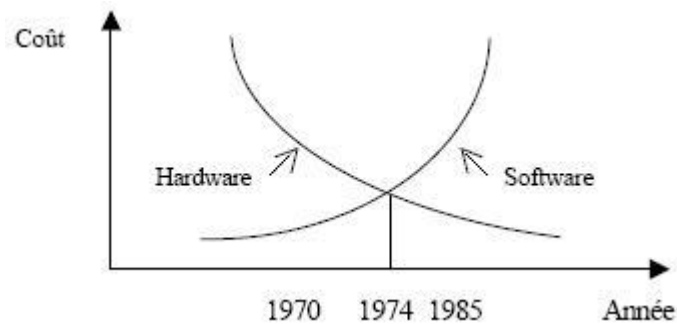


Figure D.1 – Tendence de l'évolution des coûts de développement vs les coûts d'achat de matériel dans un projet logiciel [14].

Vu l'importance d'une estimation fiable du coût dans la réussite d'un projet logiciel, le domaine d'estimation des coûts de logiciels a été l'objet de plusieurs travaux de recherche ainsi que d'une étroite collaboration entre les chercheurs et les industriels afin de développer et de valider des approches d'estimation propres au domaine du génie logiciel. D'ailleurs, nous pouvons constater que le domaine d'estimation des coûts, à travers sa littérature, est parmi les plus fertiles en génie logiciel. L'estimation du coût d'un projet logiciel comporte l'étude des trois attributs suivants : *l'effort*, *le temps* et *le coût*. Ce sont ces trois attributs qu'on désigne souvent dans la littérature par les aspects économiques d'un projet logiciel.

D.2 La mesure Homme-Mois (h.m)

L'effort de développement d'un logiciel représente le nombre de personnes ainsi que le nombre de mois nécessaires pour achever le projet. Par exemple, si 10 personnes ont travaillé pendant 5 mois dans un projet, l'effort de développement de ce projet est alors 50 Homme-Mois. Par conséquent, si on connaît les salaires mensuels de ces intervenants dans le projet, son coût est tout simplement la somme des salaires multipliés par le nombre de mois de travail. Ceci implique donc que les trois attributs : *effort*, *temps* et *coût*, sont étroitement liés. En effet, la majeure partie des coûts de développement d'un logiciel correspond aux frais salariaux des ingénieurs logiciels. Ainsi, le coût d'un projet logiciel est directement proportionnel au nombre d'Homme-Mois nécessaires à la réalisation du projet. Il y a, évidemment, d'autres coûts, par exemple le matériel, les déplacements et les formations du personnel impliqué ; mais ceux-ci sont moins difficiles à établir. En effet, ils ne sont pas basés sur des impondérables tels que la productivité du personnel impliqué dans le projet ou la taille du logiciel à développer.

Du fait que le coût de développement d'un logiciel est déterminé par le nombre de personnes et de mois de travail, c'est-à-dire l'effort de développement, les deux terminologies d'*estimation des coûts* et d'*estimation de l'effort* sont considérées équivalentes en génie logiciel. En général, la plupart des approches d'estimation des coûts de logiciels, déterminent, en première étape, l'effort de développement ; ensuite, elles évaluent le coût en utilisant les montants salariaux du personnel impliqué. L'unité Homme-Mois, utilisée souvent pour mesurer l'effort de développement d'un logiciel, a fait l'objet d'une critique pertinente de Brooks dans son article : *The Mythical Man-Months* (Brooks, 1975) :

" The man-months as a unit for measuring the size of a job is a dangerous and deceptive myth. It implies that men and months are interchangeable. Men and months are interchangeable commodities only when a task can be partitioned among many workers with no communication among them "

Brooks a mené son étude sur un des plus gros logiciels de l'époque, OS/360, et a déduit plusieurs résultats qui ont pu être confirmés par plusieurs expérimentations. L'un de ses résultats principaux a été que les hommes et les mois ne sont pas interchangeables. Ceci est venu réfuter l'hypothèse courante qui affirmait jadis qu'en réduisant le délai normal de développement d'un logiciel d'un facteur donné, les ressources humaines nécessaires étaient tout simplement multipliées par ce facteur. Cette hypothèse reste valable dans certaines industries classiques où une tâche, relevant du processus de développement, peut être facilement divisée et réalisée par un ensemble de personnes sans qu'il y ait de la communication entre ces personnes. Ainsi, dans de tels cas, l'ajout du personnel dans le projet permet de réduire son temps de développement. Cependant, en industrie et surtout l'industrie de logiciels, ce cas est

rarement rencontré. En effet, les activités de développement d'un logiciel ne sont souvent pas partitionnables du fait de leur caractère séquentiel. Elles sont parfois partitionnables avec un taux de communications très élevé entre les intervenants, et très rarement sans communications intenses entre les intervenants, comme le montre la figure D.2.

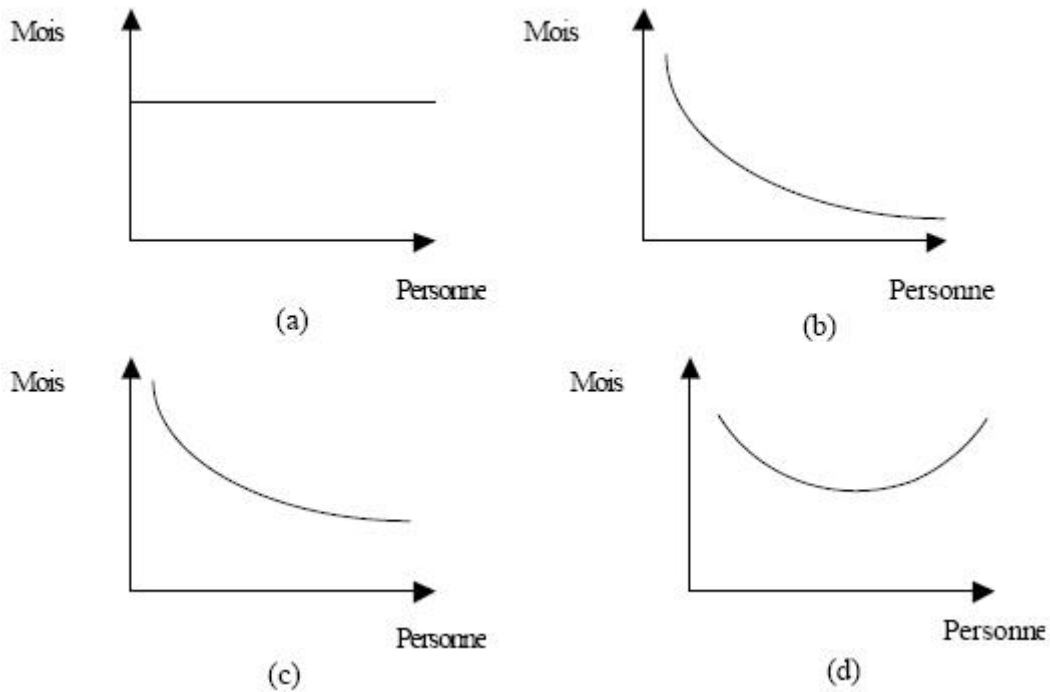


Figure D.2 – Effets de la répartition d'une tâche sur plusieurs personnes.

- (a) Tâche non partitionnable : peu importe le nombre de personnes, le temps de développement reste le même.
- (b) Tâche facilement partitionnable : le nombre de personnes réduit énormément le temps de développement.
- (c) Tâche partitionnable avec un taux faible de communications entre les intervenants : le nombre de personnes réduit moyennement le temps de développement.
- (d) Tâche partitionnable avec un taux de communications très élevé entre les intervenants : le nombre de personnes allonge le temps de développement du fait que l'effort de communications ajouté peut contrer entièrement la répartition de la tâche d'origine (Brooks, 1975).

D.3 Techniques d'estimation des coûts

Un inventaire des diverses techniques d'estimation des coûts est donné par Boehm [14]. Il identifie sept techniques différentes. Celles-ci sont basées sur l'expertise, l'estimation par analogie, la loi de Parkinson (un projet coûtera ce qu'on a décidé de dépenser), le prix du client (un projet coûtera ce que le client est disposé à dépenser), l'estimation ascendante, l'estimation descendante et la modélisation. Cependant, selon Fenton et Pfleeger [42], la loi de Parkinson et le Prix du client ne peuvent être considérés en tant que techniques d'estimation des coûts; Fenton et Pfleeger considèrent que le prix du client est une contrainte au projet qui peut déterminer sa faisabilité tandis que la loi de Parkinson exprime un but plutôt qu'une estimation. Chacune des techniques citées ci-dessus a des avantages et des inconvénients, mais le plus important selon Boehm, est qu'aucune technique ne peut être appliquée de façon exclusive. Dans ce qui suit, nous présentons brièvement les techniques d'estimation basées sur le jugement des experts, l'analogie, l'estimation ascendante et l'estimation descendante. Ensuite, l'accent sera mis sur la technique d'estimation basée sur la modélisation (Sect. D.4).

D.3.1 Jugement des experts

Cette technique consiste à consulter un ou plusieurs experts qui utilisent leurs expériences ainsi que leur compréhension du projet afin de fournir une estimation à son coût. Elle était parmi les premières techniques utilisées en estimation des coûts. Il est préférable d'avoir plusieurs valeurs estimées du coût émanant de plusieurs experts pour alléger les problèmes de subjectivité, de pessimisme et d'optimisme [14]. Il y a plusieurs façons de combiner les différentes valeurs estimées du coût d'un projet logiciel. La plus simple est d'utiliser une des méthodes statistiques telles que la moyenne ou la médiane pour déterminer une estimation du coût. Cette approche présente l'inconvénient d'être sujette aux préjugés des estimations extrêmes. Une autre méthode, référée souvent dans la littérature par la méthode *Delphi*, consiste à réitérer ce processus d'estimation plusieurs fois jusqu'à ce qu'un consensus soit établi. L'idée principale de la méthode Delphi est qu'un coordinateur distribue une description du projet logiciel (souvent le dossier d'analyse des besoins du logiciel) ainsi qu'un formulaire d'estimation aux experts; ensuite, les experts remplissent les formulaires et les renvoient au coordinateur. Ce dernier pourra, dépendamment de la version de la méthode Delphi, convoquer une réunion des experts pour discuter des différentes estimations. Ce processus est réitéré jusqu'à l'adoption d'une estimation par tout le groupe.

Avantages

- Méthode relativement bon marché.
- Estimations précises si les experts ont de l'expérience dans des logiciels similaires.

Inconvénients

- Estimations non précises s'il y a assez peu d'expertise dans le type d'application.

D.3.2 Estimation par analogie

Dans cette technique d'estimation, l'évaluateur compare le projet logiciel dont on veut estimer le coût avec des projets logiciels déjà achevés. Il doit identifier les similarités et les différences entre le nouveau projet et tous les anciens projets. Pour cela, il doit auparavant décrire les projets logiciels par un ensemble de caractéristiques. Les ressemblances entre les caractéristiques du nouveau projet avec celles des projets achevés déterminent les degrés de similarité entre les projets. Ensuite, il utilise ces degrés de similarité pour déterminer une estimation du coût du nouveau projet.

Dans sa première version informelle, l'estimation par analogie a été considérée comme une alternative de l'estimation par les jugements des experts. En effet, les experts raisonnent souvent par analogie quand on leur demande une évaluation du coût d'un logiciel à partir de son dossier d'analyse des besoins. Ainsi, les experts utilisent leurs connaissances sur des projets déjà achevés et déterminent implicitement les similarités et les différences entre le nouveau projet et les projets achevés. Cependant, plusieurs versions formelles de l'estimation par analogie ont été récemment développées [96]. Par ce fait, on la considère de plus en plus comme une technique d'estimation qui se base sur la modélisation.

Avantages

- Estimation précise si des données concernant des projets similaires sont disponibles.

Inconvénients

- Estimation impossible si des projets similaires n'étaient pas réalisés.
- Nécessite la maintenance systématique d'une base de données où on doit décrire auparavant les projets logiciels par un ensemble de caractéristiques.

D.3.3 Estimation ascendante

Dans l'estimation ascendante, le projet logiciel (ou le logiciel) est décomposé en plusieurs tâches (ou composantes) constituant ainsi une arborescence de toutes les tâches (ou composantes) du projet logiciel (ou du logiciel). Tout d'abord, on estime l'effort nécessaire pour chaque tâche du plus bas niveau dans l'arborescence ; ensuite, on détermine progressivement l'effort des autres tâches, se retrouvant dans un niveau supérieur dans l'arborescence, en combinant les efforts nécessaires associés aux sous-tâches. En général, l'effort nécessaire d'une tâche sera la somme de ceux de ses sous-tâches. Cependant, dans le cas des projets logiciels complexes, d'autres techniques plus sophistiquées, telles que celles se basant sur des formules mathématiques ou sur des règles d'induction (si-alors), peuvent être utilisées afin de mettre en valeur la complexité des interfaces de communication entre les tâches.

Avantages

- Méthode précise si la conception a été effectuée avec suffisamment de détails.

Inconvénients

- Possibilité de sous-estimer les coûts des activités au niveau global du logiciel : intégration, documentation . . .

D.3.4 Estimation descendante

Dans l'estimation descendante, on évalue une estimation globale de l'effort de tout le projet logiciel ; ensuite, on répartit cet effort total sur toutes les tâches du projet logiciel.

Dans les deux cas de l'estimation ascendante ou descendante, les valeurs estimées de l'effort sont déterminées en utilisant la technique du jugement des experts, l'estimation par analogie ou un modèle d'estimation.

Avantages

- Prendre en compte des coûts globaux comme celui de l'intégration, de la gestion de configuration, de la documentation . . .

Inconvénients

- Possibilité de sous-estimer le coût d'apporter des solutions à des problèmes techniques de bas niveau (exemple des algorithmes utilisés).

D.4 Les modèles d'estimation des coûts

Les techniques d'estimation, basées sur la modélisation, ont été développées pour pallier aux inconvénients des techniques informelles se basant essentiellement sur la disponibilité des experts. En effet, la non-disponibilité des experts cause, dans la plupart des situations, des retards dans le processus d'estimation des coûts. En plus, le processus d'estimation des coûts avec une technique informelle est très ardu du fait qu'il implique le facteur humain tout au long de la prise de régression. La modélisation s'avère donc une approche prometteuse pour la mise au point d'une technique formelle d'estimation des coûts. Elle a les avantages suivants sur les autres techniques d'estimation :

- Elle permet de rationaliser le processus d'estimation. En effet, la modélisation fournit un processus d'estimation clair, autrement dit, un processus opérationnel et praticable.
- Elle offre plusieurs possibilités aux évaluateurs afin d'améliorer la précision des estimations fournies. En effet, les évaluateurs pourront ajouter, supprimer et/ou modifier les paramètres du modèle et analyser ainsi les impacts de ces changements sur la précision des estimations.
- Elle offre une aide précieuse pour une bonne gestion d'un projet logiciel (formules de répartition de l'effort total sur toutes les phases de développement, formules du comptage du code réutilisé, ...).
- Elle permet l'automatisation et la programmation du processus d'estimation (logiciels d'estimation des coûts).

Essentiellement, les modèles d'estimation des coûts se basent d'abord sur l'identification des facteurs influençant les coûts de développement de logiciels (*cost drivers*) et deuxièmement sur l'identification de la nature de la relation exprimant l'effort en fonction de ces facteurs. L'attribut *taille du logiciel* a été souvent considéré comme étant le facteur le plus significatif pour l'estimation de l'effort de développement. Cependant, d'autres facteurs tel que l'expérience du personnel (analystes, concepteurs, programmeurs, ...) impliqué dans le projet, les méthodes utilisées dans le développement, et le degré de réutilisation ont été aussi pris en compte dans la plupart des modèles d'estimation quand la taille d'un logiciel était insuffisante pour expliquer les variations des efforts de développement des projets logiciels.

L'identification de la nature de la relation exprimant l'effort de développement en fonction des facteurs du coût représente l'étape principale dans le processus de mise au point d'un modèle d'estimation des coûts. Cette relation sera représentée par la fonction f :

$$\text{Effort} = f(X_1 + X_2 + \dots + X_M) \tag{D.1}$$

où $X_1, X_2 \dots X_M$ sont les facteurs affectant le coût de développement ; f est la relation exprimant l'effort en fonction des facteurs X_i .

La détermination de la fonction f est souvent basée sur l'analyse des données historiques collectées sur les projets logiciels déjà achevés. Ces données historiques peuvent être représentées par une matrice $N \times (M+1)$:

$$\begin{pmatrix} & \text{Effort réel} & X_1 & X_2 & \dots & X_M \\ \text{Projet 1} & E_1 & x_{11} & x_{12} & \dots & x_{1M} \\ \text{Projet 2} & E_2 & x_{21} & x_{22} & \dots & x_{2M} \\ \text{Projet 3} & E_3 & x_{31} & x_{32} & \dots & x_{3M} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \text{Projet N} & E_N & x_{N1} & x_{N2} & \dots & x_{NM} \end{pmatrix} \tag{D.2}$$

où N représente le nombre de projets logiciels déjà achevés et M le nombre de variables décrivant les projets logiciels (facteurs affectant le coût). La colonne *Effort réel* représente l'effort réel de chaque projet logiciel déjà achevé, E_i . Les données historiques sur des projets logiciels déjà achevés représentent des exemples de la relation exprimant l'effort en fonction des facteurs du coût. Le défi est donc de reconstruire cette relation à partir de ces exemples afin qu'elle soit utilisée pour prédire le coût des nouveaux projets logiciels (nouvelles données).

Il existe plusieurs techniques qui permettent la reconstruction de la fonction f à partir d'un ensemble d'exemples (associations entrées - sorties). Ces techniques proviennent principalement de trois domaines : les Statistiques, l'Approximation des fonctions et l'Intelligence artificielle. Néanmoins, nous pouvons les regrouper en deux catégories :

- Les techniques *paramétriques* : elles présument au préalable que la fonction f a une forme bien déterminée (linéaire ou analytique). Ensuite, elles déterminent les paramètres de la fonction f . Le nombre de paramètres de la fonction f est connu d'avance. Des exemples de ces techniques sont la régression linéaire simple ou multiple, l'analyse bayésienne et l'interpolation polynomiale.
- Les techniques *non-paramétriques* : elles ne fixent aucune forme à la fonction f . Elles développent des procédures d'approximation de la fonction f sans connaître sa forme. Des exemples de ces techniques sont les réseaux de neurones, le raisonnement à base de cas (*Case-Based Reasoning* -CBR-), les arbres de régression et les règles d'induction (si-alors).

Nous décrivons, dans les deux sections suivantes, des exemples de modèles d'estimation des coûts mis au point en utilisant des techniques paramétriques et/ou non-paramétriques.

D.5 Les modèles paramétriques d'estimation des coûts

Les modèles paramétriques d'estimation des coûts de développement de logiciels se basent essentiellement sur une équation centrale exprimant l'effort en fonction d'un certain nombre de variables considérées comme étant les principales conductrices du coût. En plus, le modèle pourra utiliser des tables ou des formules prédéterminées pour évaluer les entrées de son équation centrale. La mise au point d'un modèle paramétrique d'estimation suppose donc que la forme de l'équation centrale du modèle est connue. Cette équation centrale représente la relation exprimant l'effort en fonction des facteurs affectant les coûts.

Du fait que les techniques paramétriques étaient les premières appliquées dans le domaine de l'estimation des coûts, plusieurs modèles paramétriques ont été développés. Des exemples de ces modèles sont le modèle d'Halstead (Halstead, 1975), IBM-FSD (Walston et Felix, 1977), PUTNAM-SLIM (Putnam, 1978), COCOMO'81 (Boehm, 1981), COCOMOII (Boehm, 2000), et ceux se basant sur les points de fonctions (Albrecht et Gaffney, 1983; Abran et Robillard, 1994; Matson, Barrett et Mellichamp, 1994). Les modèles paramétriques peuvent être classifiés selon trois critères :

- La forme de l'équation centrale du modèle (linéaire ou analytique) ;
- le modèle est-il Statique ou Dynamique ? Un modèle est dit Dynamique si son équation centrale contient la variable *temps*. Ainsi le modèle fournit, en plus de l'effort total, l'effort instantané durant tout le processus de développement. Un modèle est dit Statique si son équation centrale ne contient pas la variable *temps* ;
- le modèle est-il Global ou Détaillé ? Un modèle est dit Détaillé si, en plus de l'effort total, il fournit aussi sa répartition sur toutes les activités et les phases du cycle de développement. Notamment, on peut remarquer qu'un modèle Dynamique est nécessairement Détaillé. Un modèle Global ne fournit que l'effort total. Un modèle Statique n'est pas nécessairement un modèle Global.

D.5.1 Les modèles linéaires

Ces modèles adoptent une équation centrale linéaire de la forme :

$$\text{Effort} = a_0 + a_1x_1 + a_2x_2 + \dots + a_Mx_M \quad (\text{D.3})$$

où les x_i sont les facteurs affectant l'effort et les a_i sont des coefficients choisis pour fournir le meilleur ajustement à l'ensemble des données historiques observées. La mise au point de tels modèles est souvent accomplie en utilisant les techniques statistiques de régression linéaire simple ou multiple selon le nombre de variables x_i . La régression linéaire consiste à analyser un ensemble de données sur des projets logiciels déjà achevés (comme celles de la matrice D.2) afin de reconstruire la relation, supposée être linéaire, exprimant l'effort (variable dépendante) en fonction des variables x_i (variables indépendantes). Dans ce qui suit, nous présentons l'état de l'art sur les applications de la régression linéaire simple et multiple pour la mise au point des modèles linéaires d'estimation des coûts.

Régression linéaire simple

L'application de la régression linéaire simple, en estimation des coûts, a souvent considéré la taille du logiciel comme étant la variable indépendante la plus significative pour la prédiction de l'effort :

$$\text{Effort} = A + B \times \text{taille} \quad (\text{D.4})$$

où A et B sont des constantes. La taille d'un logiciel est souvent mesurée par le nombre de lignes de son code source (*Kilo Source Line Of Code* -KSLOC-) ou le nombre de fonctionnalités qu'il offre à ses utilisateurs, calculé par la méthode des points de fonctions (*Function Points* -FP- ; Albrecht et Gaffney, 1983 ; Abran et Robillard, 1994). Les deux paramètres A et B sont généralement déterminés en utilisant la méthode des moindres carrés qui consiste à chercher le minimum de la somme des carrés des résidus e_i :

$$F(a, b) = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (\text{Effort}_{i,\text{réel}} - A - B \times \text{taille}_i)^2 \quad (\text{D.5})$$

Le résidu e_i représente, pour chaque projet P_i , la différence entre la valeur réelle de l'effort de développement de P_i , $\text{Effort}_{i,\text{réel}}$, et la valeur estimée de son effort par le modèle linéaire de l'équation D.4. Les valeurs A et B qui minimisent la fonction $F(a, b)$ sont celles adoptées par le modèle linéaire de l'équation D.4.

Après la détermination des deux paramètres du modèle, A et B , à partir de la base de données des projets logiciels, on teste la qualité de l'ajustement obtenu par le modèle. L'indicateur le plus couramment employé est le coefficient de détermination, R^2 représentant la proportion de la variabilité de l'effort expliquée par la taille ; ce coefficient est définie comme le rapport de l'écart type sur la moyenne arithmétique ; R^2 est compris entre 0 et 1 ; une valeur de R^2 proche de 1 implique que la taille explique bien l'effort d'un projet logiciel. Cependant, d'autres indicateurs relevant de la distribution des résidus e_i , et surtout des résidus

standardisés e_{ir} , sont nécessaires pour détecter les défaillances d'un modèle linéaire ayant un R^2 voisin de 1. En effet, les résidus standardisés doivent être distribués indépendamment selon une loi gaussienne centrée réduite (moyenne=0 et variance=1). Ces suppositions sur les résidus standardisés d'un modèle linéaire sont examinées par au moins deux graphiques :

- 1) Le graphique des e_{ir} en fonction de la taille ou l'effort estimé pour vérifier l'indépendance de la distribution des e_{ir} ;
- 2) Le graphique des e_{ir} pour vérifier la normalité de la distribution des e_{ir} .

Ainsi, si l'une des deux suppositions n'est pas satisfaite, une transformation de la variable indépendante (taille) et/ou de la variable dépendante (effort) est nécessaire pour rendre linéaire la relation exprimant l'effort en fonction de la taille. Autrement, la relation serait considérée comme étant non-linéaire et par conséquent la méthode de la régression linéaire est inadaptée pour la mise au point du modèle.

La transformation logarithmique est la plus utilisée en estimation des coûts du fait que les modèles adoptent souvent une équation centrale de la forme :

$$\text{Effort} = B \times \text{taille}^C \quad (\text{D.6})$$

Pour rendre linéaire ce genre de modèles, on transforme les deux variables *effort* et *taille* en $\log(\text{effort})$ et $\log(\text{taille})$. Ainsi, l'équation centrale transformée du modèle s'écrit comme suit :

$$\log(\text{Effort}) = \log(B) + C \times \log(\text{taille}) \quad (\text{D.7})$$

Par conséquent, on utilise la régression linéaire pour déterminer les deux constantes $\log(B)$ et C qui minimisent la somme :

$$\sum (\log(\text{Effort}_{\text{réel}}) - \log(B) - C \times \log(\text{taille}))^2 \quad (\text{D.8})$$

On détermine ainsi les deux constantes initiales B et C de l'équation centrale originale du modèle (Équation D.6). Il faut noter que la minimisation de l'équation D.9 n'entraîne pas nécessairement la minimisation de la somme des différences entre les efforts réels et estimés exprimés sur l'échelle originale. Cependant, des résultats satisfaisants sont souvent obtenus. Des exemples de modèles, mis au point en appliquant la régression linéaire après la transformation logarithmique des deux variables *effort* et *taille*, sont :

Effort = $5.2 \times \text{KLOC}^{0.91}$	Walston-Felix
Effort = $3.2 \times \text{KLOC}^{1.05}$	COCOMO simple
Effort = $5.288 \times \text{KLOC}^{1.047}$	Doty (pour KLOC > 9)

Régression linéaire multiple

La régression linéaire simple est utilisée pour la mise au point d'un modèle linéaire d'estimation dans le cas où le nombre de facteurs considérés dans le modèle est égal à un (souvent la taille du logiciel). Dans le cas où le nombre de ces facteurs est supérieur ou égal à deux, la mise au point du modèle fait appel à la régression linéaire multiple. Des exemples de facteurs affectant le coût, autres que la taille, sont l'expérience du personnel impliqué dans le développement, la complexité de l'application et la méthodologie de développement. En général, ces facteurs jouent un rôle d'ajustement de l'équation, dite Nominale, exprimant l'effort seulement en fonction de la taille du logiciel. En effet, Boehm avait constaté que des logiciels ayant les mêmes tailles ne requéraient pas nécessairement les mêmes efforts de développement [14]. La prise en compte d'autres facteurs dans l'équation centrale d'un modèle linéaire d'estimation a donc pour objectif l'explication de la variation des coûts de deux projets ayant les mêmes tailles. L'application de la régression linéaire multiple en estimation des coûts permet la mise au point de modèles linéaires ayant une équation centrale comme celle de l'équation D.3. Les paramètres a_i de l'équation sont ceux qui minimisent la somme des carrés :

$$\sum_{i=1}^n e_i^2 = \sum_{i=1}^n (\text{Effort}_{i,\text{réel}} - a_0 - a_1 x_{1,i} - a_2 x_{2,i} - \dots - a_M x_{M,i})^2 \quad (\text{D.9})$$

Les critères utilisés dans l'évaluation de la qualité de l'ajustement, dans le cas de la régression linéaire simple, sont aussi applicables dans le cas de la régression linéaire multiple. En effet, un coefficient de détermination R^2 proche de un signifie que l'ajustement est convenable ; aussi, l'analyse des résidus e_{ir} permet de vérifier les suppositions du modèle linéaire sur l'indépendance de la distribution des e_{ir} dont les graphes en fonction de la variable dépendante (effort) et les variables indépendantes (x_i) du modèle ne doivent montrer aucune tendance. En plus, il y a deux autres critères d'évaluation propres au cas de la régression linéaire multiple :

- Le premier consiste à mesurer l'impact de la multicollinéarité sur l'instabilité du modèle. En effet, si les facteurs influençant le coût de développement x_i sont très corrélés entre eux, les estimations du modèle seront entachées d'erreurs considérables même si R^2 a une valeur proche de 1. En général, l'étude de l'effet de la colinéarité entre les facteurs x_i s'effectue au moyen des facteurs d'inflation de la variance et des valeurs propres de la matrice de corrélation entre les x_i .
- Le deuxième consiste à déterminer le sous-ensemble de tous les facteurs x_i qui donne des estimations satisfaisantes des coûts. Cela permettra au modèle d'éliminer les effets des facteurs non significatifs et redondants. On procède souvent par élimination successive ou par ajout successif des facteurs dans le modèle. Dans la première stratégie, on élimine celui qui provoque la diminution la plus faible de R^2 ; dans la deuxième stratégie, on

ajoute celui qui provoque la progression la plus élevée de R^2 . Dans les deux cas, un test d'arrêt est obligatoire. Une autre variante souvent utilisée en estimation des coûts est la méthode dite *Stepwise régression*. Elle consiste à effectuer des tests de signification, à chaque étape, pour ne pas introduire une variable non significative et éliminer éventuellement des variables déjà introduites qui ne seraient plus informatives compte tenu de la dernière variable sélectionnée. La méthode s'arrête quand on ne peut ni ajouter ni retrancher des variables du modèle.

Comme dans le cas de la régression linéaire simple, souvent en estimation des coûts, la relation initiale exprimant l'effort en fonction des facteurs du coût n'est pas linéaire. La pratique consiste à utiliser la transformation logarithmique pour rendre linéaire cette relation initiale. Cette stratégie a souvent été adoptée par la communauté pour la calibration et pour l'adaptation des modèles linéaires d'estimation à des environnements différents de ceux à partir desquels ces modèles ont été initialement développés.

D.5.2 Les modèles analytiques

Dans la section précédente, nous avons étudié les modèles linéaires d'estimation, c'est-à-dire ceux qui adoptent une équation centrale linéaire, éventuellement après la transformation de la variable dépendante et/ou les variables indépendantes. Dans cette section, nous présentons une deuxième catégorie de modèles paramétriques : les modèles analytiques.

Les modèles analytiques adoptent une équation centrale non-linéaire. Leur utilisation est nécessaire quand la relation initiale, exprimant l'effort en fonction des facteurs du coût, n'est pas toujours linéaire et qu'aucune transformation des variables n'est possible afin de rendre cette relation linéaire (Fig. D.3). Dans la littérature, on ne trouve que peu de modèles analytiques qui ont été développés (Halstead, 1975 ; Putnam, 1978). En effet :

- Souvent, un modèle linéaire satisfaisant et concurrent peut être mis en place, éventuellement après la transformation des variables.
- Quand cette relation n'est pas linéaire, l'identification de la forme de la relation exprimant l'effort en fonction des facteurs du coût est souvent très ardue.
- Les praticiens, dans leur majorité, préfèrent apporter des améliorations de détail au modèle linéaire concurrent plutôt que d'étudier le modèle non-linéaire initial.

Dans ce qui suit, nous présentons brièvement deux exemples de modèles analytiques, les plus connus : le modèle d'Halstead et le modèle de Putnam.

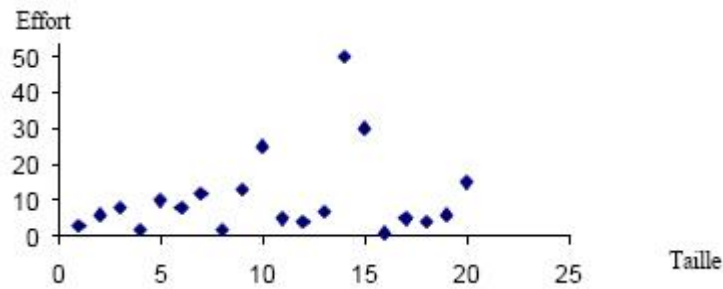


Figure D.3 – Exemple où la relation Effort-taille n'est pas linéaire.

Le modèle d'Halstead

En 1972, Maurice Halstead a entrepris une étude empirique des algorithmes afin de tester l'hypothèse selon laquelle, dans un programme, le comptage des opérateurs (expressions verbales) et des opérandes (noms ou désignations de données) pouvait faire apparaître des corrélations avec le nombre des erreurs contenues dans les algorithmes (Halstead, 1975). Suite à cette étude empirique, Halstead avait défini quatre métriques basées sur l'analyse lexicale du code source :

- μ_1 = nombre d'opérateurs distincts
- μ_2 = nombre d'opérandes distincts
- N_1 = nombre total d'utilisation des opérateurs
- N_2 = nombre total d'utilisation des opérandes

Ensuite, Halstead avait utilisé ces quatre métriques pour développer une panoplie de formules d'évaluation de certains attributs d'un logiciel tels que le vocabulaire et le volume d'un programme, l'effort et la durée de codage d'un programme. Les deux équations en rapport avec l'effort et la durée de codage d'un programme sont respectivement :

$$\begin{aligned} \text{Effort} &= \frac{\mu_1 N_2 (N_1 + N_2) \log_2(\mu_1 + \mu_2)}{2\mu_2} \\ \text{Durée} &= \frac{\text{Effort}}{18} \end{aligned} \quad (\text{D.10})$$

Cette équation d'estimation de l'effort à partir des quatre métriques de base d'Halstead n'a pas eu beaucoup d'intérêt dans la plupart des travaux de recherche en estimation des coûts du fait que plusieurs hypothèses adoptées par Halstead sont considérées comme erronées. En plus, la plupart des mesures proposées par Halstead sont basées sur le code source d'un logiciel ; les autres phases du cycle de développement telles que la spécification et la conception ne sont pas prises en considération bien qu'elles requièrent souvent un effort considérable.

Le modèle de Putnam

En 1978, Putnam a proposé un modèle d'estimation de l'effort de développement et de maintenance d'un logiciel en se basant sur les remarques et sur les conclusions que Norden avait déjà déduit en étudiant des projets de développement dans beaucoup de domaines autres que le génie logiciel [89]. En effet, Norden avait observé que la distribution de l'effort total sur tout le cycle de développement d'un projet, dans plusieurs domaines, pouvait s'approcher avec une grande précision par la fonction de Rayleigh (Fig. D.4). Ainsi, Putnam avait utilisé cette fonction pour exprimer la répartition de l'effort total de développement sur tout le cycle de vie d'un logiciel (développement et maintenance) :

$$\text{Effort}(t) = CT(1 - e^{-\frac{1}{2}(\frac{t}{t_d})^2}) \tag{D.11}$$

où $\text{Effort}(t)$ représente l'effort cumulé à l'instant t , CT est l'effort total de tout le cycle de vie du logiciel (depuis le développement du logiciel jusqu'à sa "mort"), et t_d est le temps de développement.

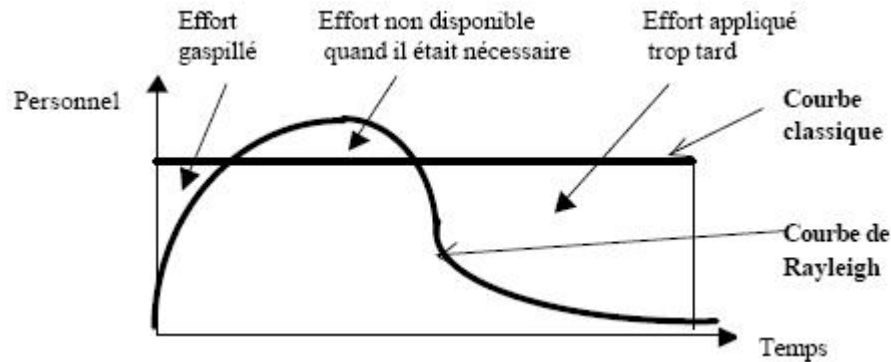


Figure D.4 – Répartition de l'effort selon la courbe de Rayleigh et la courbe classique.

Pour exprimer l'effort en fonction de la taille d'un logiciel, Putnam a constaté, après une étude empirique des projets réels, que la productivité dépend de la difficulté d'un projet selon l'équation suivante :

$$\frac{ISL}{\text{Effort}(t_d)} = kD^{-\frac{2}{3}} \tag{D.12}$$

où D est la difficulté du projet, ISL le nombre d'instructions sources livrées, et k un coefficient de correction. Ainsi, Putnam a proposé l'équation suivante pour l'effort total d'un projet logiciel :

$$ISL = E CT^{\frac{1}{3}} t_d^{\frac{4}{3}} \tag{D.13}$$

où E est un facteur technologique représentant l'effet de 14 conducteurs de coût tels que l'effet du langage utilisé, la méthodologie de développement et la procédure d'assurance qualité utilisée.

D.5.3 Les limitations des modèles paramétriques

Les modèles paramétriques d'estimation des coûts (linéaires ou analytiques) supposent la connaissance de la forme de la relation exprimant l'effort en fonction des facteurs conducteurs du coût. Autrement dit, la forme de l'équation centrale du modèle est bien connue et il suffit donc de déterminer les paramètres de cette équation en utilisant des techniques statistiques ou des méthodes théoriques du domaine de l'approximation des fonctions. En estimation des coûts de logiciels, on détermine souvent la forme de l'équation centrale du modèle en utilisant des connaissances relevant de l'environnement étudié ou en adoptant des formes déjà établies dans d'autres environnements. Ainsi, la mise au point d'un modèle paramétrique d'estimation des coûts nécessite au préalable l'identification de la forme de son équation centrale. Cela représente une limitation majeure car souvent :

- La relation entre l'effort et les conducteurs du coût ne semble avoir, dans plusieurs cas, aucune forme prédéterminée (Fig. D.3),
- les connaissances sur la relation exprimant l'effort en fonction des conducteurs du coût ne sont pas disponibles, et/ou
- Les connaissances disponibles sont imprécises et/ou incertaines.

Aussi, au fur et à mesure que le modèle linéaire est opérationnel dans un organisme, le modèle doit être capable d'apprendre afin de considérer les nouveaux changements survenus dans l'organisme au niveau du développement et/ou de la maintenance de logiciels. Malheureusement, les techniques statistiques classiques, comme la méthode de régression, n'offrent pas de mécanismes pour un apprentissage automatique. Cependant, elles adoptent des mécanismes élémentaires tels que la mise à jour de la base des données historiques, la mise à jour des contributions des observations, et la mise à jour des coefficients du modèle afin d'introduire l'apprentissage dans le modèle. Ces mécanismes sont insuffisants et très superficiels et ne peuvent doter le modèle d'un degré d'apprentissage acceptable.

Certaines limitations des modèles paramétriques sont spécifiques aux cas des modèles linéaires, en particulier à la technique de régression linéaire, comme la performance de généralisation, la présence des observations extrêmes et la contribution des observations dans la régression :

- La généralisation explique le comportement du modèle dans les cas de nouvelles observations. En estimation des coûts, elle représente donc la capacité du modèle à fournir

des estimations acceptables pour de nouveaux projets logiciels. En général, il faut être prudent, quand on se sert d'un modèle linéaire, pour prédire le coût d'un projet logiciel qui s'éloigne des limites de l'échantillon des projets logiciels ayant servi à la mise au point du modèle linéaire. Bien qu'elle n'apparaisse pas dans les équations du modèle, cette précaution est obligatoire dans le cas des modèles linéaires. Par exemple, supposons qu'un modèle linéaire ne considère que la taille pour l'estimation de l'effort et que la taille de tous les logiciels de la base de données historiques est comprise entre 10KISL et 40KISL ; dans ce cas, l'utilisation de ce modèle pour estimer le coût d'un logiciel de taille 100KISL pourra conduire à une estimation aberrante.

- L'étude des contributions de projets logiciels historiques dans la stabilité du modèle linéaire est aussi un aspect très intéressant qui doit être pris en compte dans la mise au point du modèle linéaire. On évalue souvent la contribution de chaque projet logiciel en appliquant la technique de régression sans prendre en considération ce projet logiciel. Les projets logiciels, impliquant des variations importantes dans la détermination des coefficients du modèle, sont considérés comme influençant le plus la stabilité du modèle. Afin d'éviter les influences de ces observations, la pratique consiste à les éliminer ou à pondérer les observations dans la technique de régression. La deuxième solution semble être la meilleure bien qu'il n'y ait pas une technique standard pour l'affectation des poids aux observations.

Une autre limitation des modèles paramétriques d'estimation des coûts est qu'ils requièrent une adaptation ou une calibration quand on veut les appliquer dans des environnements différents de ceux où ils ont été développés :

Calibration : La calibration d'un modèle paramétrique est sa reformulation sur une base de données de projets logiciels d'un environnement autre que celui du modèle initial. Ainsi, les équations du modèle calibré, y compris son équation centrale, ont les mêmes structures que celles du modèle initial mais pour lesquelles les coefficients sont recalculés.

Adaptation : L'adaptation d'un modèle paramétrique est la modification des formes de certaines de ses équations ainsi que l'ajout ou la suppression de facteurs affectant le coût dans le modèle.

En effet, le modèle paramétrique représente l'état de l'environnement étudié à l'aide de la forme de son équation centrale ainsi que les conducteurs du coût qu'il considère. Cependant, vu que les environnements de développement et/ou de maintenance de logiciels n'ont pas généralement les mêmes caractéristiques, l'application d'un modèle, initialement mis au point dans un environnement à un autre, n'est pas souhaitable. Des exemples de caractéristiques qui peuvent différencier deux environnements de développement de logiciels sont le type d'applications logicielles développées (Scientifique, Gestion, Business, ...), les méthodes et les

outils de développement utilisés et les contraintes imposées sur l'application (fiabilité, portabilité, complexité, ...). Souvent en estimation des coûts, les chercheurs et les industriels utilisaient ces deux techniques (calibration et adaptation) pour mettre au point des modèles propres à leurs environnements à partir de modèles déjà mis au point dans d'autres environnements. Bien que la calibration (ou l'adaptation) représente, dans certains cas, une solution au problème de l'utilisation d'un modèle paramétrique ailleurs que dans son environnement, elle reste insuffisante quand les différences entre les environnements sont très flagrantes.

D.6 Les modèles non-paramétriques d'estimation des coûts

Les modèles non-paramétriques d'estimation des coûts ont été développés pour remédier aux inconvénients cités ci-dessus dans les modèles paramétriques. Ils n'ont pas une équation centrale d'estimation de l'effort. Ainsi, les modèles non-paramétriques n'exigent au préalable aucune forme à la relation exprimant l'effort en fonction des conducteurs du coût. Ils modélisent cette relation en utilisant des techniques d'intelligence artificielle telles que les réseaux de neurones, les algorithmes génétiques et les arbres de régression. Les modèles non-paramétriques représentent donc une alternative prometteuse quand la relation entre l'effort et les conducteurs du coût semble n'avoir aucune forme prédéfinie (Fig. D.3). Dans cette section, nous présentons brièvement les modèles non-paramétriques utilisant les réseaux de neurones, les arbres de régression et les algorithmes génétiques.

D.6.1 Les modèles d'estimation utilisant les réseaux de neurones

La modélisation par les réseaux de neurones est inspirée de la structure biologique du cerveau humain. Un réseau de neurones est caractérisé par son architecture, son algorithme d'apprentissage et ses fonctions d'activation. Les modèles de réseaux de neurones peuvent être classifiés en deux catégories principales : les réseaux de neurones *feedforward* où il n'y a aucune boucle dans l'architecture du réseau, et les réseaux *récurrents* où une ou plusieurs boucles récursives apparaissent dans l'architecture du réseau. Le Perceptron multi-couches utilisant l'algorithme d'apprentissage de rétro-propagation est souvent le plus adopté en estimation des coûts de logiciels. Dans ce modèle, les neurones sont organisés en couches et chaque couche ne peut avoir des connexions que vers la couche suivante. La figure D.5 illustre un exemple d'un tel réseau pour l'estimation de l'effort de développement de logiciels. Le réseau produit un résultat (effort) en propageant ses entrées initiales (facteurs du coût ou attributs du projet logiciel) à travers les différents neurones du réseau jusqu'à la sortie. Chaque neurone

d'une couche calcule sa sortie en appliquant sa fonction d'activation à ses entrées. Souvent, la fonction d'activation d'un neurone est la fonction *Sigmoïde* définie par :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (\text{D.14})$$

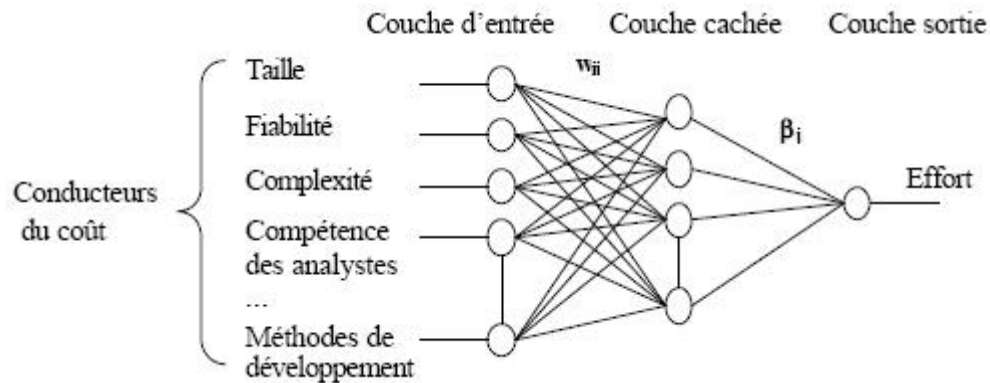


Figure D.5 – Architecture d'un réseau de neurone à trois couches pour l'estimation de l'effort.

La configuration d'un réseau de neurones pour l'estimation de l'effort de développement de logiciels nécessite le choix d'une architecture adéquate, un algorithme d'apprentissage et des fonctions d'activation. Dans le cas d'un Perceptron multi-couches, la configuration se restreint au choix des différents paramètres de sa topologie tels que le nombre de couches cachées, le nombre de neurones par couche, et les valeurs initiales des poids de connexions (w_{ij} et β_j). Dans la pratique, ces paramètres sont déterminés en se fiant à une base de données de projets logiciels déjà achevés. Ainsi, on divise la base de données en deux ensembles : un ensemble de projets logiciels pour entraîner le réseau et l'autre pour tester le réseau. L'architecture qui permettra au réseau de neurones de fournir des estimations plus précises sera choisie.

L'utilisation des réseaux de neurones pour estimer un projet logiciel a trois avantages principaux :

- Les réseaux de neurones permettent l'apprentissage automatique à partir des situations et des résultats précédents. L'apprentissage est très important pour les modèles d'estimation des coûts car l'industrie des logiciels est en évolution croissante.
- Les réseaux de neurones permettent un traitement parallèle de l'information. Ainsi, le processus d'estimation du modèle profitera de la grande puissance de calcul de toutes les machines disponibles surtout que, souvent, le processus de formulation d'une estimation engage des calculs très intensifs.
- Les réseaux de neurones peuvent modéliser les relations complexes existantes entre la variable dépendante (coût ou effort) et les variables indépendantes du modèle (facteurs

de coût). En effet, les réseaux de neurones sont reconnus en tant qu'approximateurs universels de fonctions continues à valeurs dans l'ensemble des réels.

Cependant, l'approche neuronique a trois limitations qui l'empêchent d'être acceptée comme une pratique usuelle en estimation des coûts :

- L'approche neuronique est considérée comme étant une approche boîte noire. Par conséquent, il est difficile d'expliquer et d'interpréter son processus.
- Les réseaux de neurones ont été utilisés spécifiquement pour les problèmes de classification et de catégorisation, alors qu'en estimation des coûts, nous traitons un problème de généralisation et non pas de classification.
- Il n'existe aucune démarche standard pour le choix des différents paramètres de la topologie d'un réseau de neurones (nombre de couches, nombre d'unités de traitement par couche, valeurs initiales des poids de connexions, ...).

D.6.2 Les modèles d'estimation utilisant les arbres de régression

Bien que les arbres de régression et ceux de décision soient basés sur le même principe, ils ont deux buts différents. Les arbres de régression peuvent être utilisés quand la valeur de sortie à prédire est dans une échelle d'intervalle, alors que les arbres de décision, également connus sous le nom d'arbres de classification, sont employés pour prédire la classe de sortie d'une observation : il s'agit alors de l'échelle nominale ou ordinale (catégorielle). Les deux algorithmes fonctionnent en apprenant les règles nécessaires pour classer les cas pris dans un ensemble de données d'entraînement.

Les arbres de régression constituent un outil de modélisation simple et de plus en plus répandu par la diversité des applications qui lui font appel. Plusieurs approches de construction de ces arbres ont été développées ; ces approches dépendent du type de la connaissance disponible du domaine étudié selon qu'elle est discrète ou continue. La construction de l'arbre de régression s'effectue à partir d'un échantillon de données historiques dit d'apprentissage.

La figure D.6, montre un exemple d'arbre de régression binaire dont les noeuds sont des attributs d'entrée (comme le *Nombre d'écrans*) et les arcs sont des expressions booléennes (comme le *Nombre d'écrans > 10*) et les feuilles sont les valeurs de sortie qui représentent différentes moyennes du temps de développement (comme la *Moyenne* de 112h).

La figure D.7 illustre un arbre de régression binaire construit à partir des projets du COCOMO'81. Dans cet arbre de régression binaire, la racine représente la variable *AKDSI* (*Adjusted Kilo Delivered Source Instructions*) de chaque projet logiciel ; si la valeur du *AKDSI* du projet logiciel est supérieure à 315,5, alors son effort est estimé à 9000 h.m ; sinon, on passe au deuxième niveau de l'arbre où il y a un seul nœud représentant aussi la variable *AKDSI*

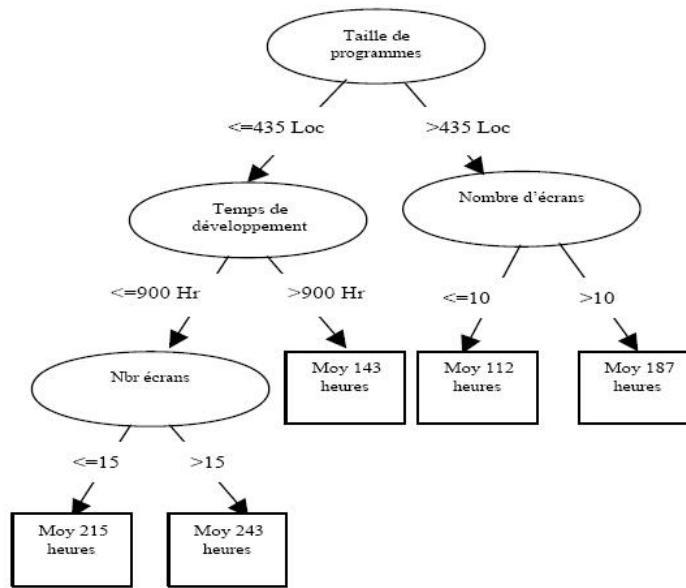


Figure D.6 – Arbre de régression pour la prédiction du temps de développement.

et ainsi de suite jusqu'à l'une des feuilles de l'arbre.

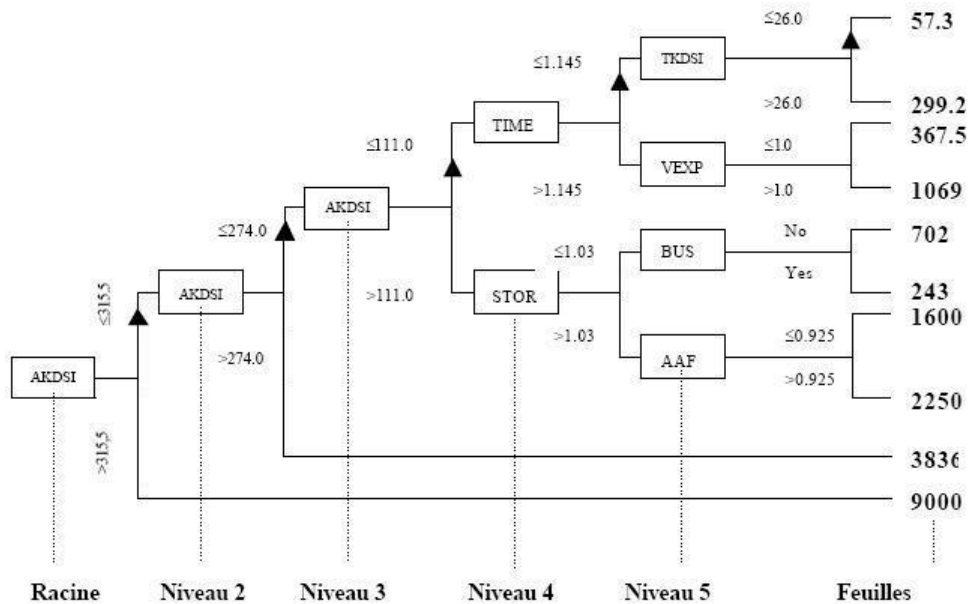


Figure D.7 – Arbre de régression construit à partir des 63 projets logiciels du COCOMO'81.

En général, la plupart des algorithmes de construction d'un arbre de régression binaire, à partir d'un ensemble de projets logiciels historiques, sont basés sur une stratégie descendante.

Cette stratégie consiste à choisir, parmi tous les attributs décrivant le projet logiciel, celui qui divise mieux l'ensemble de tous les projets logiciels en deux sous-ensembles disjoints. Souvent, on choisit l'attribut qui minimise l'erreur moyenne quadratique (*MSE*) de la variable dépendante (l'effort de développement) des projets logiciels historiques. L'erreur moyenne quadratique d'un sous-ensemble S de projets logiciels est calculée par :

$$\text{MSE}(S) = \frac{\sum_{k \in S} (\text{Effort}_k - \overline{\text{Effort}})^2}{|S|} \quad (\text{D.15})$$

où Effort_k est l'effort réel du $k^{\text{ème}}$ projet logiciel de S et $\overline{\text{Effort}}$ est la moyenne arithmétique des Effort_k .

Il y a deux cas à distinguer dans le choix de l'attribut qui divise mieux l'ensemble des projets logiciels :

- L'attribut est mesuré par des valeurs discrètes. Dans ce cas, pour chaque attribut V_i , on divise l'ensemble de tous les projets logiciels T en des sous-ensembles T_{ij} contenant des projets logiciels ayant la même valeur A_j de l'attribut V_i .
- L'attribut est mesuré par des valeurs numériques continues. Dans ce cas, pour chaque attribut V_i , on divise l'ensemble de tous les projets logiciels en $k-1$ sous-ensembles selon l'ordre des valeurs observées de V_i (où k est le nombre de valeurs observées de V_i).

Ainsi, on choisit l'attribut A_i qui maximise la différence :

$$\text{MSE} = \text{MSE}(T) - \sum_j \text{MSE}_{ij} \quad (\text{D.16})$$

L'expansion d'un noeud de l'arbre s'arrête quand le nombre de projets logiciels classifiés dans ce noeud est assez petit ou ces projets logiciels sont jugés similaires. Dans ce cas, le noeud représente une feuille de l'arbre et l'effort estimé de tout nouveau projet logiciel, qui sera classifié dans ce noeud, est calculé en fonction des valeurs réelles des efforts de tous les projets logiciels historiques de ce noeud (en général, on considère la moyenne ou la médiane).

La modélisation de l'estimation des coûts par les arbres de régression présente l'avantage d'être facile à expliquer et à utiliser contrairement au cas de celle utilisant les réseaux de neurones. En effet, un chemin dans l'arbre de régression peut être représenté par une règle du type :

Si condition1 et condition2 et ... et conditionN Alors Effort=CTS

Par exemple, dans la figure D.7, le chemin indiqué par des flèches peut être représenté par la

règle suivante :

Si (AKDSI \leq 111.0) **et** (TIME \leq 1.415) **et** (TKDSI \leq 26.0)
alors Effort = 57.3 h.m

Ce genre de règles est facilement interprétable car les règles ressemblent à celles qu'on utilise dans notre quotidien. Ainsi, une modélisation par un arbre de régression binaire n'est qu'un cas particulier d'un système à base de règles d'induction Si-Alors.

D.6.3 Les modèles d'estimation utilisant les algorithmes génétiques

Les algorithmes génétiques sont inspirés des concepts de l'évolution et de la sélection naturelle élaborés par Charles Darwin. Ils ont été utilisés comme des méthodes heuristiques dans les problèmes d'optimisation où l'espace de recherche de la solution est de taille assez grande. Le pionnier du domaine est John Holland qui a le premier tenté d'implanter artificiellement des systèmes évolutifs (Holland, 1975). L'idée de base de la théorie Darwinienne de l'évolution postule que seulement les meilleurs éléments d'une population survivent et transmettent à leurs descendants leurs caractéristiques biologiques à travers des opérations génétiques telles que le croisement, la mutation et la sélection. Dans l'implantation informatique du principe de l'évolution, le problème étudié est représenté par une chaîne binaire (formée de 0 et 1) symbolisant les chromosomes de son équivalent biologique. Ainsi, à partir d'une population, c'est-à-dire à partir d'un ensemble de chaînes binaires (chromosomes),

- les chromosomes les mieux adaptés se reproduisent le plus souvent et contribuent davantage aux populations futures (*sélection*),
- lors de la reproduction, les chromosomes des parents sont combinés et mélangés pour reproduire les chromosomes des enfants (*croisement*),
- le résultat du croisement peut être à son tour modifié par des perturbations aléatoires (*mutation*).

En général, un algorithme génétique est composé de trois étapes principales bien que d'autres variations sont possibles :

- 1) Générer aléatoirement une population initiale de N chromosomes (solutions) ;
- 2) répéter les sous-étapes suivantes jusqu'à une condition d'arrêt (souvent on fixe un temps d'exécution ou un nombre d'itérations) ;
 - (a) évaluer la *fitness*¹ de chaque chromosome de la population et sélectionner ceux

¹contribution génétique d'un individu à la génération suivante, généralement exprimée en termes du nombre de descendants produits par individu et capables de se reproduire à nouveau.

- ayant les meilleurs scores ;
- (b) reproduire une nouvelle population de chromosomes en appliquant les deux opérations génétiques (croisement et mutation) aux chromosomes sélectionnés dans l'étape précédente ;
 - (c) remplacer l'ancienne population par la nouvelle population obtenue lors des 2 étapes précédentes ;
- 3) choisir comme solution au problème celle qui a le meilleur score dans la population finale.

<pre> ALGORITHME GENETIQUE(p_c, p_m, \dots) 1 Initialiser G_0 2 MEILLEUR \leftarrow meilleur chromosome de G_0 3 MEILLEURSOL \leftarrow MEILLEUR \triangleright initialiser la meilleure solution 4 pour $t \leftarrow 0$ à T 5 $Q \leftarrow$ paires parmi les meilleurs membres de la génération G_t 6 $Q' \leftarrow$ enfants des paires dans Q en utilisant le croisement et la mutation 7 remplacer les membres les plus faibles de G_t par Q' pour créer G_{t+1} 8 MEILLEUR \leftarrow meilleur chromosome in G_{t+1} 9 si MEILLEUR est plus fort que MEILLEURSOL alors 10 MEILLEURSOL \leftarrow MEILLEUR 11 retourner MEILLEURSOL </pre>

Figure D.8 – Résumé de l'algorithme génétique.

Le principe de l'algorithme génétique est résumé sur la figure D.8. L'utilisation de l'algorithme génétique nécessite la définition des mécanismes suivants que nous détaillons dans l'annexe ?? :

- 1) Un codage des éléments d'une population.
- 2) Un choix de la population initiale.
- 3) Une fonction d'évaluation des individus, ou fonction de *fitness*.
- 4) Un opérateur de croisement.
- 5) Un opérateur de mutation.
- 6) Un principe de sélection d'individus.

Récemment, les algorithmes génétiques ont été appliqués en estimation des coûts de développement de logiciels (Shukla, 2000). Shukla a utilisé les algorithmes génétiques dans l'apprentissage d'un Perceptron multi-couches afin de trouver le vecteur des poids W qui permette au Perceptron multi-couches de fournir des estimations acceptables. Ses expérimentations se

sont basées sur la base de données du COCOMO'81. Il identifie un Perceptron multi-couches PN par un vecteur de poids $W(PN)$. À chaque vecteur de poids $W(PN)$, il associe un chromosome de M éléments où M est le nombre d'éléments de $W(PN)$. La figure D.9 illustre un exemple d'un Perceptron multi-couches ainsi que le chromosome qui lui est associé ; ce Perceptron comporte 18 poids de connexions auxquels il associe un chromosome de 18 éléments. Chaque élément du chromosome représente un poids du vecteur $W(PN)$. Pour coder un poids, il lui réserve dix bits. Ainsi, le chromosome associé sera composé de 180 bits. La fonction de *fitness* utilisée pour sélectionner les chromosomes dans une population est :

$$F(W) = \frac{1}{E(W)} \quad \text{où} \quad E(W) = \frac{1}{T} \sum_{i=1}^T (\text{Effort}_{\text{estimé},i} - \text{Effort}_{\text{réel},i})^2 \quad (\text{D.17})$$

Shukla utilise une technique hybride de sélection des chromosomes qui vont survivre dans la prochaine population. Cette technique considère les deux premiers chromosomes ayant les meilleurs scores $F(W)$ et applique sur les autres la stratégie de la roue roulante dans laquelle chaque chromosome reçoit une part de la roue proportionnelle à son score $F(W)$. Pour l'opérateur de croisement, Shukla a utilisé une version modifiée de l'opérateur standard afin d'éviter que des bits, représentant des poids appartenant à différentes couches du Perceptron, soient interchangés. Les autres paramètres de l'algorithme, tels que la taille de la population, la probabilité accordée à l'opérateur de croisement ainsi que celle accordée à l'opérateur de mutation, ont été choisis en exécutant plusieurs fois l'algorithme sur les données du COCOMO'81.

Une autre variante de l'algorithme génétique est la programmation génétique, qui a été utilisé par Dolado en 2000, et Burgess et Lefley en 2001 pour le développement de modèles d'estimation des coûts de logiciels. Cette technique est présenté plus en détail dans l'annexe ??.

L'utilisation du principe de l'évolution en estimation des coûts présente l'avantage de fournir une recherche heuristique de l'équation exprimant l'effort en fonction des conducteurs du coût. Ainsi, à partir d'un ensemble d'équations initiales, l'algorithme recherche celle qui représente adéquatement la relation. Ceci présente un avantage sur les techniques paramétriques classiques telles que la régression linéaire où une seule forme de l'équation est évaluée. Cependant, comme dans le cas des réseaux de neurones, la configuration d'un algorithme génétique (ou un programme génétique) nécessite le choix de plusieurs paramètres tels que la fonction de *fitness*, la taille de chaque population et la taille de l'arbre binaire représentant l'équation. Ces paramètres sont souvent déterminés par expérimentations. Ils dépendent donc de la base

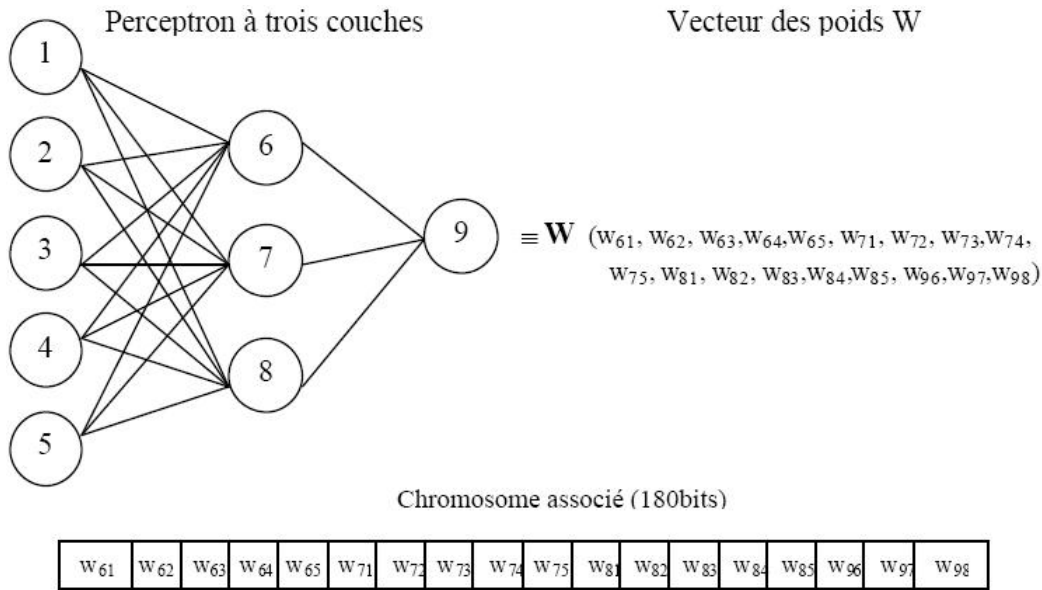


Figure D.9 – Exemple d’un Perceptron multi-couches et sa représentation avec un chromosome.

de projets logiciels utilisée. Aussi, nous pouvons adresser aux algorithmes génétiques la même critique que celle mentionnée dans le cas des réseaux de neurones au sujet de l’interprétation et de la compréhension du processus d’un algorithme génétique.

D.7 La validation des modèles d’estimation des coûts

Dans la littérature, on retrouve un ensemble de critères de validation de n’importe quel modèle d’estimation. Ces critères représentent les objectifs du modèle :

Précision : c’est l’objectif principal de n’importe quel modèle d’estimation. En effet, le rôle du modèle est de fournir des estimations précises (à un degré acceptable) aux coûts des nouveaux projets. Ainsi, pour chaque nouveau projet logiciel, le coût estimé doit être proche de son coût réel.

Constructive : le modèle doit être facilement interprétable afin qu’on puisse, quand il est nécessaire, expliquer son processus d’estimation. Ceci permettra aux estimateurs de découvrir d’autres règles (connaissances) d’estimation des coûts qui n’étaient pas connues. Ainsi, le modèle pourra être considéré comme un fournisseur, en plus des estimations, de nouvelles connaissances dans le domaine d’estimation des coûts (*Knowledge Discovery*).

Stable : le modèle doit fournir des estimations assez proches quand ses entrées (valeurs des

facteurs influençant le coût) sont de même. Il doit donc éviter les discontinuités dans l'évaluation des coûts. Ce cas peut se présenter quand certaines des entrées du modèle sont binaires.

Opérationnel : le modèle doit être facile à utiliser. Ainsi, il doit utiliser le minimum de facteurs ; ses options doivent être facile à configurer, et il doit être efficace (consomme peu de ressources en termes de temps d'exécution et d'espace mémoire).

Le premier critère de la précision d'un modèle est celui qui a été le plus débattu dans la littérature. Ainsi, dans la littérature, de nombreuses études empiriques ont été menées pour évaluer la précision des estimations de plusieurs modèles paramétriques et/ou non-paramétriques. En général, ces études empiriques consistent, premièrement, à la mise au point de modèles d'estimation à partir d'une (ou plusieurs) base de projets logiciels déjà achevés et deuxièmement, à l'évaluation de la performance de ces modèles sur la même (ou une autre) base de projets logiciels.

L'indicateur le plus utilisé est la valeur absolue de l'erreur relative d'une estimation (*Magnitude Relative Error*) MRE définit par :

$$\text{MRE} = \left| \frac{\text{Effort}_{\text{réel}} - \text{Effort}_{\text{estimé}}}{\text{Effort}_{\text{réel}}} \right| \quad (\text{D.18})$$

Autant la valeur du MRE d'une estimation est petite autant l'estimation est précise. Souvent, dans la littérature, on considère une estimation ayant une MRE inférieure ou égale à 0,25 comme étant acceptable.

Ensuite, on utilise le pourcentage de projets logiciels ayant une MRE inférieure ou égale à une valeur p (souvent $p=0,25$), $\text{Pred}(p)$:

$$\text{Pred}(p) = \frac{K}{N} \quad (\text{D.19})$$

où K est le nombre de projets logiciels ayant une *MRE* inférieure ou égale à p , N est le nombre total de projets logiciels de test. Dans la littérature, un modèle ayant une valeur de $\text{Pred}(0,25)$ égale à 70% est dit acceptable.

Un autre indicateur utilisé est la moyenne des erreurs relatives, MMRE, où N correspond au nombre de projets. Cet indicateur est définie par :

$$\text{MMRE} = \frac{1}{N} \sum_{i=1}^N \left| \frac{\text{Effort}_{\text{réel},i} - \text{Effort}_{\text{estimé},i}}{\text{Effort}_{\text{réel},i}} \right| \times 100 \quad (\text{D.20})$$

La plupart des études empiriques ont montré qu'aucune technique n'a pu prouver qu'elle performe mieux que les autres dans toutes les situations. En effet, la performance d'un modèle d'estimation dépend, au moins, de trois facteurs :

- La technique utilisée pour la mise au point du modèle. En effet, les techniques paramétriques exigent la connaissance de la forme de la relation exprimant l'effort en fonction des conducteurs du coût. Ainsi, si la forme adoptée ne représente pas convenablement la réalité, le modèle risque de fournir des estimations moins précises. Dans le cas des modèles non-paramétriques, l'idée consiste à trouver une approximation à la forme réelle de la relation. Ainsi, s'il s'avère que cette relation a une forme bien déterminée telles que linéaire ou analytique, les modèles paramétriques ont plus de chances d'être plus précis que les modèles non-paramétriques.
- Les caractéristiques de la base des projets logiciels à partir de laquelle le modèle a été mis au point. En effet, les paramètres d'un modèle d'estimation se définissent, en général, en fonction de ces projets logiciels. Des exemples de caractéristiques dont les impacts sur la performance d'un modèle ont été étudiés dans la littérature sont :

- 1) La taille de la base de projets logiciels, c'est-à-dire le nombre de projets logiciels qu'elle contient. Souvent un grand nombre de projets logiciels permet au modèle de mieux représenter son environnement.

- 2) Le nombre d'attributs décrivant les projets logiciels. Souvent, dans la littérature un modèle d'estimation ne considère que les attributs ayant des données réelles disponibles et qui sont considérés comme étant les principaux conducteurs du coût. Il exclut donc des attributs qui n'ont pas de données enregistrées bien qu'ils puissent être significatifs pour l'estimation des coûts. Aussi, la sélection d'un sous-ensemble d'attributs qui décrit mieux un projet logiciel représente une tâche très ardue dans le processus d'estimation des coûts.

- Les caractéristiques de la base de projets logiciels utilisés dans l'évaluation de la performance du modèle. En effet, il est peu probable qu'un modèle fournisse des estimations précises pour des projets logiciels qui diffèrent beaucoup de ceux de la base de projets historiques. En général, afin d'éviter les effets des différences entre les projets logiciels ayant été utilisés pour la mise au point du modèle et ceux qui seront utilisés pour évaluer la précision du modèle, la pratique consiste à diviser l'ensemble des projets logiciels en deux sous-ensembles : le premier servira pour la mise au point du modèle et le deuxième pour le test du modèle.

D.8 Conclusion

L'estimation des coûts de développement de logiciels est l'un des problèmes critiques les plus débattus en métrologie de logiciels. La maîtrise et l'évaluation de ces coûts semblent encore préoccuper aussi bien les chercheurs que les responsables et les chefs de projets logiciels. Dans ce chapitre, nous avons présenté un ensemble de techniques d'estimation des coûts, spécifiquement celles se basant sur la modélisation. Ainsi, nous avons décrit des exemples de modèles paramétriques et non-paramétriques. Chaque type de modèle d'estimation a des avantages et des inconvénients et aucun modèle n'a pu prouver qu'il performe mieux que tous les autres dans toutes les situations.

Les paramètres de la méthode d'estimation proposée

E.1 Les composants des Points de Fonction (FP)

Identifier les RET, DET et FTR

Ce sont les éléments qui permettent d'affecter un degré de complexité à chacun des 5 composants principaux des Points de Fonction et de distinguer les différentes transactions au sein du système.

	Définition	Exemples
<i>Record Element Type (RET)</i>	Un RET est un groupement de champs identifiables par un utilisateur. C'est plus facile de regarder les groupements logiques des données pour pouvoir identifier les RETs.	En-tête d'une facture, Corps d'une facture, Information sur un utilisateur. . .
<i>File Type Referenced (FTR)</i>	Un FTR est un type de fichier référencé par un processus élémentaire. Un FTR doit être un Fichier Logique Interne (ILF) E.1 ou un Fichier d'Interface Externe (EIF) E.1	Tous les ILF et les EIF de l'application, voir les exemples ?? et ??
<i>Data Element Type (DET)</i>	Un DET est un champ non répété, un attribut identifiable par un utilisateur. C'est un champ ou élément graphique d'un GUI au sens informatique.	nom, prénom, date, message d'erreur, valeur calculée, bouton. . .

Tableau E.1 – Tableau de présentation des RET, DET et FTR.

Fichiers Logiques Internes (ILF)

Un ILF est un groupe de données logiquement lié du point de vue d'un utilisateur ou un groupe d'informations de contrôles maintenues à l'intérieur de l'application. Ce groupe fait partie de l'application.

Un ILF sert à garder des données maintenues via un ou plusieurs processus élémentaires (cas d'utilisation) appartenant à l'application.

Évaluation de la complexité

	1-19 FTR	20-50 FTR	> 50 FTR
1 RET	basse	basse	moyenne
2-5 RET	basse	moyenne	haute
> 5 RET	moyenne	haute	haute

Tableau E.2 – Tableau d'évaluation de la complexité des ILF.

	basse	moyenne	haute
Nombre de Points de Fonction	7	10	15

Tableau E.3 – Tableau de conversion des ILF en FP.

Fichiers d'Interface Externes (EIF)

Un EIF est un groupe de données liées logiquement du point de vue d'un utilisateur ou un groupe d'informations de contrôle référencés par l'application mais qui n'appartiennent pas à cette application.

Le but premier d'un EIF est de garder ces données référencées par un ou plusieurs processus élémentaires (cas d'utilisation) et mis à jour par une autre application que celle mesurée, ces données peuvent être à l'intérieur ou à l'extérieur des frontières de l'application.

Évaluation de la complexité

	1-19 FTR	20-50 FTR	> 50 FTR
1 RET	basse	basse	moyenne
2-5 RET	basse	moyenne	haute
> 5 RET	moyenne	haute	haute

Tableau E.4 – Tableau d'évaluation de la complexité des EIF.

	basse	moyenne	haute
Nombre de Points de Fonction	5	7	10

Tableau E.5 – Tableau de conversion des EIF en FP.

Entrées Externes (EI)

Un EI est un processus élémentaire de l'application qui traite des données ou des informations de contrôle qui viennent de l'extérieur du périmètre de l'application. Les données traitées ont un impact sur une ou plusieurs ILF/EIF. Une information de contrôle peut n'avoir aucun impact sur les ILF/EIF.

La raison première d'un EI est de gérer (création, mise à jour, suppression) un ou plusieurs ILF/EIF et/ou modifier la logique d'exécution d'une application.

Évaluation de la complexité

	1-4 DET	5-15 DET	> 15 DET
0-1 FTR	basse	basse	moyenne
2 FTR	basse	moyenne	haute
> 2 FTR	moyenne	haute	haute

Tableau E.6 – Tableau d'évaluation de la complexité des EI.

	basse	moyenne	haute
Nombre de Points de Fonction	3	4	6

Tableau E.7 – Tableau de conversion des EI en FP.

Sorties Externes (EO)

Un EO est un processus élémentaire de l'application qui génère des données ou des informations de contrôle pour l'extérieur de l'application.

La raison première d'une EO est de fournir des informations à un utilisateur via une logique d'exécution. Cette logique d'exécution doit transformer les données sortantes. Donc les EO représentent des processus qui extraient des données de l'application en sollicitant un traitement de ces données avant traitement.

Évaluation de la complexité

	1-5 DET	6-19 DET	> 19 DET
0-1 FTR	basse	basse	moyenne
2-3 FTR	basse	moyenne	haute
> 3 FTR	moyenne	haute	haute

Tableau E.8 – Tableau d'évaluation de la complexité des EO.

	basse	moyenne	haute
Nombre de Points de Fonction	4	5	7

Tableau E.9 – Tableau de conversion des EO en FP.

Requêtes Externes (EQ)

Un EQ est un processus élémentaire de l'application qui retrouve des données ou des informations de contrôle et dont le résultat est renvoyé à l'extérieur de l'application.

La raison première d'une EQ est de retrouver des informations d'un ILF ou d'un EIF. Un EQ sélectionne, retrouve des données, qu'il transmet vers l'extérieur. Quelques spécificités des EQ :

- pas de mise à jour des ILF ;
- résulte d'un traitement d'extraction de données, sans transformation ;
- ne contiennent pas de donnée calculée en sortie (exemple des totaux).

Évaluation de la complexité

	1-5 DET	6-19 DET	> 19 DET
0-1 FTR	basse	basse	moyenne
2-3 FTR	basse	moyenne	haute
> 3 FTR	moyenne	haute	haute

Tableau E.10 – Tableau d'évaluation de la complexité des EQ.

	basse	moyenne	haute
Nombre de Points de Fonction	3	4	6

Tableau E.11 – Tableau de conversion des EQ en FP.

E.2 Les facteurs d'échelle

Ce sont des facteurs qui font partie du modèle précoce et du modèle post-architecture et qui permettent de qualifier le type de développement et la nature du projet à réaliser. La sélection de ces facteurs est basée sur le fait qu'ils sont une source significative de variation exponentielle de l'effort d'un projet et/ou la variation de productivité. Ces facteurs sont au nombre de cinq :

Précédence (PREC)

PREC caractérise le degré de familiarité et/ou innovation de l'application. Si plusieurs projets de même nature ont déjà été réalisés et que la connaissance a été capitalisée, alors PREC est très élevé.

Flexibilité de développement (FLEX)

FLEX caractérise le niveau de flexibilité auquel le développement doit se conformer. Par exemple, plus un système doit être conforme aux besoins d'une organisation cible soumise aux changements, plus la flexibilité sera élevée.

Architecture / Résolution des risques (RESL)

RESL caractérise la façon dont les incertitudes, en particulier celles qui concernent l'architecture, sont résolues. Si tous les risques sont résolus, alors RESL est extrêmement élevé. Si la gestion de risques est absente, ou très occasionnelle, alors RESL est très bas. En fait, les risques sont identifiés lors du développement, ce qui occasionne des *feed-back* très coûteux.

Cohésion de l'équipe (TEAM)

TEAM caractérise la cohésion de l'équipe de développement, en particulier pour ce qui concerne les relations MOA, MOE et les divers sous-traitants. Les différences de culture et la bonne intégration des différents métiers sont toujours la source d'ambiguïtés qui peuvent être très coûteuses.

maturité des processus (PMAT)

PMAT caractérise le niveau de maturité du processus de développement au sens du CMM.

La figure E.1 permet d'estimer la complexité des facteurs d'échelle présentés ci-dessus. La complexité est décliné en six degrés : très bas, bas, nominal, élevé, très élevé, extrêmement élevé.

E.3 Les facteurs de coûts du modèle post-architecture

Les 17 facteurs de coûts du modèle post-architecture sont utilisés pour ajuster l'effort nominal et refléter l'impact des attributs pouvant influencer le développement.

Le modèle peut être utilisé pour estimer un projet en entier ou un projet module par module et dans ce dernier cas l'estimation des facteurs de coûts peut différer d'un module à un autre selon le contexte, seul l'estimation du facteur concernant la contrainte de planification est commune à tous les modules.

Facteurs d'échelle (SFj)	Très bas	Bas	Nominal	Elevé	Très élevé	Extrêmement élevé
PREC Précédence	Totalement sans précédent	En grande partie sans précédent	En partie sans précédent	Plutôt familial	En grande partie familial	Totalement familial
FLEX Flexibilité de développement	Figé	Flexibilité occasionnelle	Une certaine flexibilité	En conformité générale	D'une certaine conformité	Objectifs principaux
RESL Architecture / résolution des risques	Un peu (20%)	En partie (40%)	Souvent (60%)	Généralement (75%)	La plupart du temps (90%)	Toujours (100%)
TEAM Cohésion de l'équipe	Interactions très difficiles	Interactions plutôt difficiles	Coopération de base	Coopération appréciable	Coopération importante	Coopération sans faille
PMAT Maturité des processus	Niveau 1 bas	Niveau 1 haut	Niveau 2	Niveau 3	Niveau 4	Niveau 5

Figure E.1 – Les facteurs d'échelle

E.3.1 Les attributs du produit

Les attributs du produit représentent la variation de l'effort causée par les caractéristiques du produit à développer. Un produit qui est complexe, qui a des exigences de haute fiabilité, ou qui utilise une grande base de données de test exigera plus d'effort. Il y a 5 attributs de produit et la complexité de l'application a l'influence la plus forte sur l'effort évalué.

Contraintes de fiabilité du logiciel (RELY)

RELY mesure l'importance accordée à la fiabilité en fonction des conséquences d'une panne. Si l'effet d'un échec du logiciel est insignifiant alors RELY est très bas. Si un échec risquerait des vies humaines alors RELY est très élevé.

Taille de la base de données (DATA)

DATA permet de capturer l'effet des fortes exigences de données d'essai sur le développement du produit. L'évaluation est faite en calculant D/P , où D est la taille de la base de données de test en octet et P la taille du programme en SLOC. La raison pour laquelle la taille de la base de données est importante à considérer est qu'il faut estimer l'effort exigé pour produire les données d'essai qui seront utilisées pour tester le programme. Ainsi, DATA permet de capturer l'effort nécessaire pour assembler et maintenir les données exigées pour achever l'essai du programme.

Besoins de documentation couverts (DOCU)

DOCU permet de mesurer la documentation produite tout au long du cycle de vie du logiciel. L'échelle d'évaluation va de très bas à très haut. La tentative d'économiser les efforts via des niveaux de documentation bas générera généralement des dépenses supplémentaires pendant la phase de maintenance du logiciel.

Complexité de l'application (CPLX)

CPLX permet de mesurer le degré de complexité générale du logiciel. La complexité est estimée à partir de 5 critères : Les contrôles, la complexité des calculs, le type des entrées/sorties, la gestion des structures des données et la gestion de l'interface utilisateur.

À travers la table E.2 et en sélectionnant les critères qui caractérisent le logiciel, l'évaluation de la complexité est la moyenne pondérée des évaluations des critères choisis.

E.3. LES FACTEURS DE COÛTS DU MODÈLE POST-ARCHITECTURE

Niveaux	Contrôle	Calcul	Entrées / Sorties	Gestion des structures des données	Gestion de l'interface Utilisateur
Très bas	Code séquentiel avec quelques opérateurs de PS : DOs, CASEs, IFTHENELSEs. Décomposition simple de modules par appels de procédures ou scripts.	Evaluation d'expressions simples : e.g., $A=B+C*(D-E)$	Instructions simples de lecture/écriture, formatage simple.	Tableaux de structure de données de base en mémoire centrale. Requêtes simples vers des bases de données.	Ecrans en entrée, générateurs de rapports.
Bas	Imbrication séquentielle d'opérateurs de PS. La plupart du temps prédicats simples.	Evaluation d'expressions, du type, $D=\text{SQRT}(B^2-4*AC)$	Pas de connaissance spécifique d'un processeur effectuée à un niveau GET/PUT.	Un fichier sans modification de ses structures de données, pas de fichiers intermédiaires. Requêtes modérément complexes vers des bases de données.	Utilisation de générateurs simples d'interfaces graphiques.
Nominal	La plupart du temps, imbrication simple. Un peu de contrôle entre modules. Tables de décision. Appels simples: envoi de messages, en particulier traitement distribué via des primitives middleware.	Utilisation de fonctions mathématiques et statistiques standards. Opérations vectorielles et matricielles de base.	Traitement des E/S, en particulier sélection des périphériques, vérification du statut de fonctionnement et traitement des erreurs.	Plusieurs fichiers en entrée et un fichier en sortie. Modification des structures de données simples, éditions simples. Requêtes complexes vers des bases de données, mises à jour.	Utilisation simple de bibliothèques d'objets graphiques.
Elevé	Opérateurs de SP fortement imbriqués composés de nombreux prédicats. Gestion de piles et files d'attente. Traitement distribué homogène. Contrôle temps réel sur la couche logiciel monoprocesseur.	Analyse numérique de base : interpolation, équations différentielles. Convergence et précision des flottants.	Opérations effectuées au niveau physique d'E/S (adressage physique, recherche et lecture ...). Recourcement des E/S optimisé.	Déclencheurs activés par le contenu des flots de données. Restauration complexe des données.	Développement et extension de bibliothèques d'objets graphiques. Traitement vocal en E/S multimédia.
Très élevé	Programmation récursive et réentrante. Gestion des interruptions avec priorités fixes. Synchronisation des tâches, appels complexes, traitement distribué homogène. Contrôle temps réel sur la couche matériel monoprocesseur.	Analyse numérique complexe mais structurée. Calcul tensoriel, équations différentielles partielles.	Opérations dédiées au diagnostic, masquage, traitement des interruptions. Gestion des lignes de communication. Systèmes enfous à haute performance.	Gestion des BDs distribuées. Déclencheurs complexes. Recherches optimisées ...	Traitement 2D/3D, modérément complexe, graphiques dynamiques, multimédia.
Extrêmement élevé	Ordonancement de ressources multiples avec changement dynamique de priorités. Contrôle temps réel sur des architectures multiprocesseurs.	Analyse numérique difficile et non structurée : analyse précise de données aléatoires émanant de bruits. Parallélisation complexe.	Programmation dépendante du temps d'accès aux périphériques, microprogrammation systèmes enfous à performance critique.	Structures relationnelles dynamiquement couplés. Gestion des données en langage naturel.	Multimédia complexe, réalité virtuelle.

Figure E.2 – Détermination de la complexité de CPLX

Réutilisation de composants de l'application (RUSE)

RUSE prend en compte l'effort nécessaire pour construire des composants en vue d'une ré-utilisation ultérieure, ce qui implique des composants plus génériques, une documentation plus importante, des tests approfondis. . .

Les différents degrés de complexité de RUSE sont :

- "**Aucune réutilisation**"
- "**Au niveau projet**" pourrait s'appliquer à la réutilisation de modules à l'intérieur du projet uniquement. à l'intérieur de plusieurs projets d'une même structure (service) uniquement.
- "**Au niveau programme**" pourrait s'appliquer à la réutilisation dans des projets différents pour une même organisation.
- "**Au niveau ligne de produit**" pourrait s'appliquer si la réutilisation est étendue à des organisations multiples pour un seul type d'application.
- "**Au niveau de plusieurs lignes de produit**" pourrait s'appliquer à la réutilisation de modules dans plusieurs types d'application pour des organisations multiples.

E.3.2 Les attributs de la plate-forme

La plate-forme désigne dans ce contexte la complexité matérielle et logicielle de la machine cible. Ainsi, les attributs de la plate-forme permettent de déterminer la qualité de service de la machine cible en terme de stabilité, de durée d'exécution et d'occupation mémoire.

Contraintes sur la durée d'exécution (TIME)

TIME est une mesure qui permet d'évaluer la disponibilité de la machine cible en mesurant la contrainte imposée sur le temps d'exécution du logiciel. L'évaluation est exprimée en termes de pourcentage de temps d'exécution consommé par rapport au temps machine.

Contraintes d'occupation mémoire (STOR)

STOR permet de mesurer la contrainte imposée sur l'espace de stockage des données. L'évaluation est exprimée en termes de pourcentage de taille de stockage nécessaire à l'application par rapport à la taille de stockage disponible.

Étant donné l'augmentation remarquable de la vitesse des processeurs et de la taille de stockage, on peut se demander si ces variables de contrainte sont toujours appropriées. Ce-

pendant, les applications elles-mêmes demandent de plus en plus de ressources ce qui justifie que ces facteurs restent toujours appropriés.

Stabilité de la plate-forme d'exploitation (PVOL)

PVOL mesure le taux de changement de l'infrastructure sur laquelle repose le logiciel. Le terme infrastructure regroupe toute les parties matérielles et logicielles de la machine cible (Système d'exploitation, Base de données...). On distingue des modifications majeures et des modifications mineures.

E.3.3 Les attributs du personnel

Après la taille de produit, les attributs du personnel ont l'influence la plus forte dans la détermination de l'effort exigé pour développer un produit logiciel. Ces attributs permettent d'évaluer la capacité de l'équipe de développement, architectes et développeurs, et leur expérience.

Maturité des architectes (ACAP)

Les architectes sont les personnes qui travaillent sur les spécifications, la conception générale et la conception détaillée. Les attributs majeurs qui devraient être considérés sont la capacité d'analyse et de conception, l'efficacité et le niveau de perfectionnement, et la capacité à communiquer et coopérer. Le niveau d'expérience n'est pas inclus dans ce facteur, il est traité par les facteurs APEX, LTEX et PLEX. Le facteur ACAP est exprimé en terme de pourcentage du niveau d'acquisition de toute l'équipe de ces attributs majeurs.

Maturité des programmeurs (PCAP)

L'évaluation devrait être basée sur les possibilités de toute l'équipe de développement. Les facteurs principaux qui devraient être considérés dans l'estimation sont les capacités de l'équipe, l'efficacité et le niveau de perfectionnement, et la capacité à communiquer et coopérer. L'expérience des programmeurs ne devrait pas être considérée ici, elle est évaluée avec les facteurs APEX, LTEX et PLEX. Le facteur PCAP est exprimé en terme de pourcentage du niveau d'acquisition de toute l'équipe de ces attributs majeurs.

Turnover du personnel (PCON)

L'échelle d'évaluation de PCON est mesurée en termes du turnover annuel du personnel du projet. La stabilité de l'équipe crée des automatismes qui permettent d'accélérer le déroulement du projet.

Expérience du domaine applicatif (APEX)

L'évaluation de ce facteur de coût dépend du niveau d'expérience de l'équipe de projet développant le système avec ce type d'applications. Les estimations sont définies en termes de durée d'expérience de l'équipe de projet avec ce type d'application.

Pratique des langages et outils de programmation (LTEX)

LTEX permet d'évaluer le niveau d'expérience du langage et outils de programmation de l'équipe de projet. Le développement du logiciel inclut l'utilisation des outils de programmation, d'analyse et de conception, de gestion de la configuration, de planification et de gestion de projet. . . Ainsi, en plus de l'expérience de la pratique du langage, l'expérience des outils de support du projet affecte l'effort de développement.

Expérience d'utilisation de la plate-forme d'exploitation (PLEX)

Le modèle post-architecture élargit l'influence de productivité à travers l'expérience de la plate-forme d'exploitation, à travers le facteur PLEX, en identifiant l'importance de comprendre l'utilisation d'une plate-forme plus puissante, y compris des interfaces utilisateur graphiques plus conviviales, la gestion des bases de données, la gestion du réseau. . .

E.3.4 Les attributs du projet

Les facteurs de projet expliquent les influences de l'environnement du projet sur l'effort estimé, ces facteurs regroupent l'utilisation des outils modernes de programmation, la (ou les) localisation(s) de l'équipe de développement et les contraintes de planification.

Utilisation d'outils logiciels (TOOL)

TOOL permet de mesurer le degré d'automatisation des outils utilisés dans le projet de développement. L'estimation de l'outil s'étend de la simple édition et codage aux outils intégrés de gestion du cycle de vie, des processus et de la réutilisation.

Développement sur plusieurs sites différents (SITE)

SITE permet de mesurer le degré de coopération entre les différents sites intervenant dans le développement. Cette coopération est exprimée en terme d'emplacement géographique des équipes de développement.

Contraintes de planification (SCED)

SCED permet de mesurer le degré de compression ou d'expansion du temps nécessaire pour finir le projet en fonction de la contrainte de temps imposée à l'équipe de développement. L'évaluation de SCED est définie en terme de pourcentage par rapport au temps nominal calculé pour le projet. Les développements accélérés tendent à produire plus d'effort dans les phases de conception afin d'éliminer les risques et de raffiner l'architecture et plus d'effort dans les phases finales afin d'accomplir plus d'essai et de produire la documentation en parallèle.

SCED est le seul facteur de coût qui affecte le projet en entier. Les autres facteurs de coût peuvent être utiliser pour décrire chaque module à part dans un projet qui comporte plusieurs modules.

La figure E.3 montre les critères d'évaluation des 17 facteurs du modèle post-architecture.

E.4 Les facteurs de coûts du modèle précoce

Le modèle précoce définit 7 facteurs de coûts contre 17 pour le modèle post-architecture à cause de la vision supposée un peu limitée de toutes les caractéristiques du projet à ce stade tôt du cycle de développement.

Ces facteurs de coûts sont utilisés pour ajuster l'effort nominal et refléter l'impact des attributs pouvant influencer le développement.

L'évaluation des facteurs de coût du modèle précoce est obtenue à partir de l'évaluation des facteurs de coût du modèle post-architecture. Le tableau E.12 montre la correspondance entre les facteurs des deux modèles.

ANNEXE E. LES PARAMÈTRES DE LA MÉTHODE D'ESTIMATION PROPOSÉE

Facteurs de coûts (CF)	Très bas		Bas		Nominal (Complexité de l'application et/ou du produit à réaliser)		Elevé		Très élevé		Extrêmement élevé	
	Insuffisant	Pertes facilement récupérables	Pertes facilement récupérables	Atributs du produit (Qualité de service du centre de calcul et de l'ordinateur cible)	Pertes modérément récupérables	Pertes financièrement importantes	Conséquences sur les vies humaines	au niveau projet	au niveau programme	au niveau ligne de produit	Conséquences sur les vies humaines	au niveau de plusieurs lignes de produit.
RELY												
Contraintes de fiabilité du logiciel												
DATA												
Taille de la base de donnée												
DOCU												
Besoins de documentation couverts												
GPLX												
complexité de l'application												
RUSE												
Réutilisation composants de l'application												
Voire Fig. 4.3												
TIME												
Contraintes sur la durée d'exécution												
STOR												
Contraintes d'occupation mémoire												
PVOL												
Stabilité de la plate-forme d'exploitation												
Atributs du personnel (Expérience et/ou maturité des programmeurs, individuellement et collectivement)												
ACAP												
Maturité des architectes												
PCAP												
Maturité des programmeurs												
PCON												
Turnover du personnel												
APEX												
Expérience des architectes du domaine applicatif												
LTEX												
Pratique des langages et outils de programmation utilisés												
PLEX												
Expérience d'utilisation de la plate-forme d'exploitation												
Atributs du projet (Processus et/ou méthodes de développement adoptés par le projet)												
TOOL												
Utilisation d'outils logiciels												
SITE												
Développement sur plusieurs sites différents												
SCED												
Contraintes de planification												

Figure E.3 – Les facteurs de coûts

Les facteurs de coûts du modèle précoce	Les facteurs de coûts du modèle post-architecture
RCPX	RELY, DATA, CPLX, DOCU
RUSE	RUSE
PDIF	TIME, STOR, PVOL
PERS	ACAP, PCAP, PCON
PREX	APEX, PLEX, LTEX
FCIL	TOOL, SITE
SCED	SCED

Tableau E.12 – La correspondance entre les facteurs de coûts des modèles précoce et post-architecture.

L'évaluation des facteurs de coûts du modèle précoce est faite en suivant le principe suivant :

- 1) On évalue les facteurs de coûts du modèle post-architecture.
- 2) À chaque facteur de coûts du modèle post-architecture, on fait correspondre un poids comme suit : 1 pour très bas, 2 pour bas, 3 pour nominal, 4 pour élevé, 5 pour très élevé et 6 pour extrêmement élevé.
- 3) Selon le tableau relatif à chaque facteur de coût du modèle précoce, on détermine son évaluation en se basant sur le poids obtenu des facteurs de coûts du modèle post-architecture correspondant.

Le tableau E.4 montre les critères d'évaluation des facteurs de coûts du modèle précoce.

La fiabilité et la complexité du produit (RCPX)

RCPX regroupe les facteurs : les contraintes de fiabilité du projet (RELY), taille de la base de données (DATA), la complexité de l'application (CPLX) et les besoins de documentation (DOCU) du modèle post-architecture.

Réutilisation de composants de l'application (RUSE)

RUSE correspond au facteur portant le même nom du modèle post-architecture.

La difficulté de la plate-forme (PDIF)

RCPX regroupe les facteurs : les contraintes de la durée d'exécution (TIME), les contraintes d'occupation mémoire (STOR) et la stabilité de la plate-forme d'exploitation (PVOL) du modèle post-architecture.

Les compétences du personnel (PERS)

PERS regroupe les facteurs : la maturité des architectes (ACAP), la maturité des programmeurs (PCAP) et le turnover du personnel (PCON) du modèle post-architecture.

L'expérience du personnel (PREX)

PREX regroupe les facteurs : expérience des architectes du domaine applicatif (APEX), expérience d'utilisation de la plate-forme d'exploitation (PLEX) et pratique des langages et outils de programmation utilisés (LTEX) du modèle post-architecture.

L'équipement (FCIL)

FCIL regroupe les facteurs : utilisation d'outils logiciels (TOOL) et développement sur plusieurs sites différents (SITE) du modèle post-architecture.

Les contraintes de planification (SCED)

SCED correspond au facteur portant le même nom du modèle post-architecture.

E.4. LES FACTEURS DE COÛTS DU MODÈLE PRÉCOCE

Les facteurs du modèle précoce		Les facteurs du modèle post-architecture		Extrêmement bas	Très bas	Bas	Nominal	Elevé	Très élevé	Extrêmement élevé
La fiabilité et la complexité du produit	RCPX	Somme des poids de RELY, DATA, CPLX, DOCU	5,6	7,8	9-11	12	13-15	16-18	19-21	
	RUSE									
Réutilisation de composants de l'application		Le poids de RUSE	N/A	N/A	2	3	4	5	6	
La difficulté de la plate-forme	PDIF	Somme des poids de TIME, STOR et PVOL	N/A	N/A	8	9	10-12	13-15	16,17	
	PERS									
Les compétences du personnel		Somme des poids de ACAP, PCAP et PCON	3,4	5,6	7,8	9	10,11	12,13	14,15	
L'expérience du personnel		Somme des poids de APEX, PLEX et LTEX	3,4	5,6	7,8	9	10,11	12,13	14,15	
FCIL L'équipement		Somme des poids de TOOL et SITE	2	3	4,5	6	7,8	9,10	11	
SCED Les contraintes de planification		Le poids de SCED	N/A	1	2	3	4	5	6	

Figure E.4 – L'évaluation des facteurs de coût du modèle précoce

Liste des figures

2.1	Les différentes équipes intervenant dans la gestion des processus	14
2.2	Cycle de vie d'un processus	15
2.3	Les objectifs opérationnels de l'entreprise	20
3.1	Entreprise "Boîte Noire"	27
3.2	Entreprise "Boîte Blanche"	28
3.3	Système d'Information	28
3.4	Une approche structurelle d'un système d'information	30
3.5	La première partie du Framework de Zachman	35
3.6	Les relations d'une cellule de Zachman à travers une même ligne	36
3.7	Les huit composants du Federal Enterprise Architecture Framework	36
3.8	Les vues opérationnelle, système et technique du DODAF	37
4.1	Les éléments d'une Méthode de Modélisation d'Entreprise	40
4.2	Un méta-modèle de processus [76]	41
4.3	Un système hiérarchique	43
4.4	Un système multi-couches	44

4.5	Description d'un processus avec IDEF0	47
4.6	Un exemple de grille GRAI	50
4.7	Un métamodèle des processus réalisé par le WfMC	52
4.8	Structure du standard RosettaNet	53
4.9	Comparaison des langages de modélisation en terme de couverture des différentes vues [76]	55
4.10	L'utilisation possible de chaque langage de modélisation [76]	55
5.1	Les catégories des fonctionnalités des standards relatifs au BPM.	58
5.2	Un exemple de processus décrit en BPMN	60
5.3	Un exemple de processus décrit en BPEL	61
5.4	Architecture d'une solution BPMS	64
6.1	La relation entre le coût et la qualité	72
6.2	Les caractéristiques du produit logiciel selon ISO 9126	74
6.3	Un processus	75
6.4	L'approche processus	76
6.5	Les 3 axes d'amélioration des processus	76
6.6	La gestion de la maturité des processus	77
6.7	Les modèles et normes traitant l'amélioration des processus	78
6.8	La courbe de Six Sigma	80
6.9	La mise en œuvre de Six Sigma	81
6.10	La méthodologie DMAIC	82
6.11	La méthodologie DMADV	83
6.12	Le cadre de référence de COBIT	85
6.13	Guide d'audit de COBIT	86
6.14	Le framework d'ITIL	89

6.15	Les processus de PMBOK	91
6.16	Les 10 domaines de SWEBOK	93
6.17	Pyramide documentaire ISO 9001	95
6.18	Les chapitres de la norme ISO	96
6.19	Le contenu de la norme ISO 9001v2000.	97
6.20	PDCA : Plan, Do, Check, Act, "la roue de Deming".	98
6.21	Les étapes de PDCA.	99
6.22	Les anciens modèles à l'origine de CMMI.	100
6.23	Représentation étagée.	101
6.24	Les domaines de processus (PA) par niveau de maturité (ML)	103
6.25	CMMI - Structure du modèle et preuves.	104
6.26	Représentation continue.	105
6.27	Le modèle IDEAL.	107
6.28	SCAMPI Class C, Class B, Class A	110
6.29	Les phases d'évaluation SCAMPI A	111
7.1	Transformations de GRAI à UML	115
7.2	Les deux solutions retenues pour l'interopérabilité entre les méta-modèles de GRAI et UML	115
7.3	Le besoin d'interopérabilité entre les modèles de deux entreprises	116
7.4	L'interopérabilité à différents niveaux de modélisation	117
7.5	Le modèle de référence de MDI appliqué au cas d'étude	118
9.1	Les objectifs du référentiel qualité	128
9.2	Le champ de couverture des normes qualité des activités de l'entreprise	129
9.3	Le champ de couverture commun des normes qualité des activités de l'entreprise	130
9.4	L'approche multi-modèles proposée	130

9.5	Le diagramme de classe d'un standard qualité.	132
9.6	Le mapping entre 2 modèles	132
9.7	Le modèle intégré	133
9.8	Le modèle intégré spécifique	134
9.9	Le diagramme d'objet des standards qualité ISO 9001v2000 et CMMI	139
9.10	Vue macroscopique de l'activité d'une entreprise	141
9.11	Modèle simplifié d'un projet	142
9.12	Modèle simplifié d'un projet impliquant un partenaire	143
9.13	Solution d'interopérabilité entre une entreprise et ses clients	144
9.14	Processus "interface"	146
9.15	Le processus "PROCES-REAL-002 : VENDRE AVEC ENGAGEMENT DE RESULTATS"	147
9.16	Le processus "PROCES-SUPP-007 : GERER LES ENTENTES AVEC LES PARTENAIRES OFFSHORE"	148
9.17	La procédure "Analyser l'impact de la modification"	150
10.1	Modèle proposé	155
10.2	La répartition des charges totales par phase	160
10.3	La répartition des délais totaux par phase	161
10.4	Le processus d'estimation incluant le calibrage.	161
11.1	Répartition du chiffre d'affaire par métier client	168
11.2	Répartition des métiers de Sylis	169
11.3	La cartographie des processus de SDIE-Sylys	175
11.4	Le résultat du mapping d'ISO 9001v2000 vers CMMI	177
11.5	Le résultat du mapping de CMMI vers ISO 9001v2000	178
11.6	Le niveau d'intégration des domaines de processus de CMMI dans une cartographie de processus ISO 9001v2000	179

11.7	Le niveau d'intégration des domaines de processus de CMMI dans un processus	182
11.8	PROCES-REAL-001 Processus Vendre avec engagement de moyens.	184
11.9	Le référentiel documentaire implémenté au sein de Sylis	190
11.10	L'étape de recette du référentiel documentaire	191
11.11	Les étapes de la méthodologie d'audit projet	193
11.12	Le tableau de référence des compétences métier [39]	195
11.13	La compétence métier audit/qualité/méthode	196
11.14	Les différentes stratégies de conduite du changement	197
B.1	Les phases d'évaluation SCAMPI A	216
B.2	Les phases d'évaluation SCAMPI A (suite)	217
B.3	Organigramme de l'évaluation SCAMPI Class A	218
D.1	Tendance de l'évolution des coûts de développement vs les coûts d'achat de matériel dans un projet logiciel	262
D.2	Effets de la répartition d'une tâche sur plusieurs personnes.	264
D.3	Exemple où la relation Effort-taille n'est pas linéaire.	275
D.4	Répartition de l'effort selon la courbe de Rayleigh et la courbe classique.	276
D.5	Architecture d'un réseau de neurone à trois couches pour l'estimation de l'effort.	280
D.6	Arbre de régression pour la prédiction du temps de développement.	282
D.7	Arbre de régression construit à partir des 63 projets logiciels du COCOMO'81.	282
D.8	Résumé de l'algorithme génétique.	285
D.9	Exemple d'un Perceptron multi-couches et sa représentation avec un chromosome.	287
E.1	Les facteurs d'échelle	300
E.2	Détermination de la complexité de CPLX	303
E.3	Les facteurs de coûts	308
E.4	L'évaluation des facteurs de coût du modèle précoce	311

Liste des tableaux

6.1	Les apports de CMMI	106
9.1	Comparaison haut niveau d'ISO 9001v2000 et CMMI	135
9.2	Traduction des terminologies	138
10.1	Tableau de conversion des FP en SLOC.	157
11.1	Les chiffres clés de Syllis en 2007	168
A.1	Explication des Process Areas de CMMI (ML 2 et 3)	212
A.2	Explication des Process Areas de CMMI (ML 4 et 5)	213
C.1	Mapping ISO 9001v2000 vers CMMI	231
C.2	Mapping CMMI pratiques spécifiques vers ISO 9001v2000	259
E.1	Tableau de présentation des RET, DET et FTR.	294
E.2	Tableau d'évaluation de la complexité des ILF.	295
E.3	Tableau de conversion des ILF en FP.	295
E.4	Tableau d'évaluation de la complexité des EIF.	296

E.5	Tableau de conversion des EIF en FP.	296
E.6	Tableau d'évaluation de la complexité des EI.	296
E.7	Tableau de conversion des EI en FP.	297
E.8	Tableau d'évaluation de la complexité des EO.	297
E.9	Tableau de conversion des EO en FP.	297
E.10	Tableau d'évaluation de la complexité des EQ.	298
E.11	Tableau de conversion des EQ en FP.	298
E.12	La correspondance entre les facteurs de coûts des modèles précoce et post-architecture.	309

Bibliographie

- [1] *Federal Enterprise Architecture Framework (FEAF) Version 1.1*. Federal CIO Council, 1999.
- [2] *Treasury Enterprise Architecture Framework (TEAF) Version 1*. Department of the Treasury Chief Information Officer Council, 2000.
- [3] *Enterprise Architecture Development Tool-Kit, V 2.2*. National Association of State Chief Information Officers (NASCIO), 2002.
- [4] *The Open Group Architecture Framework (TOGAF) Version 8.1.1, Enterprise Edition*. The Open Group, 2007.
- [5] Georges AbouHarb and François Rivard. *L'EAI au service de l'entreprise évolutive : Les échanges d'information, enjeu pour les managers*. Maxima, Août 2003.
- [6] France AFAQ AFNOR. *NF EN ISO 9001 Décembre 2000 : Systèmes de management de la qualité - Exigences*. AFAQ AFNOR, 2000.
- [7] AFAI Association Française de l'Audit et du conseil Informatique. *COBIT : Gouvernance, contrôle et audit de l'information et des technologies associées*. Association Française de l'Audit et du conseil Informatique, 2002.
- [8] Dogac Asuman. *Workflow Management Systems and Interoperability*. Springer, 1998.
- [9] Richard Basque. *CMMI un itinéraire fléché vers le Capability Maturity Model Integration*. Dunod, 2004.
- [10] Richard Basque. *CMMI : Un itinéraire fléché vers le Capability Maturity Model Integration Version 1.2*. Dunod, 2006.

- [11] Maria Bergholtz, Paul Johannesson, and Petia Wohed. *UEML : Providing Requirements and extensions for interoperability challenges*. Department of computer and systems sciences Stockholm university et le centre de recherché en automatique de nancy université de Henri Poincaré, 2005.
- [12] Peter Bernus, Laszlo Nemes, and Theodore Joseph Williams. *Architectures for Enterprise Integration*. Springer, 1996.
- [13] BOC. *ADONIS*. <http://www.boc-group.com/>.
- [14] Barry Boehm. *Software Engineering Economics*. Prentice-Hall, 1981.
- [15] Barry Boehm, Chris Abts, Winsor Brown, Sunita Chulani, Bradford Clark, Ellis Horowitz, Ray Madachy, Donald J. Reifer, and Bert Steece. *Software Cost Estimation With COCOMO II*. Prentice-Hall, 2000.
- [16] Daniel Boeri. *Maîtriser la qualité : Tout sur la certification et la qualité*. Maxima, 2003.
- [17] Jean-Pierre Bourey, Reyes Grangel, Guy Doumeingts, and Arne Berre. *Deliverable DTG2.1 : REPORT ON MODEL ESTABLISHMENT*. INTEROP, Décembre 2005.
- [18] Jean-Pierre Bourey, Reyes Grangel, Guy Doumeingts, and Arne Berre. *Deliverable DTG2.2 : REPORT ON MODEL ESTABLISHMENT*. INTEROP, Juin 2006.
- [19] Jean-Pierre Bourey, Reyes Grangel, Guy Doumeingts, and Arne Berre. *Deliverable DTG2.3 : REPORT ON MODEL DRIVEN INTEROPERABILITY*. INTEROP, Mai 2007.
- [20] BPMS.info. *Le portail dédié au management par les processus*. <http://www.bpms.info/>.
- [21] C. Braesch. Le modèle olympios. *Actes de l'école de modélisation d'entreprise, Ecole des Mines d'Albi-Carmaux*, 2002.
- [22] G.W. Brams. *Réseaux de Petri : théorie et pratique*. Masson, 1983.
- [23] BPMI Business Process Management Initiative. Business process modeling language (bpml). Novembre 2002.
- [24] BPMI Business Process Management Initiative. Business process modeling notation (bpmn), specification. Mai 2004.
- [25] Casewise. *Casewise Corporate Modeler*. <http://www.casewise.com/FR/>.
- [26] Thierry Chamfrault and Claude Durand. *ITIL et la gestion des services : Méthodes, mise en œuvre et bonnes pratiques*. Dunod, 2006.
- [27] Shi Kuo Chang. *Handbook of Software Engineering and Knowledge Engineering : Fundamentals*. World Scientific Publishing Company, 2002.
- [28] Mary Beth Chrissis, Mike Konrad, and Sandy Shrum. *CMMI : Guidelines for Process Integration And Product Improvement*. Addison-Wesley Professional, 2006.

-
- [29] Tanguy Crusson. Business process management : De la modélisation à l'exécution, positionnement par rapport aux architectures orientées services. *Intalio*, 2003.
- [30] B. Curtis, M.I. Kellner, and J. Over. *Process modeling*. Communication ACM, 1992.
- [31] Berthold Daum and Udo Merten. *System Architecture with XML*. Morgan Kaufmann Publishers, 2002.
- [32] Bernard Debauche and Patrick Mégard. *BPM Business Process Management : Pilotage métier de l'entreprise*. Hermes Science Publications, 2004.
- [33] Michel Diaz. *Les réseaux de Petri : modèles fondamentaux*. Hermes Science, 2001.
- [34] DIN. *Systèmes de management de la qualité ; Exigences ; (ISO 9001 :2000), Version trilingue EN ISO 9001 :2000*. DIN, 2000.
- [35] DODAF DoD Architecture Framework Version 1.5. http://www.defenselink.mil/cio-nii/docs/dodaf_volume_i.pdf.
- [36] G. Doumeingts. *La méthode GRAI*. Thèse d'état, Université Bordeaux I, 1984.
- [37] Christian Dumont. *ITIL : Pour un service informatique optimal*. Eyrolles, 2007.
- [38] Peter Dunkhorst. *ISO 9001 :2000*. Behrs Verlag, 2003.
- [39] EMSI École de Management des Systèmes d'Information de Grenoble. *Référentiel de compétences pour les métiers des systèmes d'information*. Transactions on Software Engineering, vol. 4, no. 4, 2007.
- [40] Hans-Erik Eriksson and Magnus Penker. *Business Modeling with UML : Business Patterns at Work*. John Wiley & Sons, 2000.
- [41] Christophe Espern. *Interopérabilité : l'Arlésienne du DRM*. La Direction du Développement des Médias, Janvier 2004.
- [42] Norman Fenton and S.L. Pfleeger. *Software Metrics : A Rigorous and Practical Approach*. International Computer, Thomson Press, 1997.
- [43] Anis Ferchichi, Michel Bigand, and Hervé Lefebvre. *An ontology for quality standards integration in software collaborative projects*. Model Driven Interoperability for Sustainable Information Systems 2008 (MDISIS'08), 2008.
- [44] Anis Ferchichi, JeanPierre Bourey, Michel Bigand, and Hervé Lefebvre. *Implementing integration of quality standards CMMI and ISO 9001v2000 for software engineering*. International Conference on Concurrent Engineering 2007 (CE'2007), Brésil, 2007.
- [45] Anis Ferchichi, JeanPierre Bourey, Michel Bigand, and Hervé Lefebvre. *Mise en place d'un référentiel d'intégration multi-vues des processus d'entreprise*. 2èmes Journées Doctorales du GDR MACS, JDMACS'07, 2007.

- [46] Anis Ferchichi, JeanPierre Bourey, Michel Bigand, and Hervé Lefebvre. *Implementing a quality reference and processes interoperability in software collaborative projects*. Journal INSIGHT, The International Council on Systems Engineering (INCOSE), 2008.
- [47] Anis Ferchichi, JeanPierre Bourey, Michel Bigand, and Hervé Lefebvre. *Implementing integration of quality standards Capability Maturity Model Integration and ISO 9001v2000 for software engineering*. International Journal of Product Lifecycle Management (IJ-PLM), 2008.
- [48] Anis Ferchichi, JeanPierre Bourey, Michel Bigand, and Barron Marc. *Design systems engineering of software products : implementation of a software estimation model*. Computational Engineering in Systems Applications 2006 (CESA'2006), Pékin, 2006.
- [49] Claude Foulard. *La modélisation en entreprise : CIM-OSA et ingénierie simultanée*. Hermes, 1994.
- [50] Thornton Gale and James Eldred. *Getting Results with the Object-Oriented Enterprise Model*. SIGS Books, Janvier 1996.
- [51] Patrick Gannon. *OASIS Interoperability opportunities and obstacles*. Présentation lors du sommet de l'interopérabilité, 2001.
- [52] Ismael Ghalimi and David McGoveran. *Standards and BPM*. Business Integration Journal - Business Process Management Journal, 2004.
- [53] Diane L. Gibson, Dennis R. Goldenson, and Keith Kost. *Performance Results of CMMI-Based Process Improvement*. Software Engineering Institute (SEI), Août 2006.
- [54] Jean-Noël Gillot. *La gestion des processus métiers : Aligner les objectifs stratégiques de l'entreprise et le système d'information par les processus*. Lulu, 2008.
- [55] Paul Hagan. *Foundational Practices and tools for EA development (EA = Enterprise Architecture)*. EABOK, 2002.
- [56] Michael Havey. *Essential Business Process Modeling*. O'Reilly, 2006.
- [57] France IDS Scheer. *ARIS Platform*. http://www.ids-scheer.fr/fr/index_fr.html.
- [58] IEEE Institute of Electrical Electronics Engineers. *Guide to the Software Engineering Body of Knowledge*. IEEE, 2004.
- [59] Intalio. *Intalio/BPMS*. <http://www.intalio.com/>.
- [60] INTEROP Interoperability Research for Networked Enterprise Applications Software. *Project presentation*. INTEROP, Septembre 2005.
- [61] M. A. Jackson and Graham Twaddle. *Business Process Implementation Building Workflow Systems*. Addison Wesley, 1997.

-
- [62] Jim Johnson. *CHAOS Chronicles 1994*. Standish Group International, 1994.
- [63] Jim Johnson. *My Life Is Failure : 100 Things You Should Know to Be a Successful Project Leader*. Standish Group International, Août 2006.
- [64] Tim Kasse. *Action Focused Assessment for Software Process Improvement*. Software engineering, 2000.
- [65] Rashid N. Khan. *Business Process Management : a practical guide*. Meghan Kiffer Press, 2004.
- [66] Setrag Khoshafian and Marek Buckiewicz. *Introduction to Groupware, Workflow, and Workgroup Computing*. Wiley, 1995.
- [67] Tom Kimbrough and Linda Levine. *The IDEALSM Transition Framework - Speeding Managed Change*. Software Engineering Institute (SEI), 1997.
- [68] Mathieu Lauras, Hervé Pingaud, and Jacques Lamothe. *Méthodes de diagnostic et d'évaluation de performance pour la gestion de chaînes logistiques : application à la coopération maison-mère filiales internationales dans un groupe pharmaceutique et cosmétique*. Thèse préparée au Centre de Génie Industriel de l'Ecole des Mines d'Albi-Carmaux, 2004.
- [69] Jean-Louis Le Moigne. *La modélisation des systèmes complexes*. Dunod, 1990.
- [70] M. Lissandre. *Maîtriser SADT*. Editions Armand Colin, 1990.
- [71] E. Lutherer. *Méthodes et outils pour la modélisation de la productique*. 1996.
- [72] Bernard Manouvrier and Laurent Ménard. *Intégration applicative EAI, B2B, BPM et SOA*. Hermes Science Publications, Mars 2007.
- [73] David McGoveran. *An Introduction to BPM & BPMS*. Business Integration Journal - Business Process Management Journal, 2004.
- [74] Mega. *MEGA Process*. <http://www.mega.com/index.asp/1/fr>.
- [75] Microsoft. *Microsoft Biztalk server*. <http://www.microsoft.com/biztalk/default.mspx>.
- [76] Hafedh Mili, Guitta Bou Jaoude, Éric lefebvre, Guy Tremblay, and Alex Petrenko. Business process modeling languages : Sorting through the alphabet soup. *Rapport de recherche, D'épartement d'Informatique, UQAM*, Janvier 2004.
- [77] MITRE MITRE Corporation. Foundational practices and tools for ea development. *EABOK*, 2004.
- [78] Robert Moeller. *Sarbanes-Oxley Internal Controls : Effective Auditing With Cobit and Itil*. John Wiley & Sons, 2008.
- [79] Chantal Morley. *La modélisation des processus : typologie et proposition utilisant UML*. Processus et systèmes d'information, Journées ADELI, Paris, 2002.

- [80] Chantal Morley, Jean Hugues, Bernard Leblanc, and Olivier Hugues. *Processus Métiers et systèmes d'information : Evaluation, modélisation, mise en œuvre*. Dunod, 2004.
- [81] Boris Mutafelija and Harvey Stromberg. *Exploring CMMI-ISO 9001 :2000 Synergy when Developing a Process Improvement Strategy*. SEPG 2003 Conference et BearingPoint, 2003.
- [82] Yannik Naudet, Thibaud Latour, Kevin Hausmann, Sven Ables, Axel Hahn, and Paul Johannesonn. *The Ontology Web Language*. Henri Tudor Public Reseach center Luxembourg, Business Information systems university of Oldenburg Germany and Royal institute of technology KTH Stockholm Sweden, 2005.
- [83] OMG Object Management Group. *MDA Guide Version 1.0.1*. OMG, 2003.
- [84] OMG Object Management Group and BPMI Business Process Management Initiative. Business process modeling notation (bpmn), version 1.0. Février 2006.
- [85] OMG Object Management Group and BPMI Business Process Management Initiative. Request for proposals for version 2.0 of bpmn. Juin 2007.
- [86] M.A. Ould. *Business Processes : Modelling and Analysis for re-engineering and improvement*. John Wiley & Sons, 1995.
- [87] Maurice Pillet. *Six Sigma : Comment l'appliquer*. Editions d'Organisation, 2003.
- [88] PMI Project Management Institute. *A Guide to the Project Management Body of Knowledge (Pmbok Guide) Third Edition*. Project Management Institute, 2004.
- [89] L. H. Putnam. *A General Empirical Solution to the Macro Software Sizing and Estimation Problem*. Transactions on Software Engineering, vol. 4, no. 4, 1978.
- [90] Thomas Pyzdek. *The Six Sigma Handbook : A Complete Guide for Green Belts, Black Belts, and Managers at All Levels*. McGraw-Hill Companies, 2003.
- [91] Robert Reix. *Systèmes d'information et Management des organisastions*. Vuibert, 2002.
- [92] M. Roboam. *La méthode GRAI : principes, outils, démarche et pratique*. Edition TE-KENA, 1993.
- [93] Mélissa Saadoun. *Technologies de l'information et management*. Hermes, 2000.
- [94] Ruth Sara Savén. Process modelling for enterprise integration : review and framework. *LiTH/IPE*, 2002.
- [95] Jaap Schekkerman. *How to Survive in the Jungle of Enterprise Architecture Framework : Creating or Choosing an Enterprise Architecture Framework*. Trafford Publishing, 2003.
- [96] Martin Shepperd and C. Schofield. *Estimating Software Project Effort Using Analogies*. Transactions on Software Engineering, vol. 23, no. 12, 1997.

-
- [97] Howard Smith and Peter Fingar. *Business Process Management : The Third Wave*. Meghan Kiffer Press, 2006.
- [98] SEI Software Engineering Institute. *The IDEALSM Model : A Practical Guide for Improvement*. Software Engineering Institute (SEI), 1997.
- [99] SEI Software Engineering Institute. *CMMI for Development, Version 1.2*. SEI, Août 2006.
- [100] H. Tardieu, A. Rochfeld, and C. Colett. *La méthode merise*. 2000.
- [101] ZIFA The Zachman Institute for Framework Advancement. <http://www.zifa.com/>.
- [102] Jean-Louis Tomas. *ERP et Progiciels intégrés : la mutation des systèmes d'information, 2e édition*. Dunod, Janvier 2000.
- [103] Ray Tricker. *ISO 9001 :2000 : The Quality Management Process*. Van Haren Pub, 2006.
- [104] B. Vallespir and G. Doumeings. *Modélisation d'entreprise : vers le système d'information*. Actes de l'école de modélisation d'entreprise, Ecole des Mines d'Alès, 2004.
- [105] Bruno Vallespir, Christian Braesch, Vincent Chapurlat, and Didier Crestani. *L'intégration en modélisation d'entreprise : les chemins d'UEML*. Conférence francophone de modélisation et simulation MOSIM'03, Avril 2003.
- [106] Wil van der Aalst and Kees Max van Hee. *Workflow Management Models, Methods, and Systems*. MIT Press, 2002.
- [107] Frank Vandenbroecke. *Combiner CMM et ISO 9001 - 2000 pour l'amélioration de processus ?* N-Tech Belgium, 2001.
- [108] François Vernadat and L. Hamaidi. *La modélisation en entreprise : Méthodes descriptives des processus opérationnels*. Economica, Paris, 1998.
- [109] John Cunningham Wood and Michael C. Wood. *W. Edwards Deming : Critical Evaluations in Business and Management*. Routledge, 2005.
- [110] J.A. Zachman, , and J.F. Sowa. *Extending and Formalizing the Framework for Information Systems Architecture*. IBM Systems Journal, 1992.

Titre : CONTRIBUTION À L'INTEGRATION DES PROCESSUS MÉTIER : APPLICATION À LA MISE EN PLACE D'UN RÉFÉRENTIEL QUALITÉ MULTI-VUES

Le croisement de concepts issus de la gestion des processus métier, des normes et standards qualité et de l'interopérabilité nous a permis de nous intéresser à l'organisation et l'intégration des processus métier d'entreprise, pour proposer une démarche de mise en place d'un référentiel qualité multi-vues.

Le but de notre travail est de montrer comment intégrer les processus métier d'une entreprise à l'aide d'un référentiel commun offrant différents points de vue. Cette démarche généralisable est appliquée à l'intégration de deux standards de qualité, ISO 9001v2000 et Capability Maturity Model Integration (CMMI), afin de générer un référentiel qualité multi-vues permettant une certification relative aux deux normes. Ce référentiel prend en compte les chapitres imposés par ISO et les recommandations de CMMI. Dans le cadre de l'implémentation du référentiel, nous nous sommes intéressés à la définition d'une méthodologie d'estimation des délais et charges des projets informatiques afin de rationaliser ce processus critique pour l'entreprise. La mise en place de ce référentiel qualité s'accompagne de la définition d'une démarche assurant l'interopérabilité des processus définis avec ceux des clients et/ou partenaires.

Une méthodologie d'audit projet, un référentiel documentaire et un référentiel des compétences viennent compléter le travail déjà réalisé afin d'assurer l'implémentation et le respect du référentiel qualité.

Mots clés : Processus métier, Référentiel qualité, ISO 9001v2000, Capability Maturity Model Integration (CMMI), Interopérabilité.

Title : CONTRIBUTION TO BUSINESS PROCESS INTEGRATION : APPLICATION TO IMPLEMENTING A MULTI-VIEWS QUALITY REFERENCE FRAME

The use of concepts borrowed from Business Process Management (BPM), quality standards and interoperability allows to be interested by the organization and intergration of enterprise business processes , bringing us to propose a step of implementing a multi-views quality reference.

The goal of our work is to present how to integrate several processes using a common reference frame offering various viewpoints. This generalizable step is applied to the integration of two quality standards, ISO 9001v2000 and Capability Maturity Model Integration (CMMI), in order to generate a multi-views quality reference frame allowing a certification relatively to the two standards. This reference frame takes into account chapters imposed by ISO and the recommendations of CMMI. During the implementation of our reference frame, we were interested to the definition of a cost and effort estimation methodology in order to rationalize this critical process for the enterprise. The implementation of the quality reference frame was accompanied by the definition of a step ensuring the interoperability of defined processes with those of customers and/or supplier.

An audit project methodology, a documentary reference frame and a competences reference frame were also developed in order to ensure the implementation and the integration of the quality reference frame.

Keywords : Business process, Quality reference frame, ISO 9001v2000, Capability Maturity Model Integration (CMMI), Interoperability.