



HAL
open science

L'Ingénierie des Exigences appliquée aux Systèmes d'Information

Camille Salinesi

► **To cite this version:**

Camille Salinesi. L'Ingénierie des Exigences appliquée aux Systèmes d'Information. Génie logiciel [cs.SE]. Université Panthéon-Sorbonne - Paris I, 2010. tel-01003149

HAL Id: tel-01003149

<https://theses.hal.science/tel-01003149>

Submitted on 10 Jun 2014

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.

Habilitation à Diriger des Recherches

Présentée à

L'Université Paris 1 Panthéon – Sorbonne

Pour obtenir

L'Habilitation à Diriger des Recherches en Sciences, Mention Informatique

par

Camille Salinesi

Docteur de L'Université Paris 6 Pierre et Mari Curie

27ème section CNU

**L'Ingénierie des Exigences appliquée aux Systèmes
d'Information**

Soutenue le 7 Décembre 2010 devant le jury composé de:

Corine Cauvet	Rapporteur
Neil Maiden	Rapporteur
Régine Laleau	Rapporteur
Oscar Pastor	Examineur
Jean Bernard Baillon	Examineur
Isabelle Comyn Wattiau	Examineur
Colette Rolland	Directeur

Remerciements

Les chercheurs, comme les navigateurs d'autrefois établissent des caps vers des destinations encore inexplorées, des mers aux contours mal définis, ou encore pour ouvrir de nouvelles routes. Au cours de ces dernières années, j'ai appris que comme l'exploration maritime d'autrefois, l'activité de recherche se pratique difficilement seul: mieux vaut travailler en équipage.

J'ai eu le privilège de m'enrichir en côtoyant un certain nombre de personnes. Dans mon équipage, et dans celui des autres navires que j'ai pu croiser.

Avant tout, je tiens à remercier le Professeur Colette Rolland, Directrice du Centre de Recherche en Informatique de L'Université Paris 1. Pendant près de quinze ans, Colette m'a donné le modèle exceptionnel d'un esprit brillant, clairvoyant, dynamique, tenace. Sa volonté de toujours ouvrir de nouvelles voies sans jamais se contenter de ce qui est tenu pour acquis est -et restera pendant de nombreuses années- source d'inspiration. Outre une formation de chercheur, Colette m'a offert un modèle qui m'a profondément marqué; j'ai bien peu à apporter en retour que ma profonde gratitude. Colette, je dois aussi vous remercier de m'avoir pris à votre bord. J'espère avoir été un équipier digne de la confiance que vous m'avez accordée.

Mes périples de doctorant puis de Maître de Conférences ont été l'occasion de croiser la route de nombreux chercheurs du monde entier: Europe, Afrique du Nord, Scandinavie, Moyen Orient, Asie, Amérique du Nord, Australie. La fréquentation de personnalités créatives, intelligentes, et le plus souvent très agréables a été un privilège. Parmi les très nombreux noms, me viennent à l'esprit ceux de Naveen Prakash, Neil Maiden, Klaus Pohl, Patrick Heymans, Eric Dubois, Michel Leonard, Jolita Ralyté, Anne Persson, Bjorn Regnell avec qui j'ai eu l'occasion de travailler en étroite collaboration et qui, plus que des collègues, sont devenus des amis. Je ne pourrai citer exhaustivement le nom de tous ceux que j'admire et que j'ai beaucoup de plaisir à croiser régulièrement, car ils sont trop nombreux. Je remercie cependant tous ces collègues proches ou moins proches qui ont été source d'échanges, d'inspiration, de plaisir, et dont le modèle me donne sans cesse le goût d'explorer de nouveaux territoires.

Le Centre de Recherche en Informatique a été un lieu de rencontre et de collaboration avec de nombreux autres équipiers: Carine Souveyet, Selmin Nurcan, Rebecca Deneckere, Daniel Diaz, Said Assar, Françoise Gire, Georges Grosz, Corine Cauvet, et les très nombreux thésards (jeunes marins?) que j'ai pu y fréquenter. J'ai pris beaucoup de plaisir à échanger des idées, explorer des controverses, approfondir des résultats, en un mot produire avec vous les efforts nécessaires à l'activité de recherche. C'est donc avec beaucoup de plaisir que je vous remercie pour la générosité dont vous faites preuve dans le partage des efforts pour border les voiles.

Depuis quelques années, j'ai aussi eu l'occasion de tirer des bords avec quelques doctorants. Anne Etien, Iyad Zoukar, Emmanuel Papadacci, Ines Gam, Elena Ivankina, Laure Hélène Thévenet, Olfa Djebbi, Elena

Kornyshova, et Raul Mazo Peña ont dû subir les aléas de mes interrogations et changements de cap. Il est important que chacun comprenne combien j'ai appris à leur contact. Je serais heureux de leur avoir apporté ne serait-ce qu'une petite portion de ce que j'ai pu apprendre pendant les bords que nous avons tirés ensemble avec Colette.

Merci aux collègues administratifs et aux autres collègues chercheurs auprès de qui je continue d'avancer et de compléter mon expérience.

Enfin, quand on quitte le rivage on y laisse une famille et des amis. Je ne pourrai jamais suffisamment les remercier, les uns pour l'amitié, les autres pour l'amour dont ils font preuve. Vous êtes pour moi une source inépuisable de quiétude intérieure. C'est toujours vers vous que se tourne mon cœur lorsque mon regard se porte vers l'horizon et que j'y vois que le vent force et que les creux se font plus profonds. Vous êtes ma détermination.

Tassinca, Août 2008.

Avant propos

Mon métier est celui d'enseignant chercheur. Le rôle de l'enseignant chercheur couvre plusieurs activités dont trois principales: la recherche, l'enseignement et la valorisation.

Dès 2000, année de mon entrée en fonction au poste de Maître de Conférences, mes efforts se sont principalement tournés vers la recherche. En effet, il m'est très rapidement apparu que l'enseignement supérieur d'une part, et l'innovation d'autre part, doivent être constamment alimentés par la recherche pour être de qualité.

Il me semble que le travail de recherche consiste pour l'essentiel à développer de nouvelles théories, les explorer, et les valider. L'objectif de valorisation amène à la mise en pratique des théories développées. Pour moi, l'effort de mise en pratique s'est, de manière assez systématique, concrétisé par le développement de méthodes, de techniques ou d'outils destinés à être utilisés dans l'industrie. Enseigner les principes de ces théories nouvelles, et des méthodes, techniques et outils qui en découlent apparaît comme une manière indirecte de valoriser les résultats de recherche auprès des entreprises et par le biais des étudiants. J'ai par ailleurs cherché à tisser des liens avec le monde industriel. Cela a non seulement été l'occasion de réaliser directement des transferts de technologie par la participation à des projets et au travers d'activités de conseil, mais aussi d'observer les réalités de terrain. Cette activité a en particulier été utile à l'identification de besoins concrets, à la définition de problématiques nouvelles et encore mal résolues, et à l'établissement de théories originales.

Je crois qu'il est aujourd'hui devenu extrêmement difficile -voir quasi-impossible- de progresser de manière isolée. Se tenir en permanence informé des publications les plus récentes est donc tout aussi important que publier. Le travail de collecte et d'approfondissement des connaissances que suppose le métier de chercheur s'avère, de manière concomitante, indispensable pour enseigner au niveau Universitaire. Tout en effectuant mes recherches, je me suis attaché à satisfaire les besoins d'enseignement. Particulièrement dans mon établissement, mais aussi dans d'autres établissements d'enseignement supérieur, grandes écoles et universités. Ceci m'a conduit à intervenir auprès de publics étudiants de différents niveaux (Licence, Master), et issus de différents domaines (gestion, mathématiques et sciences sociales, informatique). J'ai pu m'apercevoir que la "théorisation" -le développement de théories et leur confrontation aux phénomènes observés- n'était pas qu'un moyen de mieux comprendre les phénomènes du monde réel et de développer des solutions générales utiles à la résolution de problèmes concrets donc de nouveaux contenus d'enseignement. Les théories nouvelles introduisent aussi de nouvelles manières de présenter les contenus. Elles sont aussi l'occasion d'améliorer la capacité de l'enseignant à s'adapter à des situations d'enseignement diversifiées. Enfin, j'ai eu aussi l'occasion de diriger des équipes pédagogiques, et de prendre part à des activités d'ingénierie pédagogique. Mon sentiment est que les qualités de rigueur et d'organisation développées en tant que chercheur sont utiles dans ces différentes missions relatives à la pédagogie. La connaissance experte du domaine est bien entendu une contribution essentielle.

Le triptyque recherche – enseignement - valorisation auprès de l'industrie est donc pour moi la base d'un cercle vertueux: la recherche produit des connaissances nouvelles transmises au monde de l'industrie par le biais des

étudiants formés par des chercheurs. Les étudiants participent à la valorisation lorsqu'ils partent dans l'industrie, mais aussi au retour d'expérience en appliquant les théories issues des laboratoires dans le contexte de leur activité industrielle. Enfin, les entreprises qui exploitent directement les résultats de recherche pour innover contribuent par un retour d'expérience direct à la validation des théories développées par les chercheurs.

La mise en place et le développement d'un triptyque recherche – enseignement - valorisation solide ne peut se réaliser que dans le temps. L'encadrement de stagiaires, la participation à des clubs de professionnels, et surtout la collaboration avec des industriels sont des activités qui selon moi contribuent à la consolidation de ce triptyque, et donc prennent une part importante dans mon métier d'enseignant chercheur.

La participation et le montage de projets de recherche en collaboration avec des collègues académiques et industriels est aussi bénéfique pour l'activité de recherche puisque c'est l'occasion de stimuler la production de travaux de haut niveau.

Enfin, la publication scientifique d'articles dans des conférences et revues sélectives, et par le biais d'ouvrages est une activité qui m'a paru importante pour confronter mes travaux aux exigences des standards internationaux.

Pour moi, la recherche est au cœur du triptyque recherche – enseignement - valorisation. J'ai donc choisi de présenter dans ce mémoire mes travaux de recherche en priorité. Le document démarre par une introduction à mon domaine de recherche. Chacun des chapitres qui suit présente l'une des thématiques que j'ai eu l'occasion de traiter dans le périmètre de mon domaine. Chaque chapitre commence par une brève introduction à la thématique et définit mon positionnement par rapport à la communauté des chercheurs intéressés à cette problématique. Dans chaque thématique, j'ai développé plusieurs programmes de recherche, chacun intéressé à une problématique spécifique. Chaque programme de recherche fait l'objet d'une section dans le chapitre. Chaque problématique est traitée en trois parties. Tout d'abord, j'introduis l'objet de la problématique. Ensuite, je définis mon positionnement en abordant les grands principes des théories que j'ai pu développer et ce qui les rend originales par rapport aux autres approches. Enfin, je présente les résultats des travaux que j'ai menés. A la fin de chaque chapitre, j'aborde mes perspectives concernant la thématique qui y est abordée.

Sommaire

REMERCIEMENTS.....	I
AVANT PROPOS.....	III
SOMMAIRE.....	V
CHAPITRE 1.....	1
INTRODUCTION GÉNÉRALE	2
1. <i>Domaine de la recherche: l'Ingénierie des Exigences.....</i>	<i>2</i>
2. <i>Organisation du mémoire.....</i>	<i>6</i>
CHAPITRE 2.....	9
IDENTIFICATION DES EXIGENCES POUR LE DÉVELOPPEMENT DE NOUVEAUX SYSTÈMES ("FROM SCRATCH")	10
1. <i>Introduction et positionnement de mes travaux sur l'identification des exigences.....</i>	<i>10</i>
2. <i>Positionnement.....</i>	<i>12</i>
2.1. <i>Problématique de la découverte des exigences par les scénarios.....</i>	<i>13</i>
2.2. <i>Principes de l'approche L'Ecritoire.....</i>	<i>15</i>
2.2.a <i>Organisation des buts et scénarios en une hiérarchie de fragments de besoins.....</i>	<i>15</i>
2.2.b <i>Couplage bi-directionnel buts-scénarios.....</i>	<i>15</i>
2.2.c <i>Modélisation linguistique et rédaction des scénarios</i>	<i>16</i>
2.2.d <i>Découverte des buts par l'analyse des scénarios</i>	<i>18</i>
2.3. <i>Résultats</i>	<i>18</i>
3. <i>Complétude des exigences par l'analyse des menaces - l'approche RITA.....</i>	<i>23</i>
3.1. <i>Problématique de la complétude.....</i>	<i>23</i>
3.2. <i>Principes de l'approche RITA.....</i>	<i>24</i>
2.3.a <i>Modélisation causale des menaces</i>	<i>25</i>
2.3.b <i>Ontologies des causes de menaces et des stratégies de traitement.....</i>	<i>25</i>
3.3. <i>Résultats</i>	<i>26</i>
4. <i>Priorisation de grands ensembles d'exigences structurés par les buts - l'approche NENO.....</i>	<i>27</i>
4.1. <i>Problématique de la priorisation de grands ensembles d'exigences</i>	<i>29</i>
4.2. <i>Principes de l'approche NENO.....</i>	<i>31</i>
2.4.a <i>Définition des critères d'arbitrage au moyen de NFRs</i>	<i>32</i>
2.4.b <i>Clustering de la collection d'exigences en scénarios d'évolution alternatifs au moyen du modèle MAP.....</i>	<i>33</i>

2.4.c	Etude quantitative des impacts sur l'atteinte d'objectifs sous la forme d'agrégation d'évaluations	33
2.4.d	Calcul des priorités par l'évaluation qualitative, participative, et quantitative des scénarios d'évolution..	34
2.4.e	Représentation multi-formes des priorités	34
4.3.	Résultats	34
5.	<i>Perspectives de recherche</i>	37
CHAPITRE 3		39
DÉCOUVERTE DES EXIGENCES PAR MISE EN COÏNCIDENCE ("MATCHING")		40
1.	<i>Introduction et positionnement de mes travaux sur la mise en coïncidence</i>	40
2.	<i>Positionnement</i>	42
3.	<i>Mise en coïncidence par analyse de similarités – l'approche MIBE</i>	44
3.1.	Problématique dans l'élaboration d'une typologie de similarités	45
3.2.	Principes de l'approche MIBE	47
2.3.a	Typologie générique des similarités	48
2.3.b	Catalogue de métriques génériques et adaptables.....	49
3.3.	Résultats	52
4.	<i>Mise en coïncidence par résolution de contraintes – l'approche REDPL</i>	56
4.1.	Problématiques résolues par la mise en coïncidence par résolution de contraintes	57
4.2.	Principes de l'approche REDPL	59
2.4.a	Spécification de la variabilité au moyen d'un programme de contraintes.....	59
2.4.b	Mise en coïncidence au moyen d'un solveur de contraintes.....	60
2.4.c	Identification des exigences de configuration par le calcul d'optimum	61
4.3.	Résultats	61
5.	<i>Perspectives de recherche</i>	65
CHAPITRE 4		67
INGÉNIERIE DES EXIGENCES DANS LE CONTEXTE DE L'ÉVOLUTION		68
1.	<i>Introduction et positionnement de mes travaux sur le contexte de l'évolution</i>	68
2.	<i>Positionnement</i>	69
3.	<i>Spécification des exigences d'évolution par écarts – l'approche GAP</i>	69
3.1.	Problématique de la spécification d'exigences d'évolution.....	71
3.2.	Principes de l'approche GAP	72
2.3.a	Spécification des exigences d'évolution au moyen d'un langage adapté d'une typologie générique d'écarts 72	
2.3.b	Définition d'une typologie générique d'écarts au moyen d'un méta modèle générique	73
2.3.c	Définition formelle des opérateurs générique au moyen d'invariants	75
2.3.d	Démarche de définition d'un langage de spécification des exigences de changement par adaptation de la typologie générique d'écart au méta modèle spécifique	75
3.3.	Résultats	75

4. <i>Co-évolution</i>	77
4.1. Problématique de la co-évolution	79
4.2. Principes du cadre de travail de la coévolution.....	80
2.4.a Comprendre la coévolution par les relations des entités qui co-évoluent et l'expression des exigences d'évolution	80
2.4.b Mettre en œuvre la coévolution par: l'identification des exigences d'évolution, la propagation du changement, et la vérification des relations entre les entités cibles.....	80
4.3. Résultats	83
5. <i>Perspectives de recherche</i>	84
CHAPITRE 5	87
ALIGNEMENT	88
1. <i>Introduction et positionnement de mes travaux sur l'alignement</i>	88
2. <i>Positionnement</i>	88
3. <i>Modélisation et mesure de l'alignement – l'approche ACEM</i>	90
3.1. Problématique liée à la modélisation et à la mesure de l'alignement.....	91
3.2. Principes de l'approche ACEM.....	93
2.3.a Modélisation de la relation d'alignement au moyen de liens.....	93
2.3.b Mesure de l'alignement système-processus au moyen d'une typologie générique de métriques.....	94
2.3.c Adaptation de la typologie générique de métriques par instanciation de métra modèles génériques	94
2.3.d Découverte des exigences d'évolution par la modélisation et la mesure de l'alignement au moyen d'un modèle pivot	96
3.3. Résultats	98
4. <i>Modélisation de l'alignement stratégique – l'approche INSTAL</i>	100
4.1. Problématiques liées à la modélisation de l'alignement stratégique.....	101
4.2. Principes de l'approche INSTAL.....	103
2.4.a Spécification de l'exigence d'alignement stratégique au moyen d'un formalisme intentionnel.....	104
2.4.b Définition de liens structurellement complexes à partir du modèle intentionnel	105
2.4.c Définition du rôle joué par les entités alignées	106
2.4.d Mesure du degré effectif d'alignement au moyen de métriques	107
2.4.e Découverte d'exigences d'évolution par modélisation et mesure des exigences d'alignement stratégique	108
4.3. Résultats	110
5. <i>Perspectives de recherche</i>	112
CHAPITRE 6	115
CONCLUSIONS GÉNÉRALES.....	116
ANNEXES	117
RÉFÉRENCES.....	118

BIBLIOGRAPHIE	135
1. Métriques bibliographiques.....	135
2. Principales références	135
3. Chapitres de livres.....	136
4. Articles de revue	137
5. Articles publiés dans des conférences internationales avec comité de lectures et actes.....	138
6. Articles publiés dans des conférences nationales avec comités de lecture et actes	141
7. Articles publiés lors d'ateliers et forums internationaux avec actes.....	142
8. Articles présentés dans le cadre d'ateliers nationaux avec ou sans actes.....	144
9. Rapports internes et livrables de projets.....	145

CHAPITRE 1

Introduction générale

Mon domaine de recherche est celui de l'*Ingénierie des Exigences*. L'ingénierie des exigences est un domaine de recherche encore relativement jeune: jusqu'à la fin des années 80 (et même encore aujourd'hui) on parlait encore d' "analyse" pour qualifier la phase en amont de la conception des systèmes. La phase d'analyse est indispensable pour produire les spécifications du système à développer.

Traditionnellement, les spécifications de systèmes apparaissent sous forme de modèles d'ingénierie qui s'attachent à décrire les fonctionnalités des systèmes, leur structure, et leur comportement. Par exemple, les Réseaux de Petri, utilisés de manière très intensive dans le monde du génie logiciel permettent de définir des comportements complexes de systèmes de manière formelle afin d'en démontrer certaines propriétés. Dans le monde des systèmes d'information, d'autres modèles de comportement tels que les diagrammes de flux de données ou les modèles de tâche sont employés pour spécifier comment les systèmes d'information opérationnalisent les tâches administratives des organisations dont ils doivent supporter le fonctionnement. Les modèles structurels (E/R, E/A, SADT) sont aussi très utilisés. Ils permettent de définir les structures de données des informations que ces systèmes devaient permettre de gérer. Fondamentalement la position inhérente à toutes ces approches est identique: les modèles objets, tout comme les modèles de comportement ou les modèles de données s'attachent à décrire ce que les systèmes doivent être, partant d'une compréhension de phénomènes du monde réel.

1. Domaine de la recherche: l'Ingénierie des Exigences

Les cycles de vie de développement logiciels en cascade [Royce1987] ou en V sont très représentatifs de la place prise par l'activité d'analyse dans le développement des spécifications de systèmes: il s'agissait de collecter les exigences en "amont" du projet pour immédiatement concevoir la solution coïncidente [Larman2003]. Très populaires chez les industriels cette approche a rapidement montré des limites importantes dont le fameux "effet tunnel"¹ qui est apparu comme l'un des principaux syndromes [Sommerville2007], [McCracken1982]. Des études ont été menées au vu des gouffres financiers générés par les projets inachevés, les dépassements de budgets, les défaillances de systèmes, et les situations critiques générées par le rejet des systèmes par les acquéreurs et par les utilisateurs. Le constat relaté en 1994 par le Standish Group dans ses fameux rapports « *Chaos* » et « Unfinished Voyages » est sans appel: l'une des principales causes de ces difficultés résulte directement ou indirectement d'une mauvaise prise en compte des exigences. Une hypothèse se généralise alors: plus les problèmes sont

¹ On parle alors de l' « effet tunnel»: plus une erreur apparaît tard dans un projet, plus il est coûteux de la corriger. Il est donc essentiel de s'assurer de la qualité des phases amont. L'effet tunnel est aggravé par un « effet de levier»: chaque erreur faite à une étape du projet entraîne de nouvelles erreurs dans les étapes ultérieures. Le nombre d'erreurs croît donc de manière quadratique. Comme l'effet tunnel, l'effet levier justifie de s'assurer de la qualité des phases amont des projets.

identifiés et traités en amont, moins il est coûteux de les résoudre². L'Ingénierie des Exigences est ancrée dans ce défi: réduire la fréquence et l'amplitude des problèmes de développement de systèmes en appliquant en amont des méthodes techniques et outils qui garantissent -par leur dimension systématique- une meilleure prise en compte des exigences dans le développement des systèmes.

Très rapidement, il apparaît que l'ingénierie des exigences ne peut être résumée à la simple production en début de projet d'un document spécifiant ce que sera le système à construire. Il s'agit en fait d'un ensemble complexe de processus multiples qui ont lieu tout au long du cycle de développement des systèmes. L'Ingénierie des Exigences fait intervenir de nombreuses parties prenantes dont les perceptions et modes d'expression diffèrent, et parfois divergent. Elle fait appel à la créativité -nécessaire au développement de systèmes innovants- aussi bien qu'à la rigueur de spécifications formelles -nécessaire à la création de systèmes dont les qualités sont rigoureusement démontrées. Le but est bien de produire des spécifications d'exigences complètes, cohérentes, et suffisamment précises pour permettre le développement des systèmes.

Une contribution essentielle de l'Ingénierie des Exigences à l'Ingénierie des Systèmes réside dans la transition de la question du QUOI à la question du POURQUOI. Lorsqu'un ingénieur s'intéresse à la question du quoi, il étudie la nature du système: ses constituants, son comportement, ou ses fonctions. La question du pourquoi amène à s'intéresser aux raisons qui rendent ces éléments nécessaires; il peut aussi bien s'agir de l'activité des utilisateurs que du rôle joué par le système dans un système plus complexe comme par exemple une entreprise ou un système de systèmes.

A la fin des années 80, et au début 90 alors que le "tout-objet" prédomine (le paradigme orienté-objet influence les méthodes d'analyse et de conception des systèmes d'information, les systèmes de gestion de base de données, aussi bien que les langages de programmation), plusieurs voix s'élèvent exprimant l'importance d'une représentation explicite du "pourquoi" en opposition au "quoi" et au "comment" [Jacobs1993]. Les raisons des limites pratiques des méthodes traditionnelles commencent à peine à être comprises. Fondamentalement, il s'avère que les concepteurs des systèmes qui emploient les méthodes traditionnelles répondent bien aux cahiers des charges. Cependant, les méthodes traditionnelles amènent les concepteurs à définir comment ces systèmes doivent être construits – explorant ainsi l'espace de la solution, sans avoir pu explorer pourquoi les systèmes développés sont nécessaires – explorant ainsi l'espace du problème. L'investissement est donc fait dans des développements qui ne sont pas correctement justifiés. Les expériences accumulées confirment que c'est bien le développement de "solutions" sans compréhension de la nature profonde des "problèmes" qui est à l'origine des risques d'échecs constatés.

Plus de trente ans auparavant, la question du pourquoi avait pourtant déjà été placée au cœur de la problématique de développement des systèmes [Ross1977]. Ainsi, Ross précise dès 1977 que « *la définition des exigences doit indiquer: pourquoi un système est requis dans les conditions actuelles ou futures, et quelles caractéristiques de systèmes permettent de satisfaire ce contexte* ». Cependant, il n'existe alors pas de mode de représentation

2 Bien que les chiffres produits par le Standish Group et la méthode employée pour les produire fassent l'objet de controverse, ses conclusions font bien l'objet d'un consensus [Jorgensen2006] [Glass2006].

du "pourquoi" qui permette d'en effectuer une représentation systématique. Tout comme les langages de modélisation conceptuelle permettent de spécifier le "quoi" des systèmes, de nouveaux langages sont nécessaires pour spécifier le "pourquoi". C'est le propos des langages de modélisation de buts.

En 1987, K. Yue propose de représenter explicitement les buts des utilisateurs de systèmes afin de les utiliser comme critère de pertinence et de complétude des modèles d'exigences [Yue1987]. Les buts ainsi décrits permettent de valider la complétude des exigences qui leurs sont associées: *« les exigences sont complètes si elles sont suffisantes pour établir les buts qu'elles affinent. »*

A la même époque, M. Feather propose de s'inspirer de CCS [Milner1980] (Calculus of Communication Systems) et de CSP [Hoare1978] (Communicating Sequential Processes) pour représenter des propriétés externes des systèmes [Feather1987]. Dans CCS, *« le comportement d'un programme est défini par son mode de communication avec un observateur (son environnement); des arbres non ordonnés avec des arcs labellisés représentent des comportements »*. CCS comme CSP proposent de construire les systèmes par composition de sous-systèmes qui communiquent. Le comportement de systèmes complexes est donc déterminé par celui des systèmes qui les composent. Le changement de perspective introduit par le langage Gist de M Feather est important. En effet, Gist permet de décrire le comportement de systèmes non plus sous l'angle de la composition de sous-systèmes dont les caractéristiques sont déjà connues, mais de manière "ouverte" grâce à des arbres de choix dans lesquelles les branches sont représentatives des contraintes issues de l'environnement que le système spécifié doit respecter. Les buts définis dans le modèle Gist de M. Feather ne sont pas nommés explicitement ni désignés comme tels. Ils apparaissent implicitement par le biais des choix à faire pour définir les exigences que les systèmes doivent respecter, et sont formalisés par les contraintes de type prédicatives. Cette aptitude caractéristique à supporter la définition des choix à réaliser se retrouve dans tous les modèles de buts qui ont par la suite été définis.

La fin des années 80 voit naître l'Ingénierie des Exigences en tant que discipline scientifique à part entière. A cette époque naissent deux langages de modélisation de buts qui deviennent des références dans le domaine de l'Ingénierie des Exigences: KAOS [Dardenne1993], et I* [Mylopoulos1992].

Le langage KAOS propose une approche formelle essentiellement consacrée à la construction de modèles d'exigences complets et sans conflit. KAOS s'intéresse à la définition de buts "satisfaisables", c'est à dire pour lesquels la question de la satisfaction ou non satisfaction du but est un problème décidable à tout instant t (on parle en anglais de *"goal satisfaction"*). Dans KAOS, les modèles de buts sont des arbres semblables aux arbres ET/OU de décomposition de buts définis en intelligence artificielle pour spécifier l'ensemble des chemins possibles dans la résolution d'un problème. Bien que l'inspiration soit différente, l'approche de décomposition proposée par KAOS comporte certaines similitudes avec celle introduite par Gist. Ainsi, KAOS exploite systématiquement la spécification de contraintes pour formaliser les arbres de buts. KAOS va cependant bien au-delà de Gist: dans KAOS, les buts sont définis nommément, et les contraintes, exprimées sous la forme de prédicats logique du premier ordre ou de la logique temporelle, permettent à la fois de formaliser les buts et d'évaluer leur complétude et leur cohérence.

L'approche préconisée par I* repose sur un cadre de travail de nature plus qualitative. Le langage de modélisation I* introduit en effet la modélisation de buts dont la satisfaction est inscrite dans le temps et ne peut donc être vue comme un événement discret (en anglais, on emploie l'expression "*goal satisficing*"). Les modèles de buts I* sont représentés sous la forme de graphes dans lesquels les arcs définissent des dépendances positives et négatives entre les buts et les agents responsables de leur satisfaction. Les modèles de buts spécifiés avec le langage I* peuvent être particulièrement utilisés pour explorer l'environnement de fonctionnement des systèmes développés, ainsi que pour évaluer des modèles d'exigences alternatifs.

Les principaux acteurs de l'Ingénierie des Exigences, aussi bien académiques qu'industriels, ont reconnu l'importance des buts en Ingénierie des Exigences. Ainsi, selon P. Zave « *l'Ingénierie des Exigences s'intéresse aux buts du monde réel qui contraignent les systèmes logiciels et à leurs liens avec les spécifications de ces systèmes* » [Zave1997]. Le concept de but apparaît dans standard IEEE 830 de documentation des exigences [IEEE1998]. De nombreux travaux de recherche du domaine de l'ingénierie des exigences ont exploité le concept de but [Lamsweerde2004]:

- pour modéliser les dépendances entre agents en termes de satisfaction de but, négocier les exigences, gérer les conflits, explorer les obstacles à la satisfaction des buts, et raisonner sur les conséquences d'une satisfaction partielle des buts;
- pour affiner les exigences, identifier les exigences opérationnelles, démontrer leur complétude;
- pour découvrir des buts; par exemple à partir de scénarios, ou par réutiliser au travers de taxonomies;
- pour modéliser et raisonner autour de types particuliers d'exigences, telles que les exigences de sécurité;
- pour identifier les dérives dans la mise en œuvre des buts par rapport aux buts initiaux;
- pour dériver des modèles d'architecture, évaluer leur cohérence par rapport aux exigences, ou réutiliser des spécifications.

J'appartiens à un courant de chercheurs qui considère que le concept de but à un rôle majeur à jouer en Ingénierie des Exigences. Sans forcément chercher à définir « un langage de modélisation de plus », j'ai expérimenté divers langages de modélisation de buts pour résoudre divers types de problèmes. Lorsque cela s'est avéré utile, j'ai néanmoins participé au développement ou à l'amélioration de nouveaux langages de modélisation.

La quasi-totalité de mes travaux et de mes publications présentent quelques caractéristiques communes.

- Ils font apparaître la *modélisation intentionnelle* comme un vecteur important de résolution des problèmes d'Ingénierie des Exigences. L'emploi de l'expression "modélisation intentionnelle" pour qualifier l'activité de développement de modèles de buts fait apparaître explicitement qu'il s'agit de buts d'acteurs humains que le système développé doit satisfaire.
- Ils positionnent l'Ingénierie des Exigences dans le contexte de *l'Ingénierie des Systèmes d'Information*. J'ai effectué l'ensemble de mes recherches au Centre de Recherche en Informatique de l'Université Paris 1 Panthéon – Sorbonne dont la spécialité est l'Ingénierie des Systèmes d'Information. Je me suis naturellement positionné dans ce contexte pour contribuer à la recherche

du laboratoire. L'Ingénierie des Exigences a cependant un rôle à jouer dans d'autres domaines, tels que le Génie Logiciel, l'Ingénierie des Systèmes, ou l'étude des Interfaces Humain Machine, et en dehors des Technologies de l'Information et de la Communication (TIC) pour d'autres types de systèmes tels que les logiciels embarqués, les systèmes matériels, ou les systèmes de production. Certains des résultats que j'ai pu produire pourraient très vraisemblablement s'appliquer dans ces contextes. Cet axe est à envisager pour une meilleure compréhension des théories développées. J'ai néanmoins choisi de ne pas explorer cette voie pour l'instant.

- Une troisième caractéristique commune aux divers travaux que j'ai menés est ma stratégie de recherche. Pour l'essentiel, ma recherche est de nature *hypothético-déductive*. C'est à dire que partant de problèmes que j'ai choisi de résoudre, j'ai émis des hypothèses à partir desquelles j'ai essayé de déduire par raisonnement des solutions originales. La manière de définir des hypothèses et la nature des déductions n'est pas strictement mathématique; le domaine s'y prête en fait assez mal. Les hypothèses sont en général formulées sous la forme de questions de recherche définies à partir de cadres méthodologiques. Les déductions apparaissent sous la forme d'approches méthodologiques. Malgré la nature relativement « soft » de l'objet de mes recherches, j'ai toujours essayé de respecter une articulation hypothèse-solution qui soit raisonnable (pour ne pas dire aussi rigoureuse que possible), et surtout, j'ai cherché à évaluer scientifiquement mes hypothèses en menant des expériences du type de celles le plus souvent réalisées dans mon domaine. Par exemple la critique experte, les expériences empiriques, ou les études de cas.

Chacun de mes travaux fait l'objet de plusieurs publications dans des conférences internationales avec comité de sélection et dans des revues. J'ai introduit le résultat de mes différentes recherches dans mes enseignements, par exemple sur les Progiciels de Gestion Intégrés, sur l'Evolution des Systèmes, sur le lien entre Ingénierie des Exigences et Architecture, et surtout dans mon cours d'initiation à la recherche intitulé « Ingénierie des Exigences et stratégie de Décision ».

Enfin, j'ai cherché autant que faire ce peut à valoriser mes résultats de trois manières: par le développement de méthodes d'Ingénierie des Exigences (L'Ecritoire, ACEM, MIBE, NENO, INSTAL, GAP, RITA, REDPL), d'outils pour la mise en œuvre de ces méthodes (L'Ecritoire, RITA, REDPL), et lorsque cela a été possible par le déploiement de ces méthodes et outils dans le contexte industriel (collaborations avec les sociétés RCI, SNCF, Renault, BNP Paribas, Crédit Lyonnais, STAGO Instruments).

2. Organisation du mémoire

Ce mémoire est organisé en quatre chapitres principaux qui coïncident à quatre grandes problématiques de l'Ingénierie des Exigences que j'ai décidé de traiter dans ma recherche: l'identification des exigences pour le développement de nouveaux systèmes ("from scratch"), la découverte des exigences par mise en coïncidence, l'Ingénierie des Exigences dans le contexte de l'évolution, et l'alignement.

Chaque chapitre débute par une introduction générale à la problématique et au positionnement de mes travaux. Chacune des sections suivantes coïncide à une contribution spécifique dans le contexte de la problématique traitée. Chaque contribution est introduite en situant l'angle sous lequel la grande problématique est traitée et en

définissant clairement la question de recherche qui a guidé l'exploration de la problématique. Les sous sections suivantes présentent (a) une définition plus précise des problématiques traitées, (b) les caractéristiques distinctives de la théorie que j'ai développée, et enfin (c) une présentation des résultats produits. Dans tous les cas, la théorie que j'ai développée consiste en une approche méthodologique visant à résoudre les problématiques choisies.

Avant de poursuivre la présentation de mes travaux, il me paraît indispensable de rappeler que ces travaux sont le fruit de collaboration avec mes collègues, doctorants, et partenaires industriels. Dans certains cas j'ai été l'instigateur, dans d'autres je n'ai été que participant. La plupart des idées ont été développées de manière collaborative, je n'en suis pas toujours l'inventeur originel. Empruntant à nouveau la métaphore de l'exploration des océans pour représenter l'activité de recherche, je peux dire que j'ai eu autant plaisir à jouer le rôle de matelot que de second, et il m'est parfois arrivé de définir le cap sur la partie de la carte qu'il me semblait intéressant d'explorer à bord de ma petite embarcation.

CHAPITRE 2

Identification des exigences pour le développement de nouveaux systèmes ("from scratch")

1. Introduction et positionnement de mes travaux sur l'identification des exigences

Le cadre de travail dans lequel se sont inscrits mes premiers travaux de recherche (notamment ceux développés lors de ma thèse de doctorat) est le cadre traditionnel du développement d'un nouveau système. Ce cadre souvent nommé As-Is/To-Be en ingénierie des systèmes d'information fait apparaître le développement de tout système d'information comme la transition discrète d'une situation courante (As-Is) à une nouvelle situation (To-Be). En 1994, ce cadre est adapté par M. Jarke au domaine de l'Ingénierie des Exigences [Jarke1994]. Le cadre de travail défini par M. Jarke (Figure 1) fait alors apparaître trois observations centrales:

« la première est que le rôle central des spécialistes de l'informatique en Ingénierie des Exigences est abandonné en faveur d'une participation plus large, non seulement des utilisateurs et de leur managers, mais aussi d'autres "victimes" potentielles des systèmes informatisés; l'implication est que la définition des exigences doit inclure des modèles de contexte, et une traçabilité de toutes les décisions de conception jusqu'à la vision du système centralisé, ainsi qu'aux lois, politiques, et habitudes de toutes les parties prenantes.

La seconde est que l'approche initiale one-off de l'ingénierie des exigences est remplacée par une pratique de l'ingénierie des exigences qui apporte une valeur ajoutée en rassemblant les composants standards de manière innovante, plutôt que d'innover en réinventant constamment la roue.

La troisième observation est que le support conceptuel à la gestion du changement est en train de devenir la tâche centrale de l'Ingénierie des exigences. On peut voir les modèles d'exigence comme des "compagnons virtuels" qui aident les organisations (changeantes) de manière proactive à supporter leurs processus de changement au travers d'infrastructures et d'applications TIC (changeantes) »

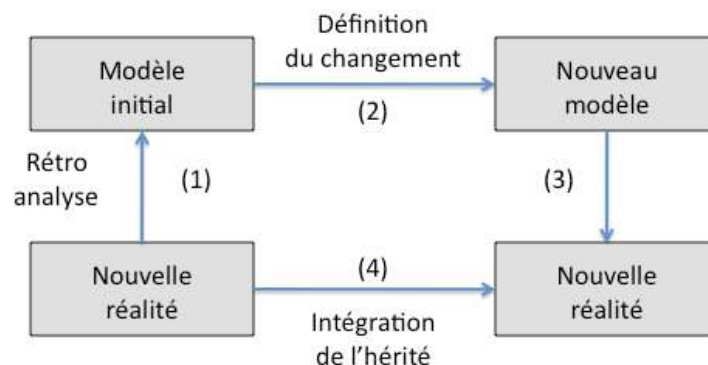


Figure 1: La gestion du changement en Ingénierie des Exigences, d'après [Jarke1994]

Le cadre de M. Jarke fait une distinction claire entre l'ancienne et la nouvelle situation toutes deux appréhendées par des modèles. Au moment où les activités d'Ingénierie des Exigences sont entreprises, ce qui va devenir l'ancienne réalité correspond en fait à la situation courante "As-Is", la nouvelle réalité est la situation future "To-Be".

La situation courante peut aussi bien être caractérisée par l'absence de système d'information informatisé, par le besoin de renouveler entièrement le système d'information hérité (legacy), ou par le besoin d'y introduire un nouveau composant à développer. La situation courante peut être modélisée. Son analyse permet de définir l'Univers de Discours en explorant un certain nombre d'éléments "contextuels" du système à développer, tels que la terminologie métier, la structure de l'organisation pour laquelle le système va être développé, ou celle d'autres systèmes avec lesquels le nouveau système devra interagir. Une caractéristique essentielle de la situation courante est qu'elle comporte intrinsèquement les éléments qui rendent le développement d'un nouveau système nécessaire, comme par exemple la présence de dysfonctionnements, des problèmes de performance, ou des choix stratégiques impliquant de nouvelles formes d'automatisation.

Le principal écart entre la situation courante et la situation future est caractérisé par le fait que cette dernière intègre un nouveau³ système. En outre, des différences peuvent apparaître au niveau de l'environnement de fonctionnement du système. Par exemple, la nature des interactions entre le nouveau système et ceux avec lesquels il interagit peut être changée. La structure de l'organisation dans laquelle le nouveau système a été introduit peut, elle aussi, changer du fait de résolution de dysfonctionnements, ou de changements organisationnels permis par la mise en place du nouveau système. Beaucoup de ces changements sont de nature fonctionnelle, par exemple lorsqu'il s'agit d'automatiser des tâches administratives. Les changements peuvent néanmoins être aussi de nature qualitative, par exemple lorsqu'il s'agit d'améliorer la performance globale de l'organisation, ou des systèmes qui y sont employés.

Le cadre As-Is/To-Be fait apparaître que le système développé trouve sa justification dans les modèles d'exigences qui définissent la situation To-Be. Une mission essentielle de l'Ingénierie des Exigences est de guider de manière systématique la découverte de ces exigences, en garantissant un certain nombre de bonnes caractéristiques telles que leur satisfiabilité, leur cohérence, ou leur complétude.

La nature optionnelle des exigences fait qu'il est le plus souvent possible de choisir entre les mettre en œuvre ou pas. La finalité du processus de priorisation est de décider de la collection des exigences qui seront réellement mises en œuvre dans le système à développer parmi toutes celles qui peuvent être considérées comme optionnelles.

La priorisation des exigences est définie comme "*la sélection des 'bonnes' exigences à partir d'un super-ensemble donné d'exigences candidates, de manière à ce que les différents intérêts, les contraintes techniques et les préférences des parties prenantes soient satisfaits et que la valeur globale pour l'organisation soit maximisée*" [Berander2005]. En pratique, le problème de la

³ Par "nouveau" j'entends "qui n'existait pas dans la situation initiale", la question soulevée est donc bien celle de l'introduction d'un système informatique qui n'existait pas auparavant.

priorisation des exigences est dû (a) au fait que l'on dispose de ressources insuffisantes (temps, argent, compétences) pour mettre en œuvre toutes les exigences, et (b) au fait que différentes parties prenantes peuvent avoir des exigences contradictoires – qu'il est difficile de mettre en œuvre dans le même système.

2. Positionnement

Les approches d'identification des exigences que j'ai développées au CRI se placent dans le cadre de travail du développement de nouveaux systèmes. Dans ce cadre de travail, les buts servent en général à identifier les exigences à l'égard du système à développer, mais aussi à les spécifier et à les analyser. Les trois problèmes que j'ai particulièrement explorés se positionnent dans le contexte de l'identification des exigences. En particulier, je me suis intéressé aux problèmes: (a) de la découverte des exigences, (b) de la garantie de leur complétude, et (c) du choix des exigences à opérationnaliser par le futur système. Dans les trois approches que j'ai développées (*L'Écritoire*, RITA et NENO), les exigences sont associées à des buts qui sont modélisés.

La prochaine section présente l'approche *L'Écritoire*. Cette approche, développée dans le contexte du Projet Européen FPV CREWS combine la modélisation des buts à celle des scénarios. D'un côté, les buts sont utilisés pour localiser les scénarios. Chaque scénario décrit partiellement comment les utilisateurs s'attendent à ce que le nouveau système participe à la satisfaction d'un but. L'analyse systématique des scénarios permet de découvrir de nouveaux buts. La collection de buts et de scénarios ainsi produite forme un ensemble de besoins qui peuvent ultérieurement être reformulés, triés et intégrés aux spécifications des exigences du système. *L'Écritoire* a fait l'objet de nombreuses publications. Cette approche figure parmi les premières à avoir considéré buts et scénarios sous l'angle de leur couplage est fréquemment citée et fait l'objet de nombreuses références.

La section suivante présente l'approche RITA. Dans cette approche, le problème n'est plus de découvrir les besoins initiaux des utilisateurs, mais de garantir la complétude d'exigences déjà spécifiées. Pour ce faire, l'approche propose d'analyser les obstacles à la satisfaction des buts auxquels les exigences déjà connues sont associées. Cette approche a été développée en partenariat avec l'ESSEC, dans le contexte d'une thèse de doctorat que j'ai co-dirigée avec Colette Rolland.

La dernière section présente l'approche NENO d'arbitrage. L'arbitrage participe à la transformation des besoins en exigences: il s'agit de retenir parmi les besoins initiaux ceux qui seront réellement pris en compte dans le développement du système. Dans cette approche, les modèles de buts ont un apport triple: d'une part, ils permettent d'organiser la collection des besoins auxquelles l'arbitrage doit être appliqué. D'autre part, ils permettent de définir les critères d'arbitrages et d'étudier les interactions entre ceux-ci. Enfin, la structure particulière des modèles de buts y est exploitée pour résoudre le problème de scalabilité rencontré dans les arbitrages appliqués à de très grandes collections d'exigences. Cette démarche a été développée en collaboration avec la société Renault au sein de laquelle elle a été expérimentée dans le cadre de la réforme des canaux de vente de véhicules neuf à l'international.

Découverte des exigences par les buts et les scénarios - l'approche *L'Écritoire*

Alors que les premières méthodes d'Ingénierie des Exigences se sont attachées aux langages et aux modèles de spécification des exigences, la communauté de l'Ingénierie des Exigences, s'est plus récemment concentrée sur les mécanismes qui régissent leur découverte.

D'une façon générale, la *découverte des exigences* est l'activité qui consiste à faire apparaître les demandes et besoins émis par les diverses parties prenantes, puis à les formuler sous la forme d'exigences à l'égard d'un système.

La découverte des exigences est étroitement liée aux tâches de validation, de trace, de configuration, et de réutilisation. Il s'agit aussi d'un processus éminemment coopératif qui fait intervenir de nombreuses parties prenantes tels que les utilisateurs, clients, experts de domaine, directeurs de projet, et même les concepteurs d'autres systèmes déjà en place. Beaucoup de ces parties prenantes n'ont pas de compétence en Ingénierie des Exigences. Ils ne savent pas comment procéder et connaissent rarement les langages et modèles de spécification des exigences. Un problème important est le 'fossé' entre les utilisateurs (qui formulent des demandes et expriment des besoins mais ne savent pas spécifier des exigences) et les développeurs (qui savent comment spécifier et exploiter les exigences, mais ne connaissent pas les besoins à spécifier). Dans ce contexte, la question de recherche que nous avons posée est:

"comment améliorer la participation des utilisateurs à l'activité de découverte des exigences?"

La communauté de l'Ingénierie des Exigences a développé depuis une quinzaine d'années un certain nombre de techniques orientées vers les utilisateurs. Les *approches de découverte des exigences par scénarios* appartiennent à cette famille de techniques: permettant aux utilisateurs d'exprimer leurs besoins dans leur propre vocabulaire. Les scénarios sont donc un outil adapté à la découverte des exigences auprès des utilisateurs.

2.1. Problématique de la découverte des exigences par les scénarios

Quoique récente, l'utilisation des scénarios pour découvrir les exigences est déjà fréquente dans la pratique industrielle. Cependant, les enquêtes auprès des industriels [Jarke1997], [Weidenhaupt1998] mettent en évidence les problèmes suivants: les utilisateurs ont besoin de plus d'aide méthodologique pour écrire les scénarios, les organiser, les représenter formellement, en extraire les besoins. Ils ont besoin de méthodes pour identifier les scénarios, et pour vérifier leur correction, leur cohérence et leur complétude. Nous avons identifié [Jarke1997] cinq groupes génériques de problèmes liés aux approches de découverte des exigences par scénarios:

- i. la difficulté de rédaction des scénarios ;
- ii. le passage de scénarios informels à des scénarios formels ;
- iii. la mise en œuvre du paradigme 'boîte noire - boîte blanche' ;
- iv. le guidage méthodologique ;
- v. la fragmentation des scénarios ;
- vi. la relation scénarios - exigences.

i. Rédaction des scénarios: notre étude de la pratique industrielle [Jarke1997] montre que les scénarios sont le plus souvent écrits sous forme textuelle. La *rédaction des scénarios* consiste non seulement à écrire les scénarios,

mais aussi à les transformer: par exemple pour corriger les erreurs qu'ils peuvent contenir, en raison de leur nature informelle. La rédaction de scénarios textuels soulève de nombreuses questions. Par exemple, les scénarios étant des textes narratifs, qu'est-ce qui les différencie d'autres types de textes? Comment vérifier que des scénarios textuels sont écrits correctement, qu'ils ne contiennent pas d'ambiguïtés, ou d'incohérences? Comment même vérifier qu'il s'agit bien de scénarios qui sont écrits? Chacune de ces questions met en lumière l'une des difficultés auxquelles on est confronté lors de la rédaction de scénarios textuels.

ii. Transformation de scénarios informels en formels: si la nature informelle des scénarios permet aux utilisateurs d'exprimer plus facilement leurs besoins, elle est aussi problématique parce que la qualité des exigences spécifiées à partir de ces besoins ne peut être contrôlée que par des moyens formels. Il est donc nécessaire de gérer les scénarios à la fois sous forme formelle et sous forme informelle. Les questions sous-jacentes au problème de la transformation de *scénarios informels en scénarios formels* sont les suivantes: quelles sont les relations entre l'expression informelle d'un scénario et sa spécification formelle? Comment s'assurer que le système produit est bien celui qui est attendu? Il est nécessaire de résoudre ces problèmes pour permettre, par exemple, l'automatisation d'un raisonnement basé sur les scénarios.

iii. Difficile application du principe boîte noire – boîte blanche: il a été montré que les acteurs de l'industrie aiment appliquer le principe *boîte noire - boîte blanche*. L'approche consiste à considérer le système de manière totalement externe (boîte noire) ou pas (boîte blanche). Le principe est de garantir que les propriétés de la boîte noire sont préservées lorsque l'intérieur en est décrit. Jarke [Jarke1997] montre que de nombreux rédacteurs mélangent différents niveaux de détail dans les mêmes scénarios, risquant par la même de générer des collections incohérentes d'exigences, ou de répéter le travail d'analyse de manière inutile. La difficulté réside dans la définition de niveaux, ainsi que dans le respect de ces niveaux au cours de la rédaction des scénarios.

iv. Manque de guidage méthodologique: les praticiens manquent de *guidage méthodologique*. Si l'on trouve dans la littérature des descriptions extensives des scénarios en tant que produits de méthodes, peu indiquent systématiquement ce qu'il est possible de faire avec les scénarios, quand le faire, ou même comment le faire. Il est généralement admis qu'un tel guidage devrait être fourni à partir d'un modèle de processus. Cependant, le plus souvent, les solutions proposées sont plutôt basées sur des heuristiques manuelles, informelles et 'artisanales' que sur des directives systématiques.

v. Fragmentation des scénarios: la *fragmentation des scénarios* pose les problèmes de leur organisation et de l'identification de nouveaux scénarios dans une collection existante [Cockburn1996]. Le principe 'boîte noire - boîte blanche' permet d'organiser les scénarios en niveaux et par conséquent, répond en partie aux difficultés dues à leur fragmentation. Cependant, cela est clairement insuffisant: il reste nécessaire d'organiser les scénarios appartenant au même niveau.

vi. Relation scénarios-besoins: tout scénario n'est qu'un fragment de connaissance du système pour lequel des besoins peuvent être spécifiés puis transformés en exigences. Les relations entre scénarios et besoins (et au-delà exigences) sont rarement explicites dans les définitions des scénarios. Pourtant, la relation scénarios-besoins n'est pas triviale: les scénarios sont-ils des besoins, des exigences? Si non, de quelle manière s'y rattachent-ils? En sont-ils issus ou inversement les besoins font-elles partie des scénarios?

Les problèmes liés à la difficulté de rédaction des scénarios, au passage de scénarios informels à des scénarios formels, à l'application du principe 'boîte noire - boîte blanche', à la fragmentation des scénarios, à la relation scénarios - exigences, et au guidage méthodologique sont les problèmes qui ont été abordés dans nos travaux.

2.2. *Principes de l'approche L'Écritoire*

Nous avons choisi de développer une *approche de découverte des besoins basée sur l'utilisation de scénarios textuels*, afin de répondre à la pratique que nous avons le plus couramment observé dans notre état de la pratique en industrie [Jarke1997]. Tout scénario encapsule une partie des connaissances que l'utilisateur a du système. L'hypothèse principale de l'approche *L'Écritoire* s'inscrit dans l'un des nombreux courants de recherche autour des scénarios [Rolland1998a]: les scénarios sont principalement exprimés de façon informelle.

Le principe élémentaire et original de l'approche *L'Écritoire* réside dans le *couplage bidirectionnel des buts et des scénarios* [Rolland1998b]. Dans un sens, les buts servent à identifier les scénarios qui les concrétisent. Dans l'autre sens, les scénarios permettent de découvrir les buts.

2.2.a Organisation des buts et scénarios en une hiérarchie de fragments de besoins

L'Écritoire formalise le couplage entre buts et scénarios fort par un méta-modèle au centre duquel se trouve le concept de *fragment de besoin*. Tout fragment de besoin associe un but à un scénario. Les fragments de besoins sont organisés à plusieurs niveaux d'abstraction dans une hiérarchie dans laquelle les liens qui les unissent sont définis de manière systématique. Trois types de liens sont définis dans le méta-modèle: le lien d'alternative (OR), le lien de complémentarité (AND) et le lien d'affinement (Refine).

Le méta-modèle conceptuel de scénarios définit d'autres aspects de l'organisation des buts et des scénarios [Benachour1998a]:

la *structure conceptuelle* des scénarios, selon laquelle les scénarios peuvent être formalisés; le méta-modèle fait apparaître le concept d'action au centre de la définition formelle des scénarios. Une typologie d'actions simples et complexes permet d'appréhender les différents phénomènes qui peuvent être décrits par les scénarios de *L'Écritoire*. Les actions complexes permettent en particulier de combiner des actions simples.

- Les *niveaux d'abstraction* auxquels les scénarios peuvent être définis. Trois niveaux d'abstraction sont formalisés en particulier: contextuel, interaction et interne. L'emploi de ces niveaux d'abstraction permet d'assurer un certain degré de cohérence entre les scénarios qui y sont décrits.

La distinction formelle des *scénarios normaux et exceptionnels* coïncidant respectivement à la description de cas nominaux et non nominaux de fonctionnement du système. Le méta-modèle *L'Écritoire* spécifie comment les scénarios normaux et exceptionnels doivent être combinés.

2.2.b Couplage bi-directionnel buts-scénarios

L'approche de découverte des besoins proposée fournit une collection complète et organisée de scénarios écrits en langage naturel spécifiant les besoins du système. Par ailleurs, elle aide à la production de cette collection de scénarios par un processus guidé transformant progressivement des scénarios décrits en langage naturel en des

textes non ambigus, correctement structurés, et intégrés dans la collection de scénarios. Enfin, elle aide à découvrir les exigences par une analyse systématique des scénarios.

Nous définissons le processus de découverte des exigences basé sur la production de collections de scénarios organisés hiérarchiquement comme un processus incrémental. Comme le montre la Figure 2, notre approche vise à *aider* à :

décrire l'usage que les utilisateurs prévoient de faire du système dans des *scénarios textuels*, nous appelons cette étape la *rédaction des scénarios*. Les directives proposées par *L'Écritoire* pour guider cette étape sont décrites dans [Benachour1998b] et [Salinesi2004a];

- intégrer les scénarios dans des *collections de scénarios*, nous appelons cette étape *l'organisation des scénarios*.
- découvrir *des exigences* par l'analyse de scénarios, nous appelons cette étape *l'analyse des scénarios*

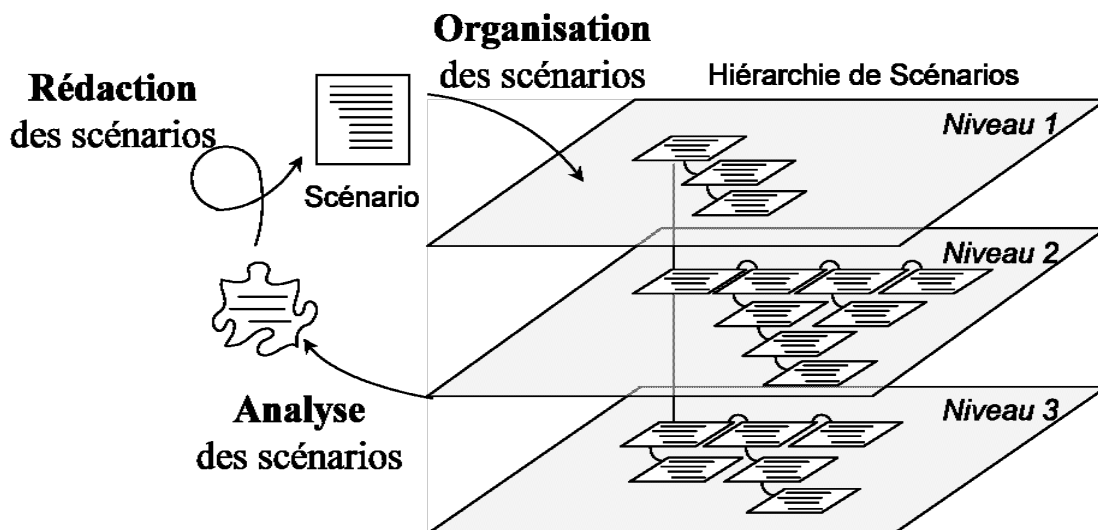


Figure 2: Aperçu de la méthode L'Écritoire

Sans entrer dans les détails, les buts jouent un rôle particulièrement important dans chaque étape guidée par *L'Écritoire*. Ils servent à identifier les scénarios et donc à en assurer l'organisation, à cerner leur contenu au moment de la rédaction, ce sont eux qui sont initialement produits lors de l'étape d'analyse des scénarios, et ils servent enfin à la formalisation des exigences découvertes.

Différentes règles sont proposées pour guider la découverte de nouveaux buts par l'analyse des scénarios. Conformément à la structure d'organisation des scénarios AND/OR/Refine préconisée par le méta-modèle conceptuel, *L'Écritoire* propose des règles pour produire spécifiquement: soit des buts complémentaires au but auquel le scénario analysé est associé, soit des buts alternatifs, soit des buts qui affinent le but de départ.

2.2.c Modélisation linguistique et rédaction des scénarios

Le choix de la forme textuelle nécessite le support du langage naturel et rend critique la *rédaction des scénarios*. Notre approche recherche un juste milieu entre un mode d'utilisation du langage naturel entièrement libre, ce qui augmenterait le risque d'erreur, et l'utilisation forcée de structures prédéfinies qui, à notre avis, contraindrait trop

le comportement du rédacteur. Notre solution combine l'emploi de la prose narrative lors de l'expression des scénarios avec l'exploitation de la structure du langage naturel lors de leur analyse.

L'étape de rédaction des scénarios est elle-même un processus complexe durant lequel un récit court et informel est écrit puis transformé en une description complète, non-ambigüe et structurée. Nous cherchons à résoudre la difficulté de rédaction des scénarios en guidant de bout en bout ce processus de transformation de scénarios informels en des scénarios formels. La transformation de scénarios définis de manière informelle en scénarios formalisés (et qui peuvent donc être analysés à leur tour) repose sur l'usage intensif d'outils d'analyse linguistique qui exploitent systématiquement un modèle linguistique de scénario.

Dans *L'Écritoire*, le modèle linguistique de scénarios délimite le sous-ensemble du langage naturel qui peut être employé pour rédiger des scénarios. L'approche *L'Écritoire* s'inspire du modèle de Chomsky qui distingue le niveau de surface (de la structure syntaxique) et le niveau profond (de la sémantique) du langage naturel. Ainsi, le *modèle linguistique* de scénarios de *L'Écritoire* est constitué de 9 *schémas sémantiques* et 68 *structures linguistiques* qui définissent formellement à ces deux niveaux le sous-ensemble du langage naturel qui peut être employé pour rédiger des scénarios.

La rédaction initiale des scénarios est guidée afin d'aider les rédacteurs à se conformer au modèle linguistique de scénarios. Ce guidage est apporté au travers de directives de rédaction qui, conformément au modèle linguistique, indiquent quels doivent être le style et le contenu de scénarios textuels. 10 *directives de style*, et 9 *directives de contenu*, sont ainsi proposées dans [Benachour1998a], [Benachour1999a], et [Salinesi2004a] et accompagnée de gabarits qui peuvent être utilisés lorsque les parties prenantes ne désirent pas rédiger les scénarios sous la forme de texte entièrement libre. En outre, des *directives de contenu* sont proposées afin de rappeler aux rédacteurs la nature conceptuelle des scénarios à rédiger. Les directives de styles indiquent les attributs attendus en association aux scénarios et liens qui peuvent être tirés entre scénarios. Elles sont définies conformément au modèle conceptuel de scénario. Les directives de style indiquent comment les scénarios textuels peuvent être rédigés. Les directives de contenu indiquent ce qui doit être mis dans des scénarios textuels. Les directives de style et de contenu sont respectivement définies conformément aux structures linguistiques et aux patterns sémantiques du modèle linguistique de scénarios.

Le modèle linguistique de scénarios est opérationnalisé par 10 *règles d'analyse syntaxique*, 68 *règles d'analyse structurelle*, et 68 *règles d'analyse sémantique*. Ces règles, définies de manière déclarative, supportent l'interprétation des scénarios décrits en langage naturel selon le modèle linguistique.

Une difficulté de l'interprétation du langage naturel tient à la liberté des rédacteurs. Bien que les directives de rédactions aident à appliquer le modèle linguistique de scénario, le respect total de celui-ci ne peut en fait jamais être garanti a priori. Le succès de l'analyse linguistique des scénarios peut donc être compromis. *L'Écritoire* propose 3 *règles de clarification*, et 7 *règles de complétude*, qui amènent les rédacteurs à identifier les erreurs de rédaction de scénario -en terme de respect du modèle linguistique- et à les corriger.

Enfin, *L'Écritoire* propose 17 *règles de transformation* du modèle linguistique au modèle conceptuel qui permettent de progresser dans l'axe de la formalisation des scénarios. Cette formalisation est nécessaire pour pouvoir supporter les étapes ultérieures d'analyse des scénarios.

Les détails du modèle linguistique, et des directives de rédaction, sont donnés dans [Rolland1998c], [Benachour1998c], et [Benachour1999a]; les règles de correction et d'analyse linguistique et de transformation sont présentées dans [Benachour1997] et [Rolland1998c] et [Benachour1999a].

2.2.d Découverte des buts par l'analyse des scénarios

Les scénarios étant '*des fragments parcellaires de connaissance*' (v), ils font toujours partie d'un ensemble. Nous proposons de traiter le problème de la fragmentation des scénarios en *organisant* ceux-ci de manière hiérarchique. Les hiérarchies de scénarios sont définies à plusieurs niveaux. Un scénario, présenté comme une boîte noire à un niveau donné, peut être vu comme une boîte blanche à un niveau inférieur. L'organisation des scénarios est donc compatible avec le principe 'boîte noire - boîte blanche' (iii).

Une fois rédigé et intégré dans une hiérarchie, un scénario peut être analysé pour découvrir de nouveaux besoins. Dans *L'Écritoire*, les besoins sont représentés par des *buts*. Chaque scénario correspond à un but donné. C'est par conséquent l'ensemble des buts attachés aux scénarios d'une hiérarchie qui définit les besoins des utilisateurs à l'égard du système. De même que pour les scénarios auxquels ils sont attachés, les buts sont liés à travers des relations définies dans la hiérarchie de scénarios. Conformément aux principales approches de modélisation de but en vigueur, les buts définis dans *L'Écritoire* peuvent être complémentaires, alternatifs, ou coïncider à différents niveaux d'abstraction ou de détail. L'analyse des scénarios est rendue systématique en exploitant cette propriété: les besoins sont extraits en complétant les hiérarchies de scénarios avec de nouveaux scénarios à rédiger et analyser, et à positionner de manière complémentaire, alternative ou affinée par rapport aux scénarios de départ. Plusieurs directives d'analyse spécifiques sont proposées pour découvrir de nouveaux besoins dans chacune de ces trois directions, comme le montrent [Rolland1998b], [Benachour1999b] et [Tawbi1999].

Conformément à la problématique (vi) la relation entre scénarios et besoins est donc bien double: les besoins permettent de rédiger les scénarios, et les scénarios servent à extraire de nouveaux besoins.

2.3. Résultats

Mes recherches sur l'ingénierie des exigences par les scénarios ont été menées dans le cadre du projet Européen LTR (Long Term Research) CREWS (FP5 No 21.903). Deux travaux ont été effectués en amont du développement de la méthode *L'Écritoire*, en collaboration avec les partenaires du projet CREWS (Faculté Universitaire Notre Dame de La Paix de Namur - Belgique, City University – Royaume Uni, Université d'Aix la Chapelle – Allemagne). Il s'agit d'une part, d'une étude de l'usage des scénarios dans la pratique industrielle de l'Ingénierie des Exigences, et d'autre part, d'un travail de recherche bibliographique qui a permis d'établir un panorama des méthodes existantes.

Les résultats produits –de manière individuelle ou collective- dans le cadre de mes travaux sur l'ingénierie des exigences par les scénarios sont les suivants:

- Une revue de la littérature structurée par un cadre multidimensionnel original ;
- Une étude de la pratique de l'ingénierie des exigences par les scénarios qui a mis en évidence l'emploi systématique du langage naturel dans la description des scénarios ;

- Une formalisation du couplage but-scénario qui permet notamment de mieux définir le concept de cas d'utilisation;
- Une formalisation linguistique des scénarios cohérente avec leur définition conceptuelle ;
- Des directives méthodologiques permettant de guider la rédaction de scénarios textuels, ainsi que leur conceptualisation et leur analyse ;
- Une évaluation de ces directives au moyen d'une série d'ateliers de travail, d'études de cas, d'expériences empiriques, et de tutoriels.

Ces résultats ont fait l'objet de plusieurs publications de haut niveau. On citera en particulier la participation à la publication d'articles dans les revues Requirements Engineering Journal [Rolland1998a], IEEE Transactions on Software Engineering [Rolland1998b], Data and Knowledge Engineering Journal [Rolland1998c], Journal of Computer Systems Science & Engineering [Benachour1999b], d'un article dans le Symposium International IEEE Requirements Engineering [Benachour1999c], et de plusieurs articles dans des ateliers internationaux et conférences de rang moindre [Benachour1997] [Benachour1998a] (International Workshop on Requirements Engineering: Foundations for Software Quality), [Benachour1998b] (European-Japanese Conference on Information Modelling and Knowledge Bases), [Benachour1998c] (International Workshop on Natural Language and Information Systems). Ces travaux ont enfin été publiés dans le cadre de ma thèse de doctorat [Benachour1999a].

Au début de mes travaux de recherche sur les scénarios, l'état de la recherche et de la pratique en ingénierie des exigences par les scénarios faisait apparaître une très grande diversité dans les approches. Non seulement les définitions du concept de scénario variaient de manière très importante, mais en plus les approches offraient des manières très diversifiées de créer et d'exploiter les scénarios. J'ai participé à l'élaboration d'un état de l'art et à sa publication dans un article de la revue Requirements Engineering qui fait l'objet de plus de 150 citations [Rolland1998a]. Un aspect original de cet état de l'art est l'emploi d'un **cadre multidimensionnel** qui permet non seulement de classer les méthodes existantes, mais aussi de les comparer et de mieux comprendre ce que l'ensemble des méthodes existantes permet de faire ou pas. Le cadre est organisé autour de quatre dimensions, le contenu, la forme, le propos et le cycle de vie qui proposent d'étudier les différentes méthodes d'Ingénierie des Exigences par les scénarios sous l'angle de quatre questions clés: "quelle est la connaissance exprimée dans un scénario?", "sous quelle forme un scénario est-il exprimé?", "pourquoi utiliser un scénario?", et "comment manipuler un scénario?". Notre approche de classification adopte une approche en facettes similaire à celle de Prieto-Diaz et Freeman pour classer des composants [Prieto-Diaz1987]. 25 facettes ont été identifiées à partir de la revue des approches existantes, et définies dans l'une des quatre dimensions du cadre. Chaque facette est un point de vue qui permet de caractériser une méthode dans une dimension et selon une métrique définie formellement. Plus de cinquante approches ont été considérées afin de garantir la qualité du cadre de classification (sa complétude notamment). Onze d'entre elles, choisies en raison de leur représentativité, sont complètement décrites selon l'ensemble des facettes définies par le cadre.

J'ai participé à l'étude de la pratique sous la forme d'une série d'entrevues menées auprès des responsables de 15 projets et complétées par une enquête de grande échelle menée auprès de 253 sociétés européennes de tailles et secteurs d'activité divers. L'étude, publiée sous la forme d'un rapport du projet CREWS [Jarke1997] est citée 17 fois. Les résultats de cette étude ont par ailleurs été repris dans [Weidenhaupt1998] (260 références). Les entrevues, structurées autour des quatre

dimensions définies par le cadre de classification développé pour notre état de l'art ont permis de caractériser les scénarios utilisés dans les projets considérés, et d'obtenir une analyse qualitative de l'efficacité des approches employées. Partant des résultats des entrevues, des questionnaires ont été développés et largement diffusés afin de pouvoir quantifier les tendances observées lors des entrevues. Outre quelques leçons pratiques à l'adresse des industriels, les résultats importants de cette étude consistent en une série de challenges pour la recherche. Ce sont les observations faites lors de ces études qui ont permis de soulever les problématiques considérées dans nos travaux. Une contribution importante de cette étude est en particulier la mise en évidence de *l'emploi quasi systématique du langage naturel pour spécifier les scénarios*. En outre, la structuration de l'étude de la pratique par le cadre de recherche bibliographique a permis de démontrer les différences dans les centres d'intérêt de la recherche et de la pratique, notamment en ce qui concerne le guidage méthodologique, et la gestion du cycle de vie des scénarios.

La méthode *L'Écritoire* **résout le problème du guidage méthodologique de la découverte des besoins par les scénarios au moyen de directives informelles et de règles de guidage semi-automatiques**. Les directives informelles et règles de guidage sont toutes basées sur une définition formelle des scénarios qui est composée de deux modèles: le modèle conceptuel de scénarios et le modèle linguistique de scénarios. Les règles de guidage sont qualifiées de "semi-automatiques" car certaines parties en sont entièrement automatisées alors que d'autres sont à la charge de l'utilisateur. On trouve parmi celles-ci des outils méthodologiques de rédaction, de transformation, d'organisation et d'analyse des scénarios.

Le **modèle conceptuel de scénarios** définit l'ensemble des concepts utilisés dans les scénarios, la relation de couplage entre scénarios et buts, les niveaux auxquels les scénarios peuvent être décrits, ainsi que les structures organisationnelles des hiérarchies de scénarios. Le modèle conceptuel de scénarios est compatible avec la vision des scénarios comme *'des fragments parcellaires de connaissance'*. En effet, une vue cohérente et complète du système ne peut être donnée que par une collection organisée de scénarios. Une contribution essentielle du modèle conceptuel de scénario réside dans la *formalisation du couplage des buts et des scénarios au travers du concept de fragment de besoin*. C'est sur ce couplage fort sur lequel repose l'ensemble de la méthode *L'Écritoire*. Cette vision développée dans l'article [Rolland1998c]. Cet article cité 130 fois a permis d'apporter une *définition formelle du concept de Modèle de Cas d'Utilisation* dont les contours étaient à l'époque extrêmement flous.

Une des grandes difficultés avec les scénarios en langage naturel résidait dans l'antagonisme entre la liberté de rédaction des scénarios et le besoin de formaliser pour en systématiser l'analyse. Notre contribution a résidé dans la *formalisation linguistique du contenu conceptuel des scénarios*. Le **modèle linguistique de scénarios** définit formellement aux niveaux syntactique et sémantique le sous-ensemble du langage naturel qui peut être employé pour rédiger des scénarios.

Dans *L'Écritoire*, l'écriture initiale des scénarios est guidée par des **directives informelles**. Celles-ci aident à limiter la taille des scénarios, à cibler leur contenu; elles fournissent si nécessaire des structures permettant d'écrire des phrases, et enfin elles indiquent ce qui est attendu à ce moment même du processus. Notre hypothèse est que les scénarios sont plus faciles à transformer et que leur analyse amène un résultat plus correct si leur écriture respecte ces directives informelles. Notre contribution au problème de la rédaction de scénarios textuels réside en *trois ensembles de directives (générales, de style et de contenu) de rédaction de scénarios*. Ces directives sont présentées de

manière informelle. Elles peuvent être utilisées séparément ou conjointement en fonction des problèmes auxquels sont confrontés les rédacteurs (apprentissage initial, aide ponctuelle, ou vérification de scénarios déjà rédigés). Les directives de rédaction de scénarios font l'objet de plusieurs publications. On peut en particulier citer l'article [Ben Achour1998a] cité 41 fois, ainsi que le chapitre de livre [Salinesi2004a].

J'ai proposé des *règles d'analyse linguistique, de correction, et d'identification de la structure conceptuelle des scénarios textuels* afin de guider leur **transformation**. Ces règles de transformation que tirent parti de la forme 'texte libre' d'expression des scénarios tout en limitant la complexité de leur interprétation au moyen des modèles conceptuel et linguistique. Notre contribution à la résolution du problème de la nature textuelle des scénarios repose *l'exploitation systématique du modèle linguistique de scénario (a) pour définir le sous-ensemble du langage naturel auxquels coïncident les scénarios textuels compatibles avec le modèle conceptuel de scénarios, (b) pour guider une rédaction de scénarios textuels conformes au modèle linguistique, et (c) pour formaliser les scénarios textuels conformément au modèle conceptuel*. Le guidage de la rédaction des scénarios s'appuie sur des directives informelles d'écriture et sur des règles formelles de correction. Le guidage de la formalisation s'appuie sur des règles d'analyse linguistique et de transformation. Toutes les directives informelles et directives formelles distinguent systématiquement les dimensions syntaxique, sémantique et conceptuelle des scénarios. Un aspect original de notre approche réside dans le fait qu'elle permette de réaliser des raisonnements automatisés au niveau conceptuel sur des scénarios écrits entièrement en langage naturel. L'hypothèse fondamentale concernant la faisabilité des règles d'analyse linguistique sur lesquels repose l'approche est qu'il faut restreindre la rédaction, l'analyse et la transformation des scénarios au strict sous-ensemble du langage naturel défini formellement par les modèles conceptuel et linguistique de scénario.

Les règles de guidage proposées pour automatiser **l'organisation** des scénarios placent ceux-ci dans une hiérarchie structurée conformément au modèle conceptuel de scénarios. Tout scénario y est associé à un but. Nous avons proposé trois règles qui appliquent de manière cohérente chacune des structures que le modèle conceptuel de scénario prévoit dans les hiérarchies de scénarios (OU, ET, et Affine). Une quatrième règle permet de vérifier les propriétés qualitatives de la hiérarchie de scénarios. Le problème de la fragmentation des scénarios est résolu en *couplant chaque scénario à un but, en le positionnant à un niveau d'abstraction formellement défini, et le liant aux autres scénarios alternatifs, complémentaires, et d'un autre niveau d'abstraction*.

De même, les règles de guidage automatisant l'étape **d'analyse** des scénarios proposent une stratégie d'analyse pour chaque type de structure définissant les hiérarchies de scénarios. Les règles que nous proposons guident l'analyse de la structure interne des scénarios (obtenue lors de leur transformation), cherchant à y découvrir les indices de buts alternatifs, complémentaires, ou d'un niveau de détail plus fin, et qui ne seraient pas déjà présents dans la hiérarchie. L'application successive des différentes règles permet de découvrir progressivement, partant d'un unique but de haut niveau, une hiérarchie de buts illustrée par des scénarios, organisée en niveaux d'abstraction, et dans laquelle les différents buts sont associés en fonction de leurs rôles respectifs. La démarche résout le problème de la boîte-noire/boîte blanche par *l'exploration progressive top-down de buts et des scénarios qui leurs sont associés*. Les fragments de besoins ainsi découverts sont destinés à être interprétés dans la perspective des exigences du système, et être priorisés afin de décider s'ils doivent être considérés lors du développement du système.

Le principe du couplage buts-scénarios et les règles d'analyse des scénarios ont fait l'objet de plusieurs publications. J'ai participé à la publication de ces résultats dans la revue IEEE Transactions of Software Engineering [Rolland1998d], la revue Journal of Computer Systems Science and Engineering [Ben Achour 1999] et [Ben Achour 1998a] sont respectivement cités 334 fois, 17 fois, et 41 fois.

Un certain nombre d'évaluations ont été réalisées tout au long de l'élaboration de l'approche *L'Écritoire* afin de vérifier systématiquement la qualité des solutions qu'elle propose, de mieux appréhender ses apports, et de mieux comprendre les améliorations qui pouvaient lui être apportées. Quatre types d'expériences ont été menées:

Ateliers de Travail: trois ateliers de travail ont été menés avec la participation de 52 professionnels Français de l'Ingénierie des Systèmes. Au cours de ces ateliers de travail, la première version d'un prototype de recherche a été présentée et son utilisation discutée. Ces ateliers de travail ont permis de vérifier dès le départ l'adéquation des solutions proposées par *L'Écritoire* aux attentes de nos collaborateurs industriels [Rolland1998a].

Etudes Empiriques: deux études empiriques l'une sur l'approche de guidage du processus de rédaction des scénarios et cas d'utilisation [Benachour1999c], et l'autre sur l'approche de guidage du processus d'analyse de scénario et d'identification de nouveaux besoins [Tawbi2000] ont été menées. Les études empiriques ont été conçues sous la forme de laboratoires vivants au cours desquels des étudiants (69 dans le premier cas, 41 dans le second) ont eu pour tâche l'application réaliste de l'approche *L'Écritoire* dans le cadre de la spécification des besoins pour des systèmes suffisamment simples à appréhender dans un temps limité. Les études empiriques ont permis de vérifier l'efficacité et l'effectivité du guidage proposé par *L'Écritoire*.

Etudes de Cas: une étude de cas complète a été menée [Rolland1999a] afin de vérifier l'utilisabilité de *L'Écritoire* dans le contexte industriel et son extensibilité à des systèmes complexes. Cette étude a permis de montrer l'adéquation de l'approche aux difficultés rencontrées lors de la réorganisation des processus d'entreprise, ici dans le cadre de la dérégulation des marchés Européens de l'électricité.

Tutoriels: six tutoriels d'une journée ont été réalisés afin d'évaluer l'application de *L'Écritoire* à l'aide de son environnement outillé. Lors de ses tutoriels, l'outil a été confié à différents groupes d'étudiants et de professionnels qui ont pu l'utiliser sous notre direction. Les discussions que nous avons eues avec les différents participants nous ont permis de mieux évaluer les difficultés d'utilisation de l'approche liées uniquement à son implémentation [Rolland1999b].

Ces expériences ont permis de mettre à jour un certain nombre de résultats:

- **Concernant l'approche globale:** les évaluations de l'utilité globale de l'approche ont été très positives avec une moyenne de 6,6 sur une échelle de 1 à 7;

Concernant l'aide à la rédaction des scénarios: la proportion de personnes qui ont pu rédiger des scénarios correctement plus de la moitié de leurs scénarios est multipliée par 2 lorsque les directives de style proposées par *L'Écritoire* sont utilisées;

- **Découverte des buts:** 93% des personnes découvrent un nombre de buts deux fois supérieur lorsqu'ils utilisent la règle A1;

- *Découverte des buts*: 92% des personnes qui utilisent les règles C1 et C2 sont amenées à corriger leurs erreurs dans la définition des fonctions du système;
- *Organisation de la collection de buts et de scénarios*: l'aide à l'organisation des buts et scénarios est jugée utiles avec une moyenne de 5,46 sur une échelle de 1 à 7;

Guidage du processus de découverte des exigences: 87 % des personnes perçoivent l'aide apportée par le processus méthodologique *L'Écritoire* comme un bénéfice majeur de l'approche;

- *Approche descendante*: 80% de la hiérarchie de buts du cas d'étude industriel ELEKTRA a été découverte à partir d'un seul but et d'un seul scénario.

Ces résultats ont été publiés de manière individuelle dans plusieurs articles [Rolland1998a] [Benachour1999c] [Tawbi2000] [Rolland1999a]. En outre, ils sont mentionnés dans les articles [Rolland1998d] [Ben Achour1999a] de plus grande envergure. Enfin, l'évaluation des directives de rédactions de scénarios a été publiée dans un article au symposium IEEE Requirements Engineering [Ben Achour 1999b], cité 51 fois.

3. Complétude des exigences par l'analyse des menaces - l'approche RITA

Les exigences se doivent de revêtir un certain nombre de qualités: en particulier d'être correctes, c'est à dire cohérentes et complètes [Zowghi2003]. Alors que de nombreux travaux se sont intéressés à l'aspect cohérence, peu de recherches ont été menées pour traiter du problème de la complétude (et de son pendant l'incomplétude) des exigences. Il est pourtant reconnu que la complétude des exigences prend une part cruciale dans la qualité des systèmes produits; les études montrent que les incidents de systèmes liés à l'informatique et les pannes informatiques trouvent plus souvent leur cause dans des problèmes de complétude des exigences que dans des erreurs de codage. Une approche naïve (quoique parfois inévitable, comme l'indique par exemple R Lutz [Lutz2005]) consiste à attendre que les effets soient constatés pour chercher la source des erreurs. Or le coût de résolution d'un problème après livraison d'un système est jusqu'à 100 fois supérieur aux coûts engendrés si le problème est découvert au niveau Ingénierie des exigences [Boehm2001].

La question qui a guidé notre recherche sur la complétude des exigences est la suivante:

"Quelle démarche méthodologique employer pour assurer en amont du développement du système un niveau de complétude de la collection exigences qui soit satisfaisant?"

3.1. Problématique de la complétude

La complétude des exigences est donc cruciale: c'est à la fois l'une des qualités les plus difficiles à obtenir, et l'une des plus importantes du point de vue du bon fonctionnement du système développé.

Les causes de collections d'exigences incomplètes sont elles-mêmes très nombreuses. Par exemple le périmètre du projet peut être mal déterminé. Il peut y avoir eu un problème de compréhension (entre parties prenantes, d'une partie prenante par rapport au problème, ou d'une partie prenante par rapport au domaine et aux exigences implicites). Ou encore, les exigences ont pu évoluer dans le temps. Nous pensons qu'une démarche

méthodologique adaptée doit, quelles que soient les raisons pratiques qui peuvent amener à une incomplétude de la collection d'exigences, traiter les trois problèmes fondamentaux suivants

- (i) **Difficulté à dimensionner systématiquement la complétude:** deux types de complétude peuvent être définis, la complétude interne et la complétude externe. La complétude interne peut être mesurée par le nombre d'exigences non formalisées, ou laissées dans l'état "à définir". La complétude externe est liée au fait qu'aucune exigence ne manque dans la collection utilisée pour développer le système. La complétude interne est relativement facile à mesurer puisqu'il suffit de considérer les propriétés des exigences déjà identifiées. En revanche, il est extrêmement difficile de juger à partir de quand une collection d'exigence est complète du point de vue de la complétude externe. Le dimensionnement de la complétude est en particulier rendu difficile en raison de l'absence intrinsèque de la collection complète d'exigences.
- (ii) **Non-prévisibilité des situations de fonctionnement du système:** l'identification des exigences repose en grande partie sur l'exploration de l'environnement de fonctionnement du système. Ainsi, la classification des utilisateurs, la connaissance des autres systèmes avec lesquels celui développé devra interagir, ou la connaissance des environnements physiques de fonctionnement du système sont des éléments sur lesquels on s'appuie fréquemment pour découvrir les exigences. Or dans les faits, ces éléments présentent le plus souvent une variabilité qu'il est le plus souvent extrêmement difficile d'embrasser complètement en raison de l'étendue des possibilités qui s'offrent à l'imagination. Une approche de complétude d'exigences doit permettre de vérifier que la collection d'exigences adresse bien l'ensemble des situations de fonctionnement possibles du système.
- (iii) **Explosion combinatoire du nombre de cas non nominaux:** la finalité de l'ingénierie des exigences étant de définir les fonctionnalités du système, les exigences s'intéressent en particulier aux cas nominaux de fonctionnement du système. Pourtant les cas de fonctionnement non-nominaux, c'est à dire dans lesquels le système est mis en situation (volontairement ou involontairement) de ne pas satisfaire une ou plusieurs exigences sont tout aussi importants que les cas nominaux du point de vue de la satisfaction des exigences réelles des parties prenantes. Or, l'exploration de haut en bas d'exigences (non fonctionnelles ou de contraintes) dévoile une explosion combinatoire du nombre de cas non nominaux. Une approche de complétude adaptée doit aider à découvrir les exigences couvrant l'ensemble des cas non nominaux possibles de manière adaptée.

3.2. Principes de l'approche RITA

L'approche d'analyse de la complétude RITA prend le parti de se baser en particulier sur les cas non nominaux de fonctionnement du système développé; il s'agit donc d'une approche "par les contingences". Une hypothèse fondamentale est qu'un certain nombre d'exigences doit avoir déjà été spécifié pour pouvoir analyser ces cas non nominaux. Dans RITA, les données d'entrée sont une collection de fragments de besoins organisés hiérarchiquement conformément à la définition de *L'Écritoire*.

2.3.a Modélisation causale des menaces

RITA identifie les cas non nominaux de fonctionnement du futur système à l'aide du concept de menace. Une menace est une cause possible de non-satisfaction d'une exigence. Dans RITA, les exigences apparaissent sous la forme de modèles de buts que le système doit permettre de satisfaire. Le concept de menace est donc étroitement lié à celui d'obstacle, lui-même traditionnellement défini comme l'ensemble des conditions de non-satisfaction de but.

L'approche employée pour modéliser les menaces s'inscrit dans la famille des approches de modélisation causale. L'occurrence d'une menace peut être provoquée par l'apparition d'un phénomène individuel, ou par la conjonction de plusieurs phénomènes. Les divers phénomènes répertoriés sont définis comme les causes de menaces. Le modèle causal de menace de RITA modélise de manière identique les causes alternatives et complémentaires. Par ailleurs, le modèle causal des menaces est hiérarchisé. Les causes de menaces définies au niveau n de la hiérarchie contribuent à l'occurrence des causes du niveau $n-1$.

2.3.b Ontologies des causes de menaces et des stratégies de traitement

Les différentes causes de menaces rencontrées dans différents contextes font apparaître des caractéristiques communes et variables qui peuvent servir à les classer. RITA propose une ontologie des causes de menaces qui classe les causes de menaces à différents niveaux. L'ontologie sert non seulement pour la capitalisation, mais aussi pour la découverte et l'analyse des menaces.

A chaque cause de menace correspond une ou différentes stratégies de traitement. Une seconde ontologie est proposée par RITA pour classer les stratégies de traitement. Le catalogue de stratégies de traitements organisé dans cette seconde ontologie est lié à l'ontologie des causes de menaces de manière à pouvoir identifier de manière systématique quelle stratégie de traitement pourrait être considérée lorsqu'une cause de menace d'un type particulier est rencontré.

Complétude des exigences par la découverte et l'analyse causale des menaces

RITA propose une démarche méthodologique cyclique en trois étapes: identification, analyse et traitement des menaces. Le problème de la prévisibilité des situations de fonctionnement du système est abordé sous l'angle des obstacles à la manière de *L'Écritoire*. De la même manière que *L'Écritoire* permet de considérer l'espace imaginable des situations de fonctionnement du système en déclinant les buts du système, RITA suggère de considérer les obstacles coïncidant aux situations de fonctionnement limites, aux cas de satisfaction partielle et aux cas de non-satisfaction totale des buts connus. Ces cas sont découverts (a) en analysant la structure des buts auxquelles les exigences sont associées et en les déclinant leurs paramètres par un mécanisme de négation, et (b) directement à partir de l'ontologie des causes de menaces. Dans le cas (a), l'analyse causale est faite de manière ascendante (en remontant de la conséquence aux causes), dans le cas (b) elle est réalisée de manière montante (en partant des causes potentielles). Comme les buts produits par *L'Écritoire*, les menaces produites par RITA peuvent appartenir à différents niveaux d'abstraction.

RITA guide la découverte de nouvelles exigences en commençant par le diagnostique des causes de menaces. L'analyse causale est guidée systématiquement par l'ontologie de causes connues. Pour chaque menace, l'analyse

causale s'arrête lorsque toutes les causes possibles identifiées dans l'ontologie ont été considérées. Les nouvelles exigences coïncident à des stratégies de traitement des menaces identifiées et analysées. La recherche systématique des nouvelles exigences est guidée dans RITA par l'exploration d'une typologie de traitements associée à l'ontologie des menaces.

L'ontologie des causes de menaces a trois utilités. D'une part, elle permet d'identifier les causes de menaces. D'autre part, elle permet de sélectionner les stratégies de traitement réellement adaptées aux menaces considérées. Enfin, elle fournit un support pour vérifier que l'ensemble possible des menaces et causes de menaces a été considéré. La complétude des exigences découvertes est garantie (a) par l'exhaustivité des types de menaces identifiées dans l'ontologie, et (b) par l'identification de stratégies de traitement adaptées pour chaque menace considérée.

3.3. Résultats

Mes recherches sur le thème de la complétude des exigences par l'analyse de menaces ont fait l'objet du co-encadrement de la thèse d'Elena Ivankina avec Colette Rolland et Isabelle Wattiau [Ivankina2009]. Les résultats originaux de ces recherches sont

- Un modèle de spécification causale des obstacles ;
- Une ontologie de menaces ;
- Une technique de découverte des menaces ;
- Une typologie d'exigences réutilisables ;
- Une démarche méthodologique supportée par un prototype logiciel ;
- Une évaluation au moyen d'études de cas.

Ces travaux ont donné lieu à la publications de 3 articles dans des conférences internationales avec comité de programme [Ivankina2004a] (International Conference on Business Information Systems), [Ivankina2004b] (European-Japanese Conference on Information Modelling and Knowledge Bases), et international [Ivankina2005] (Software Engineering Research, Management and Applications), ainsi qu'à la publication d'un article au forum d'une conférence internationale [Ivankina2004c] (Conference on Advanced Information Systems Engineering Forum), et d'un article dans un atelier international [Salinesi2008a] (International Workshop on Managing Requirements Knowledge).

Dans la méthode RITA, les buts affectés au système sont définis avec des conditions de satisfaction. RITA propose de considérer les éléments participant à la non-satisfaction (complète ou partielle) de ces conditions [Ivankina2004a]. Chaque élément de non-satisfaction de condition identifie un obstacle. Tout obstacle peut avoir plusieurs causes de différentes natures et de différentes origines. RITA propose de guider l'analyse des causes d'obstacle au moyen d'un *modèle de spécification causale des obstacles*. Ce modèle est pré-structuré sous la forme d'un modèle de type en arrête de poisson à la manière des diagrammes d'Ishikawa. L'avantage de cette approche est de maîtriser l'explosion combinatoire du nombre de causes d'obstacles en les catégorisant systématiquement par classe d'origine. Chaque cause d'obstacle ainsi définie représente une menace pour le fonctionnement correcte du système et sert dans la méthode à chercher des exigences complémentaires.

Une multitude de menaces peut apparaître pour chaque obstacle. Il est donc difficile de les recenser de façon exhaustive. Afin de pallier cette carence nous proposons une *ontologie de menaces*. Cette ontologie, générique,

propose un ensemble de types génériques de menaces [Ivankina2005]. Les différents types de menaces présentées par cette ontologie sont organisés (a) à plusieurs niveaux d'abstraction et de détail (b) sémantiquement en fonction de l'origine des menaces et conformément à la structure prédéfinie du modèle de spécification causale des obstacles. L'ontologie de menaces ayant pour ambition de couvrir exhaustivement tout type de menace, et ce quel que soit le domaine, nous l'avons construite par capitalisation. Il est possible de l'enrichir, soit verticalement par exploration systématique d'un domaine, soit horizontalement par analyse de rapports d'analyse des défaillances de systèmes.

RITA propose quatre *techniques pour guider la découverte de menaces*: à partir des obstacles, par l'exploration de cas limites au moyen de questions "what-if", par affinement de menaces connues, et par les menaces déjà constatées [Ivankina2004b]. L'explosion combinatoire du nombre de cas considéré est maîtrisée par l'utilisation systématique de l'ontologie des menaces qui permet de classer systématiquement toute menace découverte par l'une de ces quatre techniques.

Le problème de complétude ouverte est ramené à un problème fermé si l'on émet l'hypothèse que la collection d'exigences est complète si elle garantit la prise en compte de l'ensemble des menaces connues. Dans RITA, les exigences sont découvertes à partir de chaque menace, en considérant l'ensemble des stratégies susceptibles de permettre de la traiter. RITA propose donc une *typologie de stratégies de traitements et d'exigences réutilisables* qui peuvent être utilisée pour résoudre les différentes classes de menaces connues [Ivankina2004b].

La *démarche méthodologique* RITA a été formalisée au moyen du formalisme de MAP développé au CRI [Ivankina2004c]. Le principal guidage concerne la découverte de besoins complémentaires formalisés par des buts au moyen de techniques d'analyse de menaces. Par ailleurs, RITA apporte un support pour guider son enrichissement par la capitalisation de nouveaux types de menaces dans l'ontologie, et de nouvelles exigences-type associées dans la typologie des stratégies de traitement. Un *prototype logiciel* a été développé pour démontrer la faisabilité de la démarche méthodologique.

La démarche méthodologique RITA a été mise en application dans le cadre d'une *étude de cas portant sur le problème de la disponibilité de guichets automatiques bancaires dans une grande banque française* [Salinesi2008a]. Nous avons pu calculer que la mise en œuvre de certaines des nombreuses nouvelles exigences découvertes au moyen de RITA a rapporté à la banque une économie de 140.000 €, et un gain de confort pour ses clients. L'étude de cas a non seulement confirmé la faisabilité de la démarche RITA, et sa scalabilité à des problèmes de taille industrielle, mais elle a permis de montrer que l'analyse des menaces par RITA est efficace pour atteindre l'objectif de complétude des exigences. Une seconde étude de cas, elle aussi menée dans le secteur bancaire, est présentée dans [Ivankina2009].

4. Priorisation de grands ensembles d'exigences structurés par les buts - l'approche NENO

De nombreux travaux de recherche en Ingénierie des exigences sont consacrés à la définition de nouvelles démarches méthodologiques dans lesquelles les décisions sont bien identifiées, mais pour lesquelles très peu de directives sont proposées pour prendre ces décisions. On peut en effet considérer a priori qu'il est toujours

possible d'adopter une technique d'aide à la décision généraliste. Dans les années 90, il a cependant été démontré que la prise de décisions en Ingénierie des Exigences soulève des problèmes spécifiques [Evans 1997]. Depuis lors, des méthodes spécifiques ont été proposées pour tenir compte de diverses situations de prise de décisions rencontrées en Ingénierie des Exigences, par exemple pour choisir des méthodes et des outils, pour sélectionner les parties prenantes, pour évaluer la faisabilité des exigences, ou pour trier les exigences à retenir lors du développement d'un système. Cette dernière activité, nommée priorisation des exigences prend une part particulièrement importante en ingénierie des exigences. En effet, du choix des exigences mises en oeuvre dépend non seulement la qualité du système produit mais aussi la difficulté et le coût de sa production.

Deux propriétés fondamentales des exigences justifient le besoin de les prioriser. Tout d'abord, les exigences sont -par essence- de nature "optative". C'est à dire qu'elles sont telles que l'on peut décider de les mettre en oeuvre dans le système. D'autre part chaque exigence peut être mise en oeuvre de différentes manières ou à différents moments du cycle de vie du système. Des choix doivent donc être faits. Enfin, l'ingénierie des exigences fait intervenir une multiplicité de parties prenantes. Celles -ci peuvent avoir des besoins contradictoires. Elles peuvent aussi avoir des opinions différentes sur le degré d'importance des exigences à mettre en oeuvre, et donc avoir des opinions différentes lorsque des choix doivent être faits.

Le point d'entrée du processus de priorisation est une collection d'exigences optionnelles considérées individuellement. Il peut aussi bien s'agir d'exigences fonctionnelles (fonction qu'un composant du système devrait fournir), ou non fonctionnelles (qualités contraignant la mise en oeuvre d'une fonction dans le système, telle que performance, disponibilité, ou accessibilité). On parle parfois d'alternatives pour désigner l'ensemble des combinaisons que peuvent former ces exigences optionnelles. Le résultat du processus de priorisation est aussi une collection d'exigences. La collection d'exigences résultant du processus de priorisation est un sous-ensemble de la collection des exigences à laquelle la priorisation s'est appliquée. Tout comme pour la collection d'exigences qui sert de point de départ, les exigences du résultat sont individuelles, sans lien de dépendance particulier. En principe, le résultat doit être une combinaison d'exigences coïncidant à l'une des alternatives considérées.

La priorisation doit être réalisée en tenant compte de différents *critères*; il s'agit donc d'un problème d'aide à la décision multicritères. Les critères sont les dimensions prises en compte pour établir une relation d'ordre entre exigences. Parmi les critères les plus fréquemment cités, on trouve: l'importance, le coût et le risque. Les parties prenantes formulent des *évaluations* en considérant les différentes exigences sous l'angle des différents critères. Selon l'approche, l'évaluation peut être quantitative (affectation d'une valeur absolue ou relative), ou qualitative (positionnement dans une échelle ou un groupe de valeurs). Les différents algorithmes disponibles dans la littérature synthétisent à partir des différentes évaluations une *priorité* associée à chaque exigence. La priorité peut consister en un choix (inclure ou ne pas inclure l'exigence dans la configuration du système à développer), ou une valeur numérique reflétant le degré d'importance globale de l'exigence pour le système à développer.

Une difficulté importante rencontrée par les démarches méthodologiques de priorisation existantes est due à la difficulté à traiter des problèmes de grande échelle. Or le besoin de démarches adaptées aux besoins de l'industrie reste très important. Qu'ils soient de nature stratégique, tactique ou opérationnelle, les arbitrages dégagés dans un processus de priorisation des exigences prennent toujours une place importante dans le succès des projets. Il

suffit qu'un arbitrage soit pris en amont sans que l'espace du problème soit entièrement exploré, qu'il ne tienne pas compte de tous les points de vue, que ses résultats ne soient pas transparents, rationnels, ou pire qu'ils ne soient pas cohérents, pour générer par la suite une insatisfaction des utilisateurs, une frustration des parties prenantes, un accroissement des coûts de développement, des problèmes de fonctionnement, une remise en cause des objectifs de l'organisation qui utilise le système, et l'apparition de résistances allant jusqu'à l'abandon du nouveau système. Ces constats nous ont amenés à soulever la question de recherche suivante:

"quelle est la démarche méthodologique de priorisation des exigences efficace pour effectuer des arbitrages dans les diverses situations complexes rencontrées en Ingénierie des Exigences?"

4.1. Problématique de la priorisation de grands ensembles d'exigences

Le contexte dans lequel se positionnent nos travaux est celui de la priorisation de collections d'exigences auxquelles les approches traditionnelles ne s'appliquent pas car trop simplistes (ranking, méthode des 100 dollars, pondération) ou génèrent un niveau de complexité trop important (par exemple comparaisons pair à pair). Nous parlons d'arbitrages plutôt que de priorisation des exigences telles que nous venons de le définir pour trois raisons: (a) les données d'entrée du processus d'arbitrage présentent des propriétés particulières (ensembles de très grande taille, présence de dépendances entre les exigences) (b) les critères d'évaluation à prendre en compte sont très variés, et enfin (c) le résultat du processus d'arbitrage est lui aussi de nature particulière (multi-projets). Huit problèmes spécifiques à l'arbitrage des exigences peuvent être distingués.

Le *point d'entrée du processus d'arbitrage* est une collection d'exigences. Trois problèmes spécifiques doivent être pris en compte à ce niveau: (i) la multiplicité des niveaux d'abstraction auxquels les exigences sont exprimées, (ii) l'effet levier, et (iii) les contraintes de configuration.

(i) *Multiplicité des niveaux d'abstraction*: les collections d'exigences que nous considérons en entrée sont de très grande taille (plusieurs milliers d'exigences). Il apparaît que dans ces circonstances les exigences auxquelles s'appliquent les arbitrages sont plus fréquemment à différents niveaux de détail et d'abstraction. Par exemple, il peut aussi bien s'agir d'exigences qui s'appliquent au niveau opérationnel du SI, que d'exigences globales qui concernent à la fois l'organisation et le SI, voir des objectifs stratégiques. Le processus d'arbitrage doit tenir compte du fait qu'il est difficile de comparer des exigences qui n'appartiennent pas au même niveau d'abstraction.

(ii) *Effet levier*: de nombreuses dépendances apparaissent le plus souvent entre les exigences à prioriser. Or, les dépendances entre exigences peuvent générer des corrélations du point de vue de leur priorité. On observe ainsi en pratique que le degré de priorité d'une exigence peut avoir une influence systématique sur le degré de priorité d'une autre exigence. Cela est par exemple observé dans le cas des exigences d'évolution car les différentes évolutions requises peuvent être synergiques ou au contraire diverger. Ce problème est aussi associé à la multiplicité des niveaux d'abstraction, en particulier pour ce qui est des exigences liées par un lien d'affinement. Le processus d'arbitrage doit tenir compte du problème de l'effet levier en assurant que les influences positives et négatives des variations du niveau de priorité de chaque exigence sur les autres sont gérées globalement de manière cohérente.

(iii) *Contraintes d'agencement*: les nombreux types de dépendances que l'on identifie au moment de l'analyse des exigences (par exemple de complémentarité, alternative, ou exclusion) peuvent être interprétés comme des contraintes qui régissent la manière de combiner les exigences. Ces contraintes introduisent, elles aussi, de la complexité dans l'arbitrage puisque chaque décision peut avoir des conséquences multiples par répercussion sur d'autres exigences. Le processus d'arbitrage doit tenir compte des contraintes d'agencement en supportant leur spécification, et l'analyse systématique de leur conformité.

Trois problèmes spécifiques apparaissent du point de vue des *critères d'évaluation à prendre en compte lors de l'arbitrage*: (i) la non-combinabilité et (ii) la complexité de l'interdépendance des critères, et (iii) le rétro-effet des exigences sur les critères.

(iv) *Non-combinabilité des critères*: il est possible, lorsque seuls deux ou trois critères sont pris en considération, de les combiner par des techniques simples de ratio ou de moyenne pondérée. C'est par exemple ce que proposent les approches coût-valeur. Pour que ce type d'approche soit appliqué, il faut que les critères combinés soient de nature à être combinés et que la combinaison soit valable pour toutes les exigences. Or ce n'est pas toujours le cas. Ainsi, plus le nombre de critères est important, moins on a de chance de pouvoir appliquer l'approche de la combinaison. La non-combinabilité peut être due à la nature des exigences. Elle peut être aussi causée par des situations complexes d'influences croisées entre exigences selon différents critères. Ainsi, il n'est pas possible de simplement combiner les critères lorsque l'importance d'une exigence relativement à un critère a une influence sur l'importance d'une autre exigence par rapport à un autre critère. Le processus d'arbitrage doit aborder la non-combinabilité des critères en tenant compte du degré spécifique d'importance de chaque critère dans la priorité finale de chaque exigence.

(v) *Complexité de l'interdépendance des critères*: l'importance relative de deux critères peut être variable selon la situation, les points de vue ou dans le temps. Ainsi, même lorsque les critères sont combinables, il arrivera que la combinaison (par exemple les poids relatifs des critères) dépende en fonction de l'un de ces facteurs. La complexité de l'interdépendance des critères doit être prise en compte par le processus d'arbitrage en particulier au travers des facteurs générant cette complexité.

(vi) *Rétro-influence des exigences sur les critères*: comme indiqué plus haut, les critères d'arbitrage peuvent eux-mêmes être des exigences. Or de la même manière que les arbitrages appliqués à certaines exigences peuvent influencer les arbitrages appliqués à d'autres exigences, on constate dans la pratique que les arbitrages appliqués à certaines exigences peuvent influencer le degré relatif d'importance de certains critères dans la définition d'autres exigences. Cette rétro-influence fait que le processus d'arbitrage ne peut être considéré sous l'angle simple d'une cascade de décisions (d'abord les critères, puis les exigences), mais doit être vu comme un processus complexe mettant en jeu des cycles de décision.

Deux problèmes spécifiques apparaissent au niveau de la *nature des produits de l'arbitrage*: (i) projets multiples (ii) scénarios multiples.

(vii) *Projets multiples*: outre le fait que la même exigence peut revêtir différents niveaux de priorité selon le critère, il apparaît que pour le même critère, la même exigence pourra avoir différentes priorités selon le contexte de projet. Or les évolutions de systèmes d'information se réalisent en pratique par le biais de projets multiples qui

seront menés en parallèle. Ainsi, une exigence jugée peu importante pour un projet pourra être très importante pour un autre. Une exigence jugée peu prioritaire sur le plan d'un projet unique (par exemple en raison de son coût) pourra devenir importante au regard d'un ensemble de projets (le coût est partagé). Le processus d'arbitrage devra tenir compte du fait que les arbitrages sont à mener pour chaque projet d'un portefeuille de projets, sachant que la multiplicité des projets peut avoir une incidence sur la priorité des exigences dans chaque projet individuel.

(viii) *Scénarios multiples*: l'arbitrage se distingue de la priorisation au sens où la décision à prendre est double: il s'agit non seulement de décider de l'ensemble des exigences qui seront mises en œuvre, mais aussi du scénario de leur mise en œuvre. Le produit de l'arbitrage n'est donc pas une simple collection plate d'exigences. Il faut aussi avoir défini l'ordre dans lequel elles devront être mises en œuvre, par exemple en terme de répartition dans des projets successifs.

4.2. *Principes de l'approche NENO*

Afin de mieux répondre aux attentes pratiques en terme de scalabilité, et de réalisabilité du processus d'arbitrage, nous avons décidé de fonder notre approche sur une double hypothèse (a) il est plus intéressant de raisonner sur des critères d'évaluation non-prédéfinis, et par ailleurs (b) il faut chercher à mener plusieurs évaluations, chaque évaluation étant appliquée à un ensemble réduit d'alternatives.

L'approche d'arbitrage proposée produit une collection cohérente d'exigences individuelles, entre lesquelles des relations sont explicitement définies. Cette collection d'exigences spécifie le système que les parties prenantes décident communément de développer, ainsi que le déroulement de ce développement.

Pour ce faire, notre approche d'arbitrage aide à la construction de scénarios de systèmes cibles. Ces scénarios sont spécifiés par des collections d'exigences qui sont des sous-ensembles de l'ensemble des exigences de départ. Les scénarios spécifient des systèmes alternatifs cibles. Chaque scénario est réaliste dans la mesure où (a) il respecte un certain nombre de règles d'assemblage que l'approche aide à produire, et (b) toute affectation d'exigence au scénario est conforme aux choix des parties prenantes. Enfin, l'approche aide à définir les critères d'évaluation qui aident à la construction des scénarios et au calcul final des priorités appliqué aux scénarios alternatifs.

Comme le montre la Figure 3, le problème de l'arbitrage est donc décomposé en trois sous-problèmes distincts: celui de la définition de *critères d'évaluation*, celui de la construction de *scénarios alternatifs*, et celui du calcul des *priorités finales*.

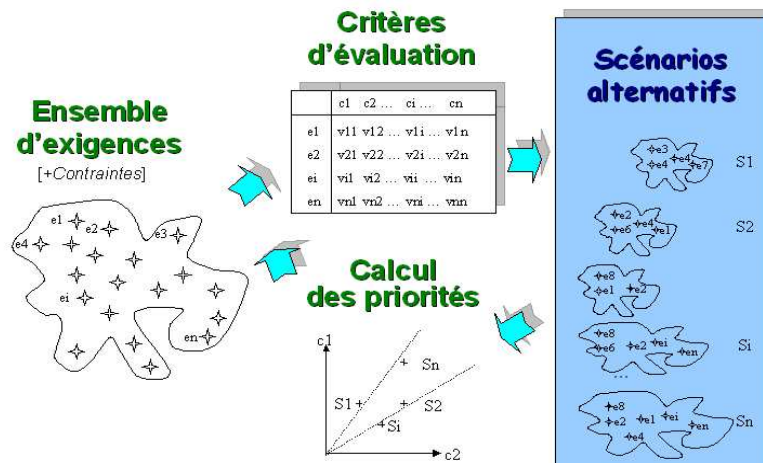


Figure 3: Aperçu de la méthode NENO

Les principes saillants de la méthode NENO sont: (a) la définition des critères d'arbitrage au moyen de NFR, (b) le clustering des exigences en scénarios d'évolution au moyen du méta modèle MAP, (c) l'étude quantitative des impacts sur l'atteinte d'objectifs sous la forme d'agrégation d'évaluations (d) le calcul des priorités par l'évaluation qualitative, participative, et quantitative des scénarios d'évolution, et (e) la représentation multi-formes des priorités.

2.4.a Définition des critères d'arbitrage au moyen de NFRs

Le choix d'appliquer le problème de la priorisation à de grands ensembles d'exigences rend critique la définition des scénarios alternatifs. Notre approche cherche un juste milieu entre la production exhaustive de toutes les combinaisons possibles d'exigences -qui rendrait encore plus critique le problème de scalabilité- et la production d'un ensemble réduit de scénarios totalement prédéfini -qui soulèverait des questions de fiabilité au moment du calcul des priorités finales car les évaluations des éléments constituant les alternatives seraient automatiquement liées.

Notre solution exploite les propriétés de la collection d'exigences initiale. En premier lieu, la distinction exigences fonctionnelles / exigences non fonctionnelles (NFR) est utilisée de manière particulière: les scénarios alternatifs sont constitués uniquement d'exigences fonctionnelles présentes dans la collection initiale d'exigences. Les exigences non fonctionnelles sont utilisées pour définir les critères d'évaluation.

L'ensemble des exigences non fonctionnelles est représenté sous la forme de buts non fonctionnels qui sont rassemblés dans un *modèle d'objectifs*. C'est ce dernier qui, représentant les objectifs à long terme de l'entreprise et les liant aux exigences non fonctionnelles permet de définir les critères sur lesquels reposeront les évaluations et donc les arbitrages. Dans NENO le modèle d'objectif est défini au moyen d'une partie du formalisme de I* identifiant les buts non fonctionnels, ainsi que leurs liens de dépendance et de contribution.

2.4.b Clustering de la collection d'exigences en scénarios d'évolution alternatifs au moyen du modèle MAP

Une seconde caractéristique clef de la méthode NENO est de présenter les scénarios d'évolution sous la forme d'une Carte qui instancie le méta-modèle intentionnel MAP (ou d'une hiérarchie de Cartes s'il est nécessaire d'affiner l'évaluation). Ainsi, chacun des scénarios alternatifs est exprimé sous une forme intentionnelle qui met l'accent sur l'enchaînement des buts à atteindre et surtout sur les manières de les atteindre. Les manières d'atteindre chacun des buts sont un facteur important puisque que nous sommes convaincus que pour qu'une méthode soit adaptée à l'arbitrage, il est important d'identifier les similarités et les écarts entre chacune des alternatives offertes aux participants. Les Cartes possèdent de multiples avantages par rapport à d'autres méthodes d'arbitrage proposant d'autres formalismes.

Elles rendent explicites et compréhensibles pour tous les participants le contenu de chaque alternative sans devoir réaliser des efforts d'apprentissage de formalisme et réduisent les ambiguïtés.

Elles permettent aux participants de mettre en exergue les points communs et surtout les différences entre chacune des alternatives et ceci au niveau majeur par l'ajout/suppression d'intentions, et au niveau mineur par l'ajout/suppression de stratégies. La collection d'exigences initiale est donc non seulement abstraite au moyen des intentions et des stratégies, mais aussi "clusterisée" en fonction des scénarios d'évolution défini par combinaison des différents buts et stratégies.

Enfin, elles permettent de présenter l'ensemble des scénarios possibles sous la forme d'une seule et même Carte. Ce formalisme peut s'avérer très utile en particulier lorsque les différences entre les scénarios sont mineurs et que le nombre d'alternatives est réduit (2 ou 3 maximum).

Par ailleurs, les scénarios alternatifs sont définis à plusieurs *niveaux d'abstraction*. A chaque niveau d'abstraction, les scénarios sont définis conformément à des contraintes d'agencement issues des liens identifiés entre les exigences de la collection initiale.

2.4.c Etude quantitative des impacts sur l'atteinte d'objectifs sous la forme d'agrégation d'évaluations

L'étude quantitative des impacts de chaque scénario sur les objectifs est réalisée via une approche basée sur l'agrégation d'évaluations de fragments. Dans la méthode NENO, les fragments sont les sections d'une Carte (ie des triplets <intention source, intention cible, stratégie>). Nous exploitons ainsi la propriété d'assemblage des Cartes selon laquelle chaque section peut appartenir à une ou plusieurs Cartes qui elles-mêmes représentent chacune un scénario possible.

L'exploitation de ce mécanisme est fondamentale. La méthode NENO pose l'hypothèse que l'évaluation finale sera plus précise en agrégeant des évaluations de sections plutôt que de demander directement aux participants d'évaluer une Carte entière. Cette hypothèse est basée sur les éléments suivants:

Un participant donné n'a pas forcément les compétences nécessaires lui permettant d'évaluer l'ensemble du périmètre d'un scénario. Cette propriété est particulièrement importante dans le cadre d'une étude d'USI qui couvre un périmètre métier très large.

L'évaluation d'un participant sera beaucoup plus précise sur une section que sur la totalité d'une Carte.

2.4.d Calcul des priorités par l'évaluation qualitative, participative, et quantitative des scénarios d'évolution

Le procédé d'évaluation est d'abord qualitatif et participatif puis ensuite quantitatif. Le résultat est exprimé à la suite d'un calcul sous forme de priorités exprimées grâce à des métriques.

Etant donné le temps imparti pour les études dans le cadre desquelles sont effectuées les arbitrages (quelques semaines, au mieux quelques mois), il n'est pas envisageable de réaliser des études quantitatives afin de mesurer l'évaluation la plus fine de chacun des scénarios relativement à chacun des critères identifiés (exemple: réduction des coûts, augmentation de la marge commerciale, ...). Quelque fois, il n'est même pas du tout possible de réaliser de telles études.

C'est pour cette raison que NENO propose une approche par évaluation d'experts. Cette approche est couramment utilisée dans les secteurs industriel et financier pour étudier des perspectives. L'objectif est de recueillir auprès de référents métiers leurs évaluations et de chercher à obtenir un consensus entre ces individus. Pour ce faire NENO exploite un mécanisme de recherche de consensus basé sur la méthode Delphi.

NENO est complétée par des algorithmes permettant de traduire les évaluations qualitatives en résultats quantitatifs. C'est cette étape qui permet d'agrèger l'ensemble des évaluations et de proposer aux décideurs une priorité unique pour chacun des scénarios.

2.4.e Représentation multi-formes des priorités

L'objectif final est de fournir aux décideurs un outil qui leur permet de prendre la décision la plus pertinente possible. NENO propose plusieurs modalités de restitutions qui permettent, suivant le nombre de scénarios alternatifs et de critères de décision, de proposer un formalisme de restitution adapté à la situation précise.

4.3. Résultats

Plusieurs résultats ont été produits à l'issue de mes travaux sur le thème de la priorisation en ingénierie des exigences.

- Un état de l'art structuré au moyen d'un cadre de classification multi-dimensionnel;
- Une technique guidant spécifiquement le choix des critères de priorisation;
- Une technique de spécification de scénarios d'évolution alternatifs qui exploite le méta-modèle intentionnel MAP;
- Une méthode, appelée NENO, qui exploite un ensemble d'algorithmes permettant, à partir d'évaluations qualitatives, de calculer des priorités affectées à chacune des scénarios alternatifs considérés;
- Une évaluation expérimentale par un cas industriel démontrant l'applicabilité de la méthode NENO dans un environnement grandeur nature;

- Une méthode pour la sélection de techniques d'aide à la décision multicritère dans le contexte des processus d'ingénierie de systèmes d'information.

J'ai encadré deux thèses dans le cadre de ces travaux. Celle d'Emmanuel Papadacci [Papadacci2008], soutenue en Mai 2008, menée dans le cadre d'une collaboration avec la société Renault, et celle d'Elena Kornyshova, en cours. Ces travaux ont fait l'objet de la publication plusieurs articles.

4 articles de revues ont été produits: [Papadacci2006b] (Revue des sciences et techniques de l'information), [Kornyshova2008e] (INSIGHT), [Kornyshova2008a] (International Journal of Information Technology and Intelligent Computing), [Kornyshova2009] (Systems Research Forum Journal).

J'ai par ailleurs publié 4 articles dans des conférences internationales avec comité de lecture: [Papadacci2006a] (Information and Communication Technologies: from Theory to Applications), [Kornyshova2007b] (International Conference on Enterprise Information Systems), et [Kornyshova2007d] (Working Conference on Situational Method Engineering).

Enfin, divers autres articles dans des ateliers et forums tels que [Kornyshova2008b] (International Conference on Advanced information Systems Engineering Forum), [Kornyshova2008c] (International Workshop on Model Driven Information Systems Engineering: Enterprise, User and System Models), [Papadacci2004a] (Requirements Engineering: Foundation for Software Quality), ou [Papadacci2005b] (Ingénierie et gestion des processus d'entreprise).

Tout comme pour nos travaux sur les scénarios, nous avons réalisé une étude bibliographique des démarches méthodologiques de priorisation des exigences. De ce travail de recherche bibliographique résulte un *état de l'art structuré au moyen d'un cadre de classification multidimensionnel*. Quatre dimensions ont été définies: l'objet, le formalisme, le contexte et le processus qui posent chacune une question sur les démarches considérées: "qu'est-ce qui est priorisé?", "sous quelle forme est exprimé la priorisation?", "pourquoi utiliser la démarche de priorisation?", et "comment le processus est-il réalisé?". Diverses facettes sont définies dans chaque dimension afin de répondre aux questions posées de manière aussi exhaustive que possible. Dix approches, choisies en raison de leur représentativité de l'ensemble des démarches méthodologiques de priorisation existantes, sont choisies et classifiées à l'aide du cadre.

Dans le domaine même des démarches méthodologiques, notre contribution est constituée de deux éléments principaux: (a) une approche nouvelle de la priorisation qui considère les exigences non pas individuellement, mais comme des ensembles complexes, structurés et de grande taille, et (b) une approche nouvelle des critères de priorisation.

Concernant les critères de priorisation, nous avons démontré qu'il était possible de surmonter les problèmes de non combinabilité, de dépendances complexes, et de retro-effets des décisions sur les critères. Pour ce faire, j'ai proposé une *technique guidant spécifiquement le choix des critères de priorisation* [Papadacci2005a]. Dans notre approche, les critères de priorisation d'exigences fonctionnelles sont des exigences non fonctionnelles. Par ailleurs, toutes les exigences fonctionnelles ne sont pas considérées simultanément. Une première activité de priorisation est donc appliquée pour définir les critères qui serviront eux-mêmes à prioriser les exigences fonctionnelles. Cette priorisation est répétée à chaque fois qu'apparaît une nouvelle situation de priorisation.

Je me suis intéressé au calcul des priorités appliqués à des scénarios d'évolution de systèmes d'information définis par des assemblages spécifiques d'exigences plutôt que sur des exigences individuelles. Ces scénarios sont définis à différents niveaux d'abstraction, ce qui permet de dé-corréler les évaluations des différentes exigences— une évaluation faite à un niveau n n'engage pas à une évaluation sur laquelle on ne pourrait plus revenir au niveau plus détaillé n-1. Un second élément dans le travail de priorisation est donc l'affectation de chaque exigence aux scénarios qui sont construits. La solution que je propose est une *technique de spécification de scénarios d'évolution alternatifs qui exploite le méta-modèle intentionnel MAP* [Papadacci2005b].

Les éléments de scénarios qui appartiennent à différents niveaux d'abstraction sont liés de manière à guider progressivement les parties prenantes dans des évaluations de plus en plus fines au moment du calcul final des priorités. Ainsi lorsqu'une évaluation est faite à un niveau n, certaines exigences sont écartées par respect des contraintes d'agencement et en fonction des résultats des algorithmes de calcul des priorités. D'autres évaluations et calcul de priorités restent cependant possibles au niveau plus détaillé n-1. Chaque évaluation peut être effectuée à l'aide d'un nombre réduit de critères spécifiques, résolvant ainsi le problème de la complexité de l'interdépendance des critères et celui de leur non combinabilité. Chaque calcul de priorité ainsi effectué à un niveau n permet d'entrer dans un nouveau cycle d'arbitrage au niveau n-1. En particulier, de nouveaux critères doivent être définis spécifiquement pour les exigences identifiées comme devant être considérées au niveau n-1, supportant ainsi la *rétro influence des exigences sur les critères*.

Nous avons spécifié une *méthode, appelée NENO, qui exploite un ensemble d'algorithmes permettant, à partir d'évaluations qualitatives, de calculer des priorités affectées à chacune des scénarios alternatifs considérés*. La méthode NENO a été pensée pour guider en particulier l'arbitrage d'exigences d'évolution de systèmes d'information dans le contexte d'études d'urbanisation des Systèmes d'Information USI. Cette méthode est décrite par un modèle de processus et un modèle de produit définis formellement [Papadacci2006].

Par ailleurs, nous avons mené *une évaluation expérimentale par un cas industriel démontrant l'applicabilité de la méthode NENO dans un environnement grandeur nature* [Papadacci2005b]. Cette évaluation a non seulement permis de faire apparaître les qualités de notre approche, mais aussi de démontrer la validité de nos hypothèses de travail. En outre, l'utilisation expérimentale, mais non moins concrète, de NENO au sein de la direction de l'USI du groupe Renault a permis de montrer que l'application de NENO avait pour effet:

- d'aider à construire une représentation synthétique et argumentée des alternatives, des critères de décision et des évaluations qui aident concrètement les décideurs à prendre les décisions qui s'imposent en termes d'évolutions de leur SI,
- d'aider à poser clairement et formellement les questions et sortir d'un sentiment d'informalité de la prise de décisions. Ces décisions peuvent être ainsi actées et la documentation associée peut être réutilisée pour justifier les choix budgétaires et organisationnels indispensables à la mise en œuvre de l'évolution, et
- de raccourcir la durée de prise de décisions. Le fait de disposer d'un processus d'arbitrage bien défini permet de le planifier et de le réaliser dans un temps souvent plus court en particulier lorsque le consensus n'est pas facilement atteignable.

Nous avons en outre pu observer que les démarches traditionnelles n'auraient pas permis de faire face à la quantité d'exigences considérées dans l'étude de cas, ni de résoudre certains problèmes spécifiques de l'urbanisation du système d'information, comme par exemple la définition d'un portefeuille de projets au lieu d'un système individuel.

Un problème important est l'intégration de la démarche méthodologique de priorisation NENO dans des démarches méthodologiques généralistes d'Ingénierie des Exigences. L'étude de cas que nous avons réalisée dans le contexte d'une démarche d'Urbanisation de Système d'Information nous a montré d'une part l'importance de cette intégration, mais aussi que celle-ci pouvait être requise dans des situations très différentes. Etant donné la difficulté à prévoir toutes ces situations, nous avons adopté une approche situationnelle d'Ingénierie des Méthodes. Je propose une *méthode pour la sélection de techniques d'aide à la décision multicritère dans le contexte des processus d'ingénierie de systèmes d'information*. Cette méthode distingue deux niveaux de décision: la décision d'ingénierie, et le choix de la méthode à mettre en oeuvre pour prendre cette décision. Notre contribution est donc double [Kornysheva2007a], [Kornysheva2007b], [Kornysheva2007c], [Kornysheva2007d], [Kornysheva2008]. Du point de vue de l'Ingénierie des Exigences, nous menons une catégorisation systématique des situations de décision et des méthodes les plus adaptées. Du point de vue de l'Ingénierie des méthodes, nous participons au développement d'une approche nouvelle d'assemblage situationnel de fragments méthodes.

5. Perspectives de recherche

L'élucidation des exigences « from scratch » a longtemps été l'objet central des différentes recherches menées dans le domaine de l'ingénierie des exigences. Deux concepts ont pris une place particulièrement importante dans la modélisation des exigences: les buts et les scénarios. En ce qui concerne les scénarios, la question de leur utilisation dans les entreprises reste posée. En particulier, plus de dix ans après les études de terrain menées dans le contexte du projet CREWS, on ne sait pas dire quelles sont les approches les mieux adaptées aux problématiques industrielles actuelles pour rédiger les scénarios et les analyser. J'ai ainsi récemment pu observer dans le cadre de ma participation aux travaux de l'Association Française d'Ingénierie des Systèmes et à l'occasion d'une étude sur l'emploi des scénarios au sein de la société Renault que de nouvelles formes de scénarios apparaissent. Par ailleurs, mon expérience avec le développement de l'outil REGAL de partage de bonnes pratiques en Ingénierie des Exigences, utilisé par l'INCOSE pour le partage de bonnes pratiques en ingénierie des exigences, m'a montré que la culture des scénarios en entreprise a considérablement évolué ces dix dernières années. Il est à mon avis temps de mener de nouvelles études empiriques afin de collecter des données à jour concernant le niveau actuel de maîtrise des concepts d'ingénierie des exigences, et les nouvelles attentes des entreprises à l'égard des méthodes techniques et outils exploitant ces concepts. Un exemple concret concerne la problématique de l'innovation dans les sociétés du secteur des services. Au cours d'une série d'interviews que j'ai récemment menées pour analyser les nouvelles compétences requises par les entreprises de service, j'ai pu observer l'importance que présente l'activité d'analyse des besoins pour mieux innover dans ce secteur. Le Directeur des Systèmes d'Information, le Directeur des Technologies ainsi que les membres du Comité Exécutif de l'entreprise doivent être capables de sentir les besoins des clients et de se représenter comment l'entreprise peut satisfaire ces besoins, indépendamment des systèmes informatiques et des systèmes d'information existants. De nouvelles méthodes, techniques et outils, me paraissent nécessaires. Celles-ci doivent être adaptés à ce type de parties prenantes habituées à travailler par « coup d'œil », sur des prises de décision rapides prises à partir de descriptions à la fois extrêmement synthétiques et convaincantes.

Bien que longtemps décriée en raison de leurs limites, les outils d'analyse linguistiques reviennent régulièrement sur le terrain de l'ingénierie des exigences. L'idée de recherches sur l'analyse d'exigences rédigées en langage

naturel au moyen d'outils linguistiques n'est pas récente. Les principales perspectives concernent l'industrialisation des résultats déjà produits par leur incorporation dans des outils de l'industrie, par exemple pour supporter des activités de rédaction et de vérification de cahiers des charges. Le projet Par ailleurs, l'apparition de nouvelles théories et de nouveaux outils d'analyse linguistique offrent régulièrement de nouvelles opportunités de recherche. De même l'apparition de nouveaux meta-modèles en ingénierie des exigences soulève de manière récurrente la question de leur production à partir de descriptions textuelles. Je compte étendre mon expérience dans le domaine de la production de directives de rédaction, de vérification, de correction et de conceptualisation en m'intéressant à de nouveaux meta modèles et en m'appuyant sur de nouveaux outils d'analyse linguistique.

La question de la priorisation des exigences pose une difficulté liée à la mise en pratique de techniques trop complexes au regard des attentes des parties prenantes. Mon expérience dans le cadre de la collaboration avec la société Renault m'a montré que celles-ci veulent en effet participer à des ateliers de prise de décision rapides. Les parties prenantes sont aussi très méfiantes à l'égard des techniques automatisées et attendent de la transparence dans les calculs mis en oeuvre. Les perspectives de recherche relèvent autant du choix des techniques adaptées aux circonstances que de la définition de nouvelles approches. Les nombreux travaux en théorie de la décision montrent que de nouvelles approches de priorisation d'exigences peuvent encore être proposées. La question du choix d'une technique dans le panel des techniques existantes est un enjeu proportionnel à l'importance des conséquences des décisions. Or, on ne sait aujourd'hui pas encore prédire de manière fiable le degré d'efficacité d'une technique dans un contexte donné. Les travaux à mener dans ce domaine sont théoriques et empiriques. Sur le plan théorique, il faut savoir caractériser les différentes techniques de manière très fine (notamment beaucoup plus fine que ce que proposent les états de l'art). Sur le plan empirique, il faut rassembler les données qui attestent de l'efficacité des différentes techniques dans différents contextes.

Enfin, je suis convaincu que de nouveaux outils pour stimuler la créativité sont nécessaires. La production d'heuristiques nouvelles, reposant éventuellement sur des concepts issus d'autres domaines comme cela a été le cas avec les ontologies ou les techniques d'ingénierie dirigée par les modèles offrent à cet égard des perspectives à mon avis très intéressantes.

CHAPITRE 3

Découverte des exigences par mise en coïncidence ("matching")

1. Introduction et positionnement de mes travaux sur la mise en coïncidence

La pratique de l'ingénierie des exigences fait apparaître de nombreuses situations dans lesquelles plusieurs modèles conceptuels doivent être manipulés simultanément. C'est par exemple le cas lorsque l'on doit tenir compte de points de vue multiples [Easterbrook1991], [Sabetzadeh2005] lors d'opérations de transformation de modèles [Goguen2002], dans le cadre d'une activité de réutilisation [Spanoudakis1993], ou lors de la mise en place d'un système adaptable [Soffer2005]. Ces différentes situations nécessitent d'identifier les éléments qui coïncident entre les modèles manipulés. On parle alors de "mise en coïncidence" entre modèles. De manière générale, l'établissement de coïncidences entre modèles résulte en une série de relations (établies à priori ou à posteriori) entre les éléments des modèles. Ces relations permettent de découvrir une série d'exigences. D'un point de vue pratique, les approches connues de mise en coïncidence trouvent leurs limites dans les situations suivantes:

- Tout d'abord, lorsque le même phénomène est modélisé de manières différentes. La situation la plus courante est celle d'emploi de termes différents pour désigner le même phénomène. Certains langages de conception ou méta-modèles offrent des concepts redondants (par exemple dans UML, un cas d'utilisation peut être une méthode de classe). Cela ouvre la possibilité de représenter le même phénomène de différentes manières. Enfin, il est fréquent que les choix de conception amènent à envisager (et donc produire) différentes spécifications en fonction de raisons extérieures aux phénomènes décrits dans les modèles eux-même.
- Par ailleurs, on rencontre une barrière de la complexité dans la recherche de coïncidences lorsque la taille et le nombre des modèles considérés sont importants. De manière naïve, la recherche de coïncidences entre une paire de modèles contenant respectivement n et m éléments sera de complexité quadratique puisque $n*m$ coïncidences possibles doivent être considérées. Par extension l'application de ce type d'approche à une large collection de modèles de grande taille génère une explosion combinatoire du nombre de coïncidences à considérer, chaque élément de chaque modèle étant susceptible de coïncider à n'importe lequel de tous les éléments de tous les autres modèles.
- Enfin, l'application de certaines approches de mise en coïncidence peut s'avérer difficile voir impossible lorsque les modèles considérésinstancient des méta-modèles différents. Les différences entre méta-modèles peuvent parfois être faibles. C'est par exemple le cas d'un certain nombre de méta-modèles pour la spécification de lignes de produits adaptés du méta-modèle FODA. Cependant dans les faits, lorsque l'on manipule plusieurs méta-modèles, ceux-ci sont le plus souvent très différents. En effet, les

phénomènes qui doivent être modélisés sont variés et il n'existe pas de méta-modèle qui les couvre tous d'un bloc⁴. C'est par exemple le cas lorsque différents domaines d'ingénierie interviennent, ou lorsque les parties prenantes se concentrent sur des facettes différentes des mêmes systèmes.

Nous nous sommes tout particulièrement intéressés au cas de la mise en place de systèmes adaptables, qu'il s'agisse de Lignes de Produits (LP), de composants disponibles sur l'étagère (COTS), de services, ou de Progiciels de Gestion Intégrés (PGI). Comme de nombreux auteurs l'ont observé (par exemple [Arsanjani2001], [Soffer2003] ou [Garlan1995]), la principale difficulté de la mise en coïncidence dans ce type de projet tient à la disparité des langages dans lesquels les parties en présence, d'une part les experts du système, et les experts métiers de l'organisation d'autre part, ont l'habitude de s'exprimer. On parle alors de « discordance conceptuelle ». Les deux parties en présence dans ce type de projet emploient chacun leur documentation, conservent leur propre langage, et rencontrent des difficultés à se faire comprendre de l'autre partie. Le risque principal de cette incompréhension mutuelle est de produire une adaptation de système qui ne coïncide pas aux besoins réels des futurs utilisateurs. Cela peut conduire à un échec total du projet [Esteves2001].

La description des fonctionnalités d'un système adaptable se focalise sur le « quoi » et le « comment » de ses opérations ; par exemple, sur les données qui doivent être maintenues/fournies et actions qui doivent être effectuées. La mise en coïncidence de ces fonctionnalités avec les besoins de l'organisation n'est pas une tâche simple. Les experts du système l'obtiennent souvent en forçant l'organisation à aligner ses besoins aux capacités du système.

D'un autre côté, l'organisation qui souhaite mettre en place un système adaptable possède un certain nombre d'objectifs et de stratégies qui justifient les investissements. Les acteurs de l'organisation voudraient savoir comment le système pourra répondre à leurs objectifs et coïncider à leurs stratégies. Pour cette raison, ils souhaitent faire une mise en coïncidence au niveau haut (niveau intentionnel) entre leurs objectifs et les objectifs que le système pourrait permettre de satisfaire. L'expression des objectifs et des stratégies est intentionnelle, contrairement aux fonctions de systèmes qui sont exprimées de manière opérationnelle.

La différence de niveau d'abstraction utilisé par l'organisation (intentionnel) et le système adaptable (opérationnel) pose un problème lors de la mise en place du système. En fait, au niveau opérationnel, il est difficile de comprendre comment répondre aux besoins de l'organisation pour deux raisons:

- (a) la quantité de détails à manipuler est grande et la maîtriser est difficile ;
- (b) les organisations pensent en termes de buts et d'objectifs et non pas en termes de fonctionnalités du système alors que c'est l'inverse pour les experts du système.

4 De ce point de vue, UML ou SySML apparaissent plus comme des collections de méta-modèles que comme des assemblages de méta-modèles intégrés de manière profonde.

Les experts métier s'expriment en objectifs, buts à atteindre et stratégies pour le faire. Ils sont à un niveau intentionnel d'expression de leurs problèmes. Au contraire les experts du système sont dans leur monde technique ; ils s'expriment au niveau des solutions, de manière opérationnelle. Une facette de la discordance conceptuelle est donc liée au fait que les deux parties prenantes ne s'expriment pas au même niveau d'abstraction.

Il y a aussi discordance dans la forme de l'expression. En effet, le langage des experts d'entreprise est concis et informel. Un document « stratégique » résume en quelques pages les situations auxquelles on doit faire face. A l'inverse, le langage de l'éditeur de système adaptable est technique. Il se base sur les concepts et modèles de base de données, d'interface, de collaboration entre objets, d'accès, etc.

Toutes ces différences rendent le problème de la mise en coïncidence difficile voir impossible à résoudre de manière systématique si les deux expressions en présence sont dans une telle situation de discordance. Notre position est que la solution nécessite une meilleure homogénéité dans l'expression des besoins de l'organisation et des fonctionnalités du système.

2. Positionnement

Un point commun fondamental des travaux cités au début de ce chapitre ([Easterbrook1991], [Sabetzadeh2005], [Goguen2002], [Spanoudakis1993], [Soffer2005]) est qu'ils reposent sur une approche qui consiste à chercher des morphismes⁵ entre des modèles formels très détaillés.

Or, d'une part, les systèmes auxquels nous sommes confrontés qu'ils soient logiciels, d'information ou d'entreprise sont de plus en plus complexes du point de vue de leur structure et de leur comportement. D'autre part, il est reconnu que les mécanismes d'abstraction sont un des rares moyens d'aide à la maîtrise de la complexité. Nous pensons que *le paradigme intentionnel* est un outil puissant de représentation et de raisonnement qui permet de traiter des problèmes d'ingénierie de systèmes complexes par des mécanismes d'abstraction appropriés au domaine d'investigation du CRI. L'abstraction intentionnelle consiste à échapper aux détails du monde opérationnel (par exemple celui des processus d'entreprise et de la description détaillée de qui fait quoi, quand et avec quelles ressources) en énonçant les problèmes en termes intentionnels, c'est-à-dire dirigés par les buts à atteindre sans dire par quelles procédures détaillées les atteindre.

L'approche que nous avons retenue pour la mise en coïncidence des systèmes adaptables avec les besoins des organisations dans lesquels ils sont mis en place est la *mise en coïncidence au niveau intentionnel*. Cette approche a été retenue pour les raisons suivantes.

- (i) Cette approche est compatible avec les approches connues d'ingénierie des exigences. Les langages orientés « but » sont utilisés dans les approches d'ingénierie des exigences où ils ont montré leur utilité

5 Soient R et S deux relations binaires respectivement sur E et F, et f: E → F une application de E dans F. f est un morphisme de (E,R) dans (F, S) si et seulement si: $\forall (x, y) \text{ dans } E^2, xRy \Rightarrow f(x)Sf(y)$

[Potts1994], tout comme dans le domaine de la refonte des processus métier (BPR) [Yu1994]. Dans le processus d'ingénierie des exigences, les langages intentionnels (orientés but) ont montré qu'ils permettaient un meilleur alignement des exigences du système aux besoins de l'organisation. Dans le BPR, ils guident la refonte des processus métier. On peut donc penser que les approches orientées but peuvent aider à résoudre le problème (b) ci-dessus. En évitant les détails inutiles, ces approches aident à se focaliser sur ce qui doit être réalisé et sur les stratégies nécessaires pour cela ; ceci contribue aussi à la résolution du problème (a).

- (ii) Elle établit un équilibre entre le système et l'entreprise qui est le client et prend tous les risques. Cela est par exemple vital dans des projets dans lesquels la seule partie de l'équipe projet qui soit experte dans le système à adapter est externe à l'entreprise. Grâce à l'approche choisie, la responsabilité des décisions peut être partagée puisque les enjeux système sont mieux compris et maîtrisés par les experts métier.
- (iii) Elle établit un équilibre entre le fournisseur du système adaptable et le client qui, compte tenu de la taille des enjeux, nous semble salutaire. Etant forcé d'employer un langage compréhensible par les experts métier, le fournisseur n'est plus le seul capable d'aborder le projet dans la perspective du système. Les solutions qu'il propose peuvent ainsi être négociées par les experts métier. Le fournisseur est amené à changer d'attitude soit en démontrant plus systématiquement la capacité des solutions qu'il propose à répondre aux besoins du client, soit en associant plus étroitement les experts métier à l'élaboration de ces solutions.
- (iv) Le CRI a développé un langage, intitulé MAP, qui repose sur le couplage intention-stratégie et un mécanisme d'abstraction. MAP s'appuie sur un méta-modèle permettant de représenter un problème par un ordonnancement flexible d'intentions à atteindre (buts) et de stratégies pour le faire (approches pour atteindre les intentions). Le mécanisme d'abstraction permet d'énoncer les couples intentions-stratégies à différents niveaux d'abstraction. L'emploi du langage MAP permet d'appréhender la notion d'objectif ou de but du système à adapter de manière systématique. Un but du système exprime la finalité des tâches pour lesquelles une ou plusieurs des fonctionnalités du système adaptable sont utiles, tout en abstrayant la définition de ces tâches et de leur réalisation. Ce faisant, on met l'emphase sur le propos du système, son but. Cela aide à sélectionner les fonctions du système qui rencontrent les buts du système.

L'alternative retenue nécessite de positionner la description du système à adapter au niveau intentionnel alors que les documentations de systèmes adaptables (qu'il s'agisse de COTS, d'ERP, de services web, ou de lignes de produit) sont en général au niveau opérationnel ou technique. Le travail d'abstraction à réaliser doit permettre de répondre systématiquement à la question « à quels besoins cette/ces fonctionnalités du système permettent-elles de répondre ? ». Ce travail supplémentaire peut être assimilé à une activité de rétro-modélisation. Il repose sur l'expertise du système à adapter et peut exploiter une documentation existante, éventuellement déjà spécifiée sous la forme de modèles structurés.

Cette rétro-modélisation doit être effectuée par un expert du système et doit être faite pour chaque nouveau système considéré. En revanche, une fois modélisés pour un système donné, les buts de ce système restent

relativement stables. Les modèles de but de chaque système sont donc acquis pour les projets ultérieurs et peuvent être réutilisés moyennant des modifications mineures. Le surplus de travail nécessaire pour la rétro-modélisation d'un système n'est donc pas nécessairement pénalisant.

Nous avons décidé de considérer deux aspects particuliers dans le problème de la mise en coïncidence des modèles de système et des besoins: d'une part la recherche de similarités entre les modèles mis en coïncidence, et d'autre part l'établissement de coïncidences par analyse de la cohérence des modèles à un niveau global.

Ces deux séries de travaux sont respectivement présentées dans les sections 2 et 3 ci-après.

3. Mise en coïncidence par analyse de similarités – l'approche MIBE

La majorité des systèmes installés aujourd'hui ne le sont plus à partir d'une spécification d'exigences spécifiques mais par adaptation d'un système de type PGI. Le développement 'à partir de rien' est devenu une exception. L'ingénierie des exigences doit donc s'adapter à ce nouveau type d'ingénierie. Notre hypothèse de recherche est que le problème se pose en termes de recherche de coïncidence ('matching') entre l'ensemble des exigences de l'organisation et l'ensemble des exigences que le progiciel est susceptible de satisfaire. En fait les problèmes se posent de manière similaire dans diverses situations telles que l'évolution d'un système, la configuration de progiciel, l'assemblage de composants ou la configuration de ligne de produits. La particularité de notre approche de résolution de la coïncidence est de poser l'hypothèse que l'on peut réduire les problèmes soit à une évaluation des similarités soit à l'identification d'écarts. Nous nous sommes donc intéressés à la définition de techniques d'analyse de similarité qui soient adaptées à la résolution du problème de mise en coïncidence.

L'utilité de techniques d'analyse de similarité a été démontrée dans différents contextes de l'ingénierie des SI et en ingénierie des exigences. Par exemple, des techniques de mesure de similarité ont été proposées pour évaluer la réutilisabilité des composants de schémas conceptuels [Castano1992] et pour sélectionner ces composants [Jilani1997]. [Bianco1999] proposent des métriques de similarité pour l'analyse des schémas de bases de données hétérogènes. Plusieurs autres approches de mesure de similarité ont été proposées pour aider à l'indexation [Diamantini1999], au regroupement et à la sélection de données similaires [Papadopoulos1999]. Des métriques de similarités ont été appliquées à la recherche d'éléments dans des textes [Besançon1999]. Dans le domaine de la re-ingénierie des SI, des mesures de distance ont été employées pour évaluer les changements de modèles d'entreprise [Poels2000]. [NattOchDag2001] propose d'utiliser des techniques automatiques d'analyse de similarité pour regrouper des demandes de changement et identifier les exigences à mettre en oeuvre dans différentes versions d'un système.

L'analyse de similarité est un élément clef dans la résolution du problème de mise en coïncidence. Il s'agit d'établir de manière semi-automatique, voir entièrement automatique le rapprochement entre les éléments des différents modèles mis en coïncidence. Une préoccupation importante est la qualité des résultats obtenus, sachant que les modèles mis en coïncidence peuvent différer grandement du point de vue de leur structure, des découpages qu'ils présentent, du niveau d'abstraction auquel ils ont été définis, des termes choisis pour labelliser les éléments qu'ils contiennent, etc. La question de recherche soulevée est donc la suivante:

"Comment automatiser l'analyse de similarités dans le contexte de la mise en coïncidence de modèles tout en préservant la qualité des résultats?"

Une *similarité* exprime une ressemblance entre deux éléments provenant de modèles différents. La ressemblance peut porter sur différents *aspects* des éléments comparés. Un *type de similarité* peut donc être défini pour chaque aspect comparé. L'ensemble des types de similarités constitue une *typologie*. Chaque type de similarité est défini dans la typologie par une fonction qui: (i) précise au travers des arguments le type des éléments comparés et les aspects pris en compte dans la comparaison, et (ii) définit le type de la valeur recherchée (numérique ou logique). Les divers types de similarité peuvent être évalués au moyen de mesures de similarité. Une *mesure de similarité* est donc associée à chaque type de similarité. Les mesures s'emploient avec les modèles qui sont mis en coïncidence. Chaque mesure utilise une formule qui: (i) s'applique aux arguments indiqués par la fonction associée au type de similarité, et (ii) calcule une valeur dont le type est compatible avec celui recherché, comme défini par la fonction associée au type de similarité.

Différents types de similarités peuvent être définis selon le contexte et l'utilisation. La diversité est importante: certaines approches permettent de comparer des textes, d'autres tiennent compte des liens qui sont définis dans des modèles, certaines approches tiennent compte du méta-modèle instancié d'autres pas, etc. Dans chaque cas, la démarche consiste à développer une typologie de similarités proposant un ou deux types de similarités spécifiques aux modèles comparés, avec les mesures associées. La principale préoccupation pour répondre à notre question de recherche est l'élaboration d'une typologie de similarités adaptée.

3.1. *Problématique dans l'élaboration d'une typologie de similarités*

La qualité des similarités mesurées dépend en premier lieu de la qualité de la typologie de similarités sur laquelle repose cette mesure. Exploiter une typologie de similarités spécifique aux modèles comparés présente trois inconvénients. D'une part, cela ne permet pas d'assurer la qualité de la typologie employée. D'autre part cela ne permet pas d'assurer la qualité de chacune des mesures qui la compose. Enfin, les typologies spécifiques définissent des mesures difficiles à transposer et dont la qualité doit à chaque fois être re-démontrée. Nous avons recensé huit problèmes-clefs dans l'élaboration d'une typologie de similarité de qualité. Quatre d'entre eux sont d'ordre global et s'appliquent à la typologie à développer dans son ensemble. Les quatre autres sont d'ordre individuel et doivent être considérés pour l'élaboration de chaque type de similarité dans la typologie. Ces problèmes sont:

- i. l'incomplétude,
- ii. la pauvreté sémantique,
- iii. l'incohérence,
- iv. l'inadaptabilité,
- v. l'incorrection,
- vi. l'absence de généralité,
- vii. la difficulté à automatiser, et

viii. le manque de profondeur.

D'un point de vue global, une typologie de similarités doit satisfaire des qualités de complétude, de richesse sémantique, de cohérence et d'adaptabilité. Les problèmes rencontrés lorsque ces différentes qualités ne sont pas satisfaites sont les suivants.

i. Incomplétude: une typologie est complète si elle permet d'exprimer tout type de similarité pouvant exister pour un type de modèle donné. Inversement, une typologie de similarité n'est pas complète s'il existe au moins une paire de modèles pour laquelle une similarité qui ne peut être produite par aucun type de similarité de la typologie. Dans ce dernier cas, la mise en coïncidence réalisée au moyen de la typologie de similarités est elle-même incomplète. Le résultat de la mise en coïncidence risque en conséquence de n'être que partielle.

ii. Pauvreté sémantique: une typologie est sémantiquement riche (respectivement pauvre) si elle (ne) permet (resp. ne permet pas) d'exprimer toutes les similarités désirées sans combiner plusieurs types de similarités. La pauvreté sémantique n'est pas identique à l'incomplétude dans la mesure où l'absence d'un type de similarité est palliée par la combinaison. Cependant, l'emploi d'une typologie complète mais sémantiquement pauvre provoque une explosion combinatoire du nombre de combinaisons à considérer pour aboutir à une analyse de similarité exhaustive.

iii. Incohérence: lorsque la même similarité peut être produite par deux types de similarités différents de la même typologie, celle-ci est incohérente. A l'inverse, une typologie est cohérente si elle ne contient pas de types de similarités redondants. En pratique, l'emploi d'une typologie de similarité incohérente génère une augmentation du travail (redondant) de mise en coïncidence. L'incohérence, tout comme la pauvreté sémantique pose donc un problème de scalabilité.

iv. Inadaptabilité: une typologie de similarité est inadaptable lorsqu'elle ne peut être employée que pour des modèles qui instancient un couple de méta-modèles fixe. En pratique, l'inadaptabilité d'une typologie de similarité empêche son exploitation dans les environnements dans lesquels différents types de méta-modèles sont employés, et éventuellement adaptés en fonction des circonstances. Il est à l'inverse intéressant que la typologie puisse être adaptée à différents types de modèles au moyen de mécanismes systématiques.

Sur le plan individuel, les qualités que doit revêtir chaque type de similarité sont: correction, généralité, automatisabilité, et profondeur. Les problèmes rencontrés lorsque ces qualités ne sont pas satisfaites sont les suivants:

v. Incorrection: un type de similarité est incorrect s'il n'est pas conforme à la définition des éléments comparés et des modèles auxquels ils appartiennent. L'incorrection d'un type de similarité peut entraîner une difficulté à automatiser la métrique qui lui est associée, et à rendre l'interprétation de ses résultats arbitraire.

vi. Absence de généralité: un type de similarité n'est pas générique si la mesure qui lui est associée ne peut être appliquée à différents types d'éléments issus de différents types de modèles. L'absence de généralité d'un type de similarité rend son exploitation difficile. A chaque fois que de nouveaux méta-modèles sont employés, le type de similarité doit être re-défini de manière ad-hoc.

vii. Difficulté à automatiser: un type de similarité est difficile à automatiser lorsque la mise en oeuvre de la métrique qui lui est associée nécessite un environnement technique non standard. A l'inverse, l'évaluation des mesures doit pouvoir se faire de manière réaliste dans un environnement technique standard.

viii. Manque de profondeur: un type de similarité manque de profondeur lorsque sa définition ne tient pas compte de la sémantique des éléments comparés et de leur situation dans les modèles auxquels ils appartiennent. La production de mesures de similarités de qualité nécessite l'emploi de types de similarités qui satisfont la qualité de profondeur. A l'inverse, une mesure basée sur un type de similarité qui manque de profondeur sera elle-même superficielle.

3.2. Principes de l'approche MIBE

Comme le montre la Figure 4, la mise en coïncidence est effectuée en partant de l'analyse des similarités entre ces modèles intentionnels. La mise en coïncidence produit un modèle intentionnel qui établit la coïncidence entre les éléments des modèles considérés initialement. Lorsque des coïncidences y sont définies, cela signifie que le système est capable de satisfaire les exigences de l'utilisateur. L'introduction d'éléments supplémentaires dans ce modèle est indicative soit du besoin de satisfaire des exigences de l'utilisateur avec des éléments qui doivent être développés dans le système, soit à l'inverse que des éléments du système sont adoptés (par nécessité ou par opportunisme) alors qu'ils ne faisaient pas parties des éléments initialement requis.

Le cadre de travail dans lequel nous nous positionnons est présenté à la Figure 4. Conformément au cadre de travail de M. Jarke [Jarke1994], notre cadre se positionne dans un contexte d'évolution dans lequel l'organisation passe d'une situation présente à une situation future.

La situation présente est caractérisée par les modèles *As-Wished BM* (Business Modèle) et *Might-Be SFM* (System Functionality Model) qui représentent les besoins métier de l'organisation et les fonctionnalités offertes par l'ERP respectivement, avant le projet.

La situation future est caractérisée par les modèles *To-Be BM* et *To-Be SFM* représentant la situation du métier et du système respectivement, après le projet.

La Figure 4 montre que la mise en coïncidence s'établit entre modèles exprimés dans le langage unique MAP.

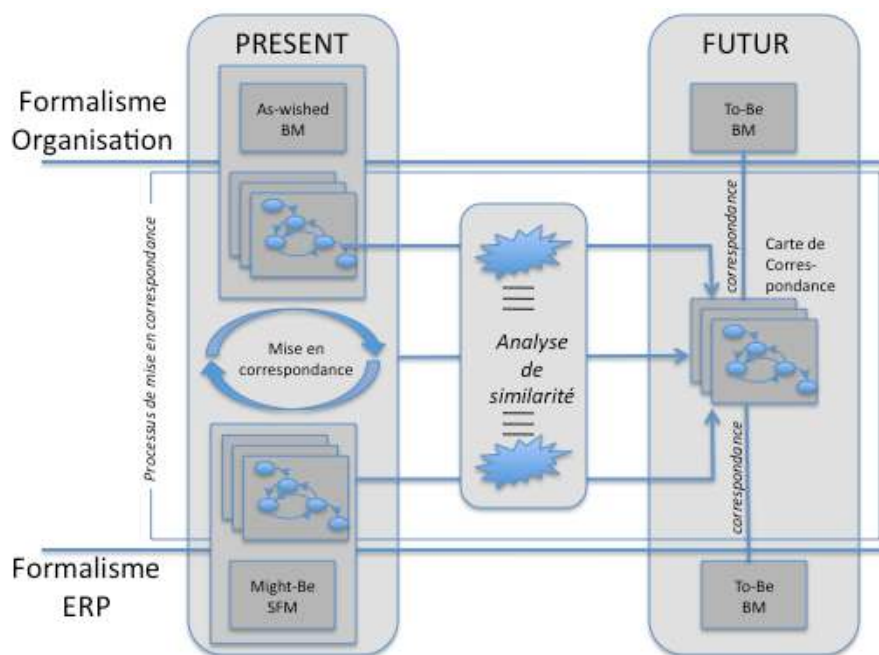


Figure 4: Approche MIBE

Nous utilisons le langage MAP pour représenter les modèles As-Wished BM et Might-Be SFM qui sont l'entrée du processus de mise en coïncidence. Les modèles spécifiés au moyen du langage MAP sont appelés *cartes*.

A partir des cartes produites nous réalisons leur *mise en coïncidence*. Pour cela nous utilisons des techniques d'*analyse des similarités* entre les cartes de l'organisation et celles de l'ERP. En fonction des différentes similarités détectées, une carte commune, appelée *carte de coïncidence*, est construite. Cette carte représente les besoins de l'organisation qui peuvent être satisfaits par des fonctionnalités du système et les exigences auxquelles le système ne peut pas répondre mais que l'organisation souhaite prendre en compte dans le système futur, même si cela nécessite des personnalisations importantes du système. Elle identifie aussi des fonctionnalités additionnelles du système que l'organisation décide d'adopter même si ces fonctionnalités ne coïncident pas à ses besoins initiaux.

A partir de la carte de coïncidence, les modèles cibles du côté organisation (To-Be BM) et du côté système (To-Be SFM) peuvent être produits en utilisant le formalisme retenu par l'entreprise. Dans ce cas, on peut être sûr qu'il existe une coïncidence entre ces deux modèles puisqu'ils sont issus de la même carte. Exprimer ces derniers modèles avec d'autres formalismes que le formalisme des cartes MAP peut parfois coïncider aux exigences de l'équipe projet de ne pas influencer ou remplacer le développement des modèles existants préconisés pour décrire la situation actuelle et/ou la situation cible.

2.3.a Typologie générique des similarités

Le procédé que nous avons employé pour recenser les types de similarité repose sur une approche systématique. L'approche a été celle de la méta-modélisation. Tout d'abord, un méta-modèle générique a été défini pour

spécifier de manière formelle les modèles devant être mis en coïncidence et les éléments qu'ils comportent (liens, non-liens, complexes, simples). Une typologie générique a été établie en application du cadre de Cavano et McCall. Le méta-modèle nous a permis d'identifier deux facteurs (le *facteur intrinsèque* et le *facteur structurel*), et quatre critères (*synonymie*, *hyponymie / hyperonymie*, *compositionnel*, *relationnel*). Différents types de similarité (30 en tout comme le montre le tableau ci-après) ont été définis pour chaque facteur puis pour chaque critère en conformité avec le méta-modèle. Enfin, des mécanismes d'adaptation sont proposés pour passer du niveau générique auquel le méta-modèle et la typologie ont été initialement proposés au niveau spécifique, qui est défini par les méta-modèles réellement employés pour développer les modèles mis en coïncidence.

Facteurs	Critères	Types	Description
Intrinsèque	Synonymie	TIS1	Deux éléments ont des propriétés dont les noms sont identiques
		TIS2	Deux éléments ont des propriétés dont les noms sont semblables
		TIS3	Deux éléments ont des propriétés dont les noms sont proches
		TIS4	Deux éléments ont des propriétés dont les valeurs sont identiques
		TIS5	Deux éléments ont des propriétés dont les valeurs sont semblables
		TIS6	Deux éléments ont des propriétés dont les valeurs sont proches
	Hyponymie Hyperonymie	TIH1	Deux éléments ont des propriétés dont les noms sont hyponymes
		TIH2	Deux éléments ont des propriétés dont les noms sont hyperonymes
Structurel	Compositionnel	TIH3	Deux éléments ont des propriétés dont les valeurs sont hyponymes
		TIH4	Deux éléments ont des propriétés dont les valeurs sont hyperonymes
		TSC1	Deux éléments composés ont la même cardinalité
		TSC2	Deux éléments composés sont identiques
		TSC3	Deux éléments composés sont semblables
		TSC4	Deux éléments composés sont proches
	TSC5	Deux éléments composés sont hyponymes	
	TSC6	Deux éléments composés sont hyperonymes	
	Relationnel	TSR1	Deux éléments non-liens sont cibles du même nombre de liens
		TSR2	Deux éléments non-liens sont sources du même nombre de liens
		TSR3	Deux éléments liens ont des éléments sources identiques
		TSR4	Deux éléments liens ont des éléments sources semblables
		TSR5	Deux éléments liens ont des éléments sources proches
		TSR6	Deux éléments liens ont des éléments sources hyponymes
		TSR7	Deux éléments liens ont des éléments sources hyperonymes
		TSR8	Deux éléments liens ont des éléments cibles identiques
		TSR9	Deux éléments liens ont des éléments cibles semblables
		TSR10	Deux éléments liens ont des éléments cibles proches
		TSR11	Deux éléments liens ont des éléments cibles hyponymes
		TSR12	Deux éléments liens ont des éléments cibles hyperonymes
TSR13		Deux éléments ont la même profondeur	
TSR14		Deux éléments ont la même hauteur	

Figure 5: Typologie générique de similarités

Tous les types de similarité ne s'appliquent pas à tous les éléments du méta-modèle générique. Ainsi, les types TIS1 et TIH1 s'appliquent à tout type d'éléments (lien, non-lien, simple et complexes). Le type TSC1 ne s'applique qu'aux éléments complexes, etc.

2.3.b Catalogue de métriques génériques et adaptables

Conformément à l'extension (facteur – critère – métrique) du cadre de Cavano et McCall [Cavano1988] qui introduit des types génériques de similarité, les mesures de similarité proposées en complément de chaque type de similarité sont définies au moyen de *métriques*.

Une mesure est associée à chaque type générique de similarité. A l'inverse, plusieurs mesures peuvent exploiter la même métrique. Nous n'utilisons pas directement les coefficients de Dice, Jaccard ou Cosine pour évaluer les

différents types de similarité. Cependant, les formules qui les définissent ont inspiré la définition de plusieurs des métriques proposées. Ce sont donc des versions modifiées des formules des coefficients de Dice Jaccard et Cosine qui apparaissent au travers de plusieurs des métriques que nous proposons (telles que IDE, SEM, PRO, HYPO, HYPERO, etc). Les métriques définies par les trois formules modifiées aident à calculer la similarité entre deux éléments (A et B) exprimés sous forme textuelle. Alors que les formules initiales des coefficients de Dice, Jaccard et Cosine ne considèrent que les termes *exactement identiques* entre deux textes, nous proposons de tenir compte des relations sémantiques profondes. Ainsi, les métriques définies par les coefficients Dice modifié, Jaccard modifié et Cosine modifié exploitent la notion de synonymie, d'hyponymie et d'hyperonymie. Les métriques que nous proposons sont les suivantes:

- Les métriques SIM, IDE, SEM et PRO permettent de calculer, au moyen de Dice, Jaccard ou Cosine modifié, la similarité (respectivement l'identité, la ressemblance, et la proximité) entre propriétés d'éléments ou, lorsque ces propriétés sont textuelles, à des termes apparaissant dans ces propriétés (par exemple un nom). Dans ces formule, différents degrés d'importance peuvent être accordés à la synonymie et à l'hyperonymie par rapport à l'identité.
- Les métriques HYPO et HYPERO s'appliquent à deux éléments (e_1 , e_2) et rendent un résultat « vrai » si e_1 est « hyponyme » (respectivement hyperonyme) de e_2 et un résultat « faux » s'ils ne le sont pas.
- La métrique TAUX_CAR (CAR pour « cardinalité ») permet d'évaluer le type de similarité TSC1 en calculant la proportion entre la cardinalité de deux éléments complexes.
- Les métrique IDE_P, SEM_P, PRO_P, HYPO_P, HYPERO_P (« P » pour « Pondéré ») s'appliquent à deux éléments complexes et rendent une valeur « vrai » s'ils sont identiques (respectivement semblables, proches, hyponymes ou hyperonymes), c'est à dire constitués d'éléments eux même identiques (respectivement semblables, proches, ou en relation d'hyponymie ou d'hyperonymie),
- Les métrique TAUX_NLE et TAUX_NLS (NLE pour « Nombre de Liens Entrants », NLS pour « Nombre de Liens Sortants ») s'appliquent à des éléments non liens pour évaluer la proportion du nombre de liens dont les éléments comparés sont la cible (respectivement la source).
- La métrique TAUX_PROF (respectivement TAUX_HAUT) permet de comparer le niveau de profondeur pour atteindre les éléments comparés à partir de la racine des hiérarchies auxquelles ils appartiennent respectivement (respectivement des hiérarchies dont ils sont la racine).

Les métriques que nous proposons sont de nature générique. Elles peuvent être utilisées avec différents langages employés pour construire les modèles, par exemple le modèle de Cas d'Utilisation, le modèle Entité-Relation, le langage MAP, la modélisation I*, KAOS, un langage de modélisation de processus tel que celui employé dans l'outil ARIS, ou même le langage naturel. Des typologies spécifiques de similarités peuvent être définies à partir de nos métriques de similarités génériques par un mécanisme d'instanciation et d'adaptation qui utilise la spécification du méta modèle spécifique visé dans les terme de notre méta modèle générique.

Par exemple, si l'on utilise le modèle de Cas d'Utilisation pour documenter les besoins d'une entreprise et les services offerts par un ERP, alors les exigences de conformité de l'ERP aux besoins sont définies au niveau modèle part des valeurs de similarité entre modèles de Cas d'Utilisation. Ces similarités et les mesures qui leurs sont associées peuvent être utilisées, par exemple, pour identifier la conformité des composants de l'ERP. La typologie spécifique de similarités peut permettre d'évaluer la conformité telle que « le système doit offrir des services à l'acteur X » qui sont exprimées sous la forme de « *deux acteurs X et Y ont des propriétés dont la valeur est identique* ». Ce type de similarité est défini avec la mesure associée par adaptation de la typologie générique de similarité en conformité avec le méta-modèle spécifique de Cas d'Utilisation.

Exploitation des métriques de similarité dans une démarche de mise en coïncidence top-down

Notre processus de mise en coïncidence est présenté en Figure 6 en application au cas de la mise en coïncidence entre les besoins d'une organisation et ceux qu'un ERP (Enterprise Resource Planning ou Progiciel de Gestion Intégré) est capable de satisfaire. L'activité centrale est la mise en coïncidence. Elle prend en entrée les modèles dits « Might-Be » et « As-Wished », et produit en sortie des modèles « To-Be ». Comme le montre la Figure, le processus de mise en coïncidence consiste à :

- Abstraire à partir des fonctionnalités existantes du système adaptable l'ensemble des besoins et exigences qui y sont incorporés et les représenter dans les modèles *Might-Be*.
- Décrire les besoins de l'organisation dans les modèles *As-Wished*.
- Réaliser la mise en coïncidence entre l'organisation et le système au niveau intentionnel en utilisant l'analyse des similarités et proposer des alternatives de coïncidence. Enfin choisir l'alternative de coïncidence qui satisfait le mieux l'organisation et dériver les adaptations et extensions des fonctionnalités de système exigées au moyen de modèles *To-Be*.

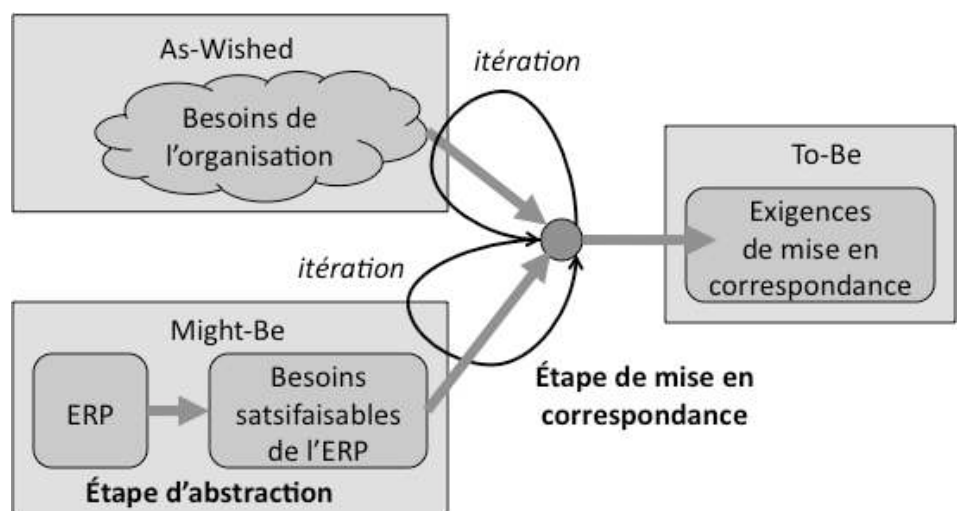


Figure 6: Aperçu du processus de mise en coïncidence

Le processus de mise en coïncidence est un processus cyclique. Il peut y avoir plusieurs raisons pour entamer un cycle dans l'analyse des coïncidences. D'une part l'information apportée par les coïncidences identifiées peut être

jugée insuffisante. Le problème est alors d'affiner l'analyse des coïncidences. Un tel affinement peut aller jusqu'au niveau de description opérationnelle où les modèles mis en coïncidence coïncident à des fonctionnalités du système. Inversement, l'itération peut avoir lieu pour définir des coïncidences complexes à partir de coïncidences élémentaires. Enfin, on peut réaliser des itérations lorsque l'on juge que les modèles à partir desquels les coïncidences sont définies sont trop peu riches. Il s'agit alors d'affiner les modèles mis en coïncidence pour trouver de nouvelles coïncidences.

3.3. Résultats

Le recours à la modélisation pour traiter le problème de mise en coïncidence est l'une des caractéristiques principales de nos travaux. Partant d'une grande quantité d'informations relatives à l'organisation et au système choisi, les modèles permettent de formaliser et d'abstraire les informations utiles en laissant de côté les détails encombrants.

L'utilisation d'un modèle unique pour exprimer à la fois les besoins de l'organisation et les fonctionnalités du système facilite leur mise en coïncidence et résout le problème de discordance conceptuelle.

Quatre résultats principaux ont été produits dans le cadre de nos travaux sur la mise en coïncidence par analyse de similarité:

- un cadre de travail,
- une typologie de similarités,
- un processus méthodologique pour la mise en coïncidence,
- une méthode, intitulée MIBE pour la personnalisation d'ERP, et
- une expérimentation de la méthode MIBE par mise en oeuvre dans un projet d'intégration d'ERP.

Les travaux de recherche que j'ai menés sur le thème de la mise en coïncidence d'exigences par analyse de similarité ont fait l'objet de la co-direction de la thèse des thèses de Iyad Zoukar, et de Ramzi Bouzid. Ces travaux ont aussi fait l'objet d'une collaboration avec la SNCF dans le contexte du projet ELAN d'installation de l'ERP PeopleSoft pour le support de la chaîne logistique. Cette collaboration a permis de mener une recherche action au cours de laquelle la méthode MIBE a pu être élaborée, puis d'une étude de cas dans le cadre duquel une application complète a pu être réalisée. Enfin, j'ai pu expérimenter plus partiellement la méthode dans le cadre d'une mise en application dont l'objet était la personnalisation de l'ERP NAVISIO pour les besoins de l'association caritative Dr Souris.

J'ai publié 6 articles dans des conférences internationales avec comité de programme: [Salinesi2004b] (International Conference on Systems Engineering), [Salinesi2004c] (International Conference on Advanced information Systems Engineering), [Zoukar2004a] (International Conference on Enterprise Information Systems), [Zoukar2004c] (Agent-Oriented Information Systems), [Zoukar2004d] (Information Resources Management Association), et [Salinesi2007b] (International Enterprise Distributed Object Computing Conference) ainsi qu'un article de conférence nationale [Zoukar2004b] (Informatique des Organisations et Systèmes d'Information et de Décision).

Le développement de l'approche MIBE résulte en premier lieu de la mise en place d'un *cadre de travail qui fait apparaître l'analyse de similarité comme un élément central de la résolution du problème de mise en coïncidence*. Conformément aux objectifs que nous nous sommes fixés, ce cadre de travail fait apparaître un niveau intermédiaire au sein duquel les modèles mis en coïncidence sont représentés dans le paradigme intentionnel [Zoukar2004a], [Salinesi2004c], [Zoukar2004b]. Cette approche offre une manière originale de résoudre le problème de discordance conceptuelle [Zoukar2004c].

Le second résultat significatif de notre recherche réside dans l'élaboration d'une *typologie de similarités générique et adaptable* au méta-modèle employé dans les modèles entre lesquels la similarité est recherchée. L'approche consistant à partir d'un niveau générique est adaptée de [Rolland2004]. Cette approche nous permet d'être indépendant des langages éventuellement déjà employés au sein des organisations ou pour modéliser les systèmes [Salinesi2004b], [Zoukar2004a]. Nous avons associé à chaque type générique de similarité une mesure de similarité qui permet de l'évaluer. Les mesures de similarité sont définies formellement au moyen de métriques inspirées de l'état de l'art. Cette démarche a été appliquée au langage MAP, ce qui a permis de définir systématiquement la typologie spécifique de similarités coïncidante.

Nous pensons que la typologie spécifique de similarités ainsi produite revêt les qualités initialement recherchées: complétude, richesse sémantiquement et cohérence.

- La complétude de la typologie spécifique proposée pour les cartes est directement liée à la complétude de la typologie générique. Notre état de l'art montre qu'aucune des approches existantes ne propose autant de types de similarités. Les types de similarités produits ont par ailleurs, tous été utilisés dans le cadre industriel de notre projet à la SNCF.
- La richesse sémantique n'est pas démontrée mais semble se confirmer empiriquement. L'expérience n'a pas fait émerger le besoin d'utiliser des combinaisons de types et mesures de similarité ; chaque type de similarité a démontré sa propre utilité de manière individuelle.
- De la même manière, nous avons pu observer de manière empirique que chaque besoin de similarité pouvait être exprimé individuellement au moyen d'un type de similarité. La typologie semble donc être effectivement cohérente.

Par ailleurs, nous pensons que sur le plan individuel, chaque type de similarité de la typologie est correct, automatisable et profond.

- Chaque type de similarité est correct dans la mesure où il a été défini conformément au méta-modèle de produit spécifique. La conformité est apparente au travers des liens explicitement définis entre le méta-modèle spécifique, le méta-modèle générique, et les métriques qui s'y appliquent.
- L'évaluation de chacun des types de similarité a été rendue automatisable par une définition systématique au moyen de formules reposant sur des métriques toutes définies d'après le méta-modèle générique.
- La profondeur des types de similarités est liée à l'exploration systématique des quatre facteurs au moyen des coefficients de Dice, Jaccard et Cosine modifiés en y introduisant des éléments d'analyse sémantique. Il est à noter que l'aspect profond de l'analyse se limite à la sémantique individuelle des termes. De

nombreux linguistes ont cependant démontré qu'une analyse sémantique pouvait être réalisée au niveau de la phrase. Nous n'avons pas choisi d'utiliser les outils existants de marquage sémantique ou d'analyse sémantique des textes car ils introduisent une complexité supplémentaire pour un apport qui n'est pas encore clairement identifié. Une étude de cette approche serait cependant intéressante.

Notre troisième résultat significatif réside dans une démarche méthodologique nommée MIBE. *La méthode MIBE guide la découverte et l'analyse des exigences pour la personnalisation d'un ERP*. Un modèle de processus, nommé MAP-MIBE, spécifie cette démarche méthodologique sous la forme d'une carte [Zoukar2004d]. Une large partie de ce processus guide la mise en coïncidence entre les besoins d'une organisation, modélisés par des cartes intentionnelles métier, et les fonctionnalités d'un ERP, exprimées par des cartes système. La mise en coïncidence est basée sur l'analyse de similarités entre les cartes métier et système. Cette analyse guide la production d'un ensemble d'alternatives de coïncidence et aide à en choisir une. L'alternative de coïncidence choisie exprime les besoins pour la personnalisation de l'ERP.

Malgré la très large adoption des ERP par les entreprises, il y a peu de travaux de recherche académiques sur les ERP dans une perspective d'ingénierie [Esteves2001]. Les discussions avec les participants de différents projets nous ont pourtant confirmé l'importance des problèmes évoqués ci-dessus, et le manque d'attention qui leur sont apportés par les consultants et les chefs de projets. Alors qu'il est maintenant reconnu que c'est une mauvaise approche, de nombreux chefs de projets continuent d'employer une approche dirigée par la solution car ils pensent que cela va simplifier la mise en place de l'ERP, réduire le risque, et faciliter le projet de ré-ingénierie des processus organisationnels [Robey2002].

Les travaux menés par ailleurs peuvent être catégorisés en deux familles: la famille orientée management et la famille orientée système. Dans la famille management, les recherches ont tenté de définir l'impact d'installations ERP sur la culture d'entreprise [Krumbholz2000], sur l'organisation [Robey2002], ou sur les processus métier [Esteves2002]. Dans la famille système, le propos a été de guider l'identification et la sélection de l'ERP le plus adapté [Ncube2000], et de guider la découverte d'exigences pour le choix du paramétrage d'ERP le plus adapté [Finkelstein2002].

Notre approche est à la croisée de ces deux familles. L'hypothèse principale est que dans un projet ERP, le problème du changement organisationnel et d'ingénierie du système sont intimement entremêlés. Par conséquent, la démarche méthodologique proposée n'est ni dirigée par le métier, ni par le système, et elle matérialise le lien entre les deux par des modèles conceptuels sur lesquels il est possible de raisonner dans un esprit de co-conception.

Le quatrième résultat est une *expérimentation par le biais d'une étude de cas en grandeur réelle* menée en collaboration avec à la Direction Déléguée des Systèmes d'Information de la Direction de l'Infrastructure de la SNCF qui est en charge du développement et de la maintenance des systèmes d'information supports de l'activité de la Direction de l'Infrastructure de la SNCF [Zoukar2004c], [Zoukar2004d]. Le cas étudié est issu du projet ELAN d'implantation de l'ERP PeoplSoft pour supporter la chaîne logistique de la SNCF. Le périmètre du projet ELAN étant très large, l'étude a été volontairement délimitée en appliquant l'approche à deux des processus

ELAN. Les besoins relatifs à ces deux processus ont été modélisés à l'aide du formalisme MAP, en une hiérarchie de cartes sur trois niveaux. L'ERP choisi est PeopleSoft. Les fonctionnalités de PeopleSoft pour ces deux processus ont également été modélisées par une hiérarchie de cartes. Une autre expérimentation de moindre envergure a été menée avec une association caritative pour la personnalisation de l'ERP NAVISIO [Salinesi2007b].

Durant notre travail dans ce projet, nous avons observé les approches utilisées en pratique et leurs limites, ce qui a renforcé notre conviction de la nécessité d'une approche globale pour traiter le problème d'ingénierie des exigences dans l'implantation d'un ERP.

La complexité des processus concernés par ce projet et son grand périmètre nous ont permis d'évaluer la faisabilité de notre approche dans un grand projet. Cependant, l'étendue du projet sur plusieurs années ne nous a pour l'instant pas permis d'avoir un retour d'expérience et une validation finale de la méthode proposée. L'étude de cas nous a cependant permis de valider un certain nombre d'hypothèses:

- Les cartes sont faciles à lire et comprendre par les acteurs métier de l'organisation et les acteurs techniques du système.
- Le mécanisme d'abstraction/d'affinement proposé par le formalisme MAP est utile lors de la modélisation des besoins. Ceci permet de modéliser les besoins de façon globale, et de ne détailler que ceux pour lesquels cela s'avère nécessaire.
- Lors de la construction des cartes, l'approche Top-down permet de structurer l'analyse des besoins et de produire des cartes à plusieurs niveaux. Ceci permet de communiquer les besoins progressivement. Par exemple, la carte du plus haut niveau sert à montrer à la direction générale une vue globale des besoins du projet. Des cartes opérationnelles sont un moyen efficace pour communiquer des aspects détaillés, importants pour certains utilisateurs du projet, etc.
- Les cartes se révèlent utiles pour assurer une traçabilité du processus de mise en coïncidence.
- L'analyse des similarités est un moyen efficace pour comparer systématiquement des points précis d'un grand nombre de modèles.
- L'approche de mise en coïncidence proposée limite les influences subjectives des consultants.
- Les négociations, lors de la construction des alternatives de coïncidence, donnent à l'organisation une meilleure maîtrise de ses prises de décision.

Une caractéristique distinctive de la méthode MIBE est la généricité. La méthode MIBE est générique pour plusieurs raisons:

- 1) Le niveau d'abstraction (intentionnel) auquel elle se place la rend indépendante du domaine de l'application et du type de technologie utilisée.
- 2) Elle est indépendante des langages spécifiques d'expression des As-Is, Might-Be, As-Wished et To-Be. Elle ne dépend que du langage MAP. Le processus de mise en coïncidence est donc défini une

fois pour toutes. Toutes les directives du processus sont génériques et peuvent s'appliquer dans tous les cas d'installation d'ERP.

- 3) Nous pensons qu'elle peut s'appliquer à d'autres cas de changement, en particulier:
 - a. le cas de l'adaptation d'un système d'information de référence,
 - b. dans le développement d'un système To-Be par assemblage d'éléments tels que des composants logiciels ou des produits COTS (Commercial Off-The-Shelf).

En effet, le processus d'évolution dans chacun de ces deux cas s'apparente au processus de mise en coïncidence.

Enfin, lors de notre étude nous avons observé les dimensions importantes du problème du matching, que ce soit en termes de données à manipuler et de calculs à réaliser. L'emploi du formalisme des cartes nous a permis de faire face à cette difficulté grâce aux mécanismes d'abstraction qu'il propose. Nous pensons cependant qu'une automatisation des calculs est nécessaire pour que la démarche méthodologique puisse être utilisée au plan industriel. Ceci nécessite la réalisation d'un outil qui édite les cartes, calcule leurs similarités et propose des alternatives de coïncidence.

4. Mise en coïncidence par résolution de contraintes – l'approche REDPL

Pour des raisons essentiellement économiques, la communauté du génie logiciel développe des lignes et familles de logiciels. Celles-ci sont construites pour pouvoir faire l'objet d'adaptation par sélection d'un ensemble de variantes, au moment de leur implémentation. La variabilité, concept clef des lignes de produits logicielles, doit être prise en compte au niveau des exigences, tant fonctionnelles que non fonctionnelles. Modéliser la variabilité des exigences et guider le processus de sélection des variantes les plus appropriées à un système donné sont deux questions complémentaires. La sélection de variantes est un processus d'ingénierie des exigences dans lequel les exigences sont des exigences d'adaptation. Le challenge des approches de découverte d'exigences d'adaptation est double: non seulement il faut éviter le déséquilibre qui apparaît systématiquement selon que l'on est dirigé par les exigences organisationnelles ou par les fonctionnalités du système, mais en plus il faut garantir que les variantes choisies au travers des exigences d'adaptation sont cohérentes.

Notre question de recherche est donc la suivante:

"comment garantir que les collections d'exigences d'adaptation qui définissent le système cible sont cohérentes"?

Un point commun essentiel entre les ERP, les services web, ou les lignes de produit, est que les adaptations possibles et leurs incompatibilités peuvent être définies dans les spécifications du système adapté. C'est donc à ces spécifications que les exigences d'adaptation du système cible doivent être conformes.

4.1. Problématiques résolues par la mise en coïncidence par résolution de contraintes

Les méthodes, techniques et outils d'ingénierie des exigences qui s'intéressent au problème de l'adaptation de systèmes ont toutes pour point commun la mise en avant de la variabilité dans la spécification du système adaptable. Du fait que ces systèmes sont destinés à être utilisés dans des contextes variables, ils sont conçus pour satisfaire des exigences variables. Les langages de modélisation qui permettent de spécifier la variabilité des exigences sont nombreux. Leur point commun est de représenter les (a) les décisions qui peuvent être prises en terme d'adaptation, et (b) la spécification de l'assemblage d'exigences par les décisions d'adaptation qui peuvent être prises. Un certain nombre de difficultés fondamentales est cependant soulevé par l'emploi de ce type de langage pour traiter la question de la cohérence avec:

- i. Pré-ordonnement des exigences d'adaptation dans les modèles du système adaptable
- ii. Barrière de la complexité algorithmique de la vérification de cohérence
- iii. Multiplicité des modèles du système adaptable
- iv. Limitation du pouvoir d'expression de ces langages pour spécifier des règles de cohérence

i. Pré-ordonnement des exigences: la création d'un produit par adaptation est une activité longue qui fait l'objet de négociations complexes. Les parties prenantes sont nombreuses: ingénieurs de différents domaines, marketing, utilisateurs, acheteurs, décideurs, etc. De nombreuses dépendances apparaissent entre des arbitrages qui résultent parfois d'équilibres savamment construits suite à de nombreux allers-retours. Certaines exigences d'adaptation peuvent aller à l'encontre d'autres exigences, et les choix doivent souvent être effectués de manière globale. Or, les modèles qui permettent de spécifier la variabilité dans les systèmes adaptables raisonnent en terme de décision et non pas d'exigence d'adaptation. Les décisions d'adaptation sont organisées de manière prédéfinie. Chaque prise de décision joue de manière locale comme le choix d'une exigence parmi n pré-définies. Il n'est donc pas possible d'adopter n'importe quelle exigence à n'importe quel moment: les parties prenantes sont contraintes de respecter l'ordre pré-établi des décisions dans les modèles du système adaptable. Par ailleurs, ces modèles font apparaître une forme d'organisation entre niveaux de décision: les décisions prises à un niveau forcent ou interdisent d'autres décisions définies aux niveaux inférieurs. Il en résulte une forme de déterminisme dans le choix des exigences qui va à l'encontre de la pratique selon laquelle les choix d'exigences d'adaptation sont effectués de manière globale. Le pré-ordonnement des exigences d'adaptation au travers des modèles de variabilité est donc en contradiction avec la pratique.

ii. Complexité algorithmique: la sémantique des modèles des systèmes adaptables peut être définie par la collection des combinaisons admissibles d'exigences d'adaptation. Or, malgré les restrictions que ces modèles permettent de définir, le nombre de combinaisons possible augmente de manière exponentielle avec le nombre d'exigences d'adaptation possibles. Sur de grands systèmes, vérifier qu'une combinaison d'exigences d'adaptation est cohérente avec les spécifications du système pose un problème de complexité algorithmique, en particulier à partir du moment où les décisions d'adaptation sont définies de manière indépendante.

iii. Multiplicité des modèles: les langages de modélisation de systèmes adaptables sont destinés à être employés à créer des modèles utilisés individuellement. Chaque type de langage permet de spécifier les points de vues de certains types de parties prenantes, mais aucun ne permet vraiment de tous les représenter. La prise en compte du point de vue de l'ensemble des parties prenantes lors d'un projet d'adaptation de système nécessite l'emploi de plusieurs modèles définis au moyen de différents langages. Le problème de la multiplicité des modèles du système adaptable est double. Tout d'abord, les exigences d'adaptation qui peuvent être exprimées au moyen de ces différents modèles sont de nature différente. On peut ainsi avoir des exigences non fonctionnelles, des exigences fonctionnelles, des exigences relatives au coût ou à la technologie employée pour le développement, etc. Les modèles à partir desquels ces différentes exigences sont produites étant séparés, les exigences restent elles-mêmes séparées. D'autre part, il est difficile d'appréhender la cohérence des exigences globalement alors qu'aucune forme de dépendance ne relie les modèles à partir desquelles elles sont créées.

iv. Limitation du pouvoir d'expression: les langages au moyen desquels sont créés les modèles de systèmes adaptables permettent de spécifier des règles de cohérence qui se limitent pour l'essentiel à l'exclusion (si l'exigence d'adaptation R1 est choisie, alors l'exigence R2 doit être exclue) et à l'inclusion (si l'exigence d'adaptation R1 est choisie, alors l'exigence R2 doit l'être aussi). Ces deux types de règles sont très limités et ne permettent pas d'exprimer un certain nombre de dépendances, par exemple relativement aux valeurs d'attribut des exigences (si la valeur de l'attribut A d'une exigence R1 est $> C$ alors...), ou aux ensembles d'exigences (si la cardinalité de l'ensemble d'exigences S est $> n$ alors...). Pourtant ces types de contraintes apparaissent souvent dans les faits. Par exemple lorsque des arbitrages doivent être effectués entre des niveaux de performance (tels que vitesse versus continuité de service) la cohérence des exigences d'adaptation doit être définie au moyen de valeurs ("si la continuité de service doit être garantie avec un taux d'interruption $< x_1$, alors il n'est pas possible d'exiger un temps d'exécution du service $< x_2$ s"). Un autre exemple est lorsque la cohérence des exigences est définie par symétrie (par exemple: "le système doit offrir autant de manières de relancer le processus qu'il y en a de l'interrompre") ce type de règle de cohérence entre les exigences d'adaptation ne peut être défini ni par des exclusions ni par des inclusions. Ces limitations du pouvoir d'expression des langages de modélisation de la variabilité peuvent être associées à la nature finie de leur sémantique. En effet, tout modèle défini au moyen de ces langages peut être reformulé en forme normale conjonctive comme l'ensemble des systèmes cibles qu'ils permettent de définir. Il n'est donc pas possible de définir des modèles de systèmes adaptables qui permette de créer une infinité de systèmes-cible. Cette contrainte n'est pourtant validée d'aucune manière, que ce soit empiriquement ou sur le plan théorique.

Nos travaux ont traité ces différents problèmes plus spécifiquement dans le contexte des lignes de produit. Les lignes de produits regroupent des produits qui revêtent des caractéristiques communes tout en présentant des différences. Les lignes de produits sont conçues de manière à supporter la réutilisation de différents éléments tels que composants techniques, documentation, données d'ingénierie, etc. L'approche que nous avons définie tente de résoudre les problèmes définis ci-dessous dans une approche qui permette de répondre à la question de recherche que nous nous sommes posée.

4.2. *Principes de l'approche REDPL*

Nous avons observé que les langages de modélisation de la variabilité existants permettent de conduire l'activité de configuration à la manière d'un programme procédural. L'enchaînement des décisions au travers desquelles sont spécifiées -implicitement- les exigences de choix d'éléments dans une configuration est en fait pré-établi dans le modèle de variabilité initial. Prenant le contre-pied de ce type d'approche, nous avons choisi de considérer le problème de la configuration comme un problème de résolution d'optimum. Lorsque l'on définit une configuration, celle-ci doit satisfaire un certain nombre de critères objectifs, tels que minimiser le coût de production, maximiser les temps de réponse, etc. Considérer la satisfaction de ces objectifs sous l'angle de sélections successives d'éléments ne permet d'établir que des optimums locaux, sans que l'on puisse savoir si globalement la configuration est optimale ou pas. Par conséquent, le choix d'intégrer ou pas un élément à une configuration n'est donc pas une décision indépendante: il doit être possible de la prendre de manière coordonnée pour l'ensemble de la configuration. Dans REDPL, les exigences de configuration peuvent donc être définies progressivement ou en bloque jusqu'à atteindre la configuration qui satisfait les objectifs initiaux.

2.4.a **Spécification de la variabilité au moyen d'un programme de contraintes**

La programmation par contraintes est un outil reconnu pour la résolution de problèmes d'optimisation combinatoire complexes. Les plates-formes de résolution de contraintes disponibles offrent des solveurs qui permettent de résoudre différents types de problèmes de contraintes de manière efficace. Nous avons fait le choix d'utiliser la programmation par contrainte pour résoudre le problème de la configuration à partir de modèles de. REDPL appartient donc la famille des approches dans lesquelles les modèles de variabilité sont spécifiées au moyen de la programmation par contrainte. Dans l'approche REDPL, le modèle de variabilité est un programme de contraintes qui, du point de vue sémantique, correspond implicitement à l'ensemble des configurations possibles du modèle de variabilité. Le programme de contrainte permet de spécifier les contraintes qui régissent les agacements possibles d'éléments non seulement du point de vue de leur compatibilité (inclusion, exclusion classiques), mais aussi du point de vue de leur contribution (positive ou négative) aux objectifs de la configuration.

RED-PL ne se limite pas à traduire des modèles de variabilité spécifiés dans des langages tels que FODA sous la forme de contrainte comme c'est par exemple le cas pour la plupart des approches de ligne de produit qui utilisent la programmation par contrainte. L'idée est au contraire que le modèle de variabilité est le programme de contraintes. La programmation par contrainte étant de nature déclarative, il est possible d'augmenter un programme par contrainte par simple ajout de contraintes. Il est donc possible de rassembler au sein d'un seul programme de contraintes des contraintes qui coïncident à différents modèles de variabilité, y compris s'ils sont spécifiés au moyen de différents langages. Inversement, les langages de modélisation de la variabilité ne permettent en général de représenter que des contraintes binaires de type booléen (qui amènent à inclure ou pas l'élément qui correspond au terme d'une variante dans la configuration).

La programmation par contrainte élargit considérablement le champ des possibilités en permettant de définir des contraintes binaires et n-aires, ainsi que des contraintes sur des valeurs entières, sur des ensembles, etc. Certaines de ces contraintes ne pourraient pas être spécifiées avec les langages de modélisation de variabilité qui existent actuellement.

Enfin, le fait de rassembler en un programme de contraintes unique des spécifications habituellement éparpillées dans des modèles de variabilité séparés ouvre le champs des possibilités du point de vue de la modélisation de variabilité. On peut par exemple procéder à une activité de vérification globale et démontrer des propriétés sémantiques d'un ensemble de modèles (telles que la vivacité d'un élément) qui ne pourraient pas être prouvées sans que les contraintes coïncidant au différents modèles soit considéré simultanément.

2.4.b Mise en coïncidence au moyen d'un solveur de contraintes

Comme le montre la Figure 7, la mise en coïncidence entre les exigences d'adaptation spécifiées par les parties prenantes (As-Wished) et les exigences que le système peut satisfaire (Might-Be) est faite par un solveur de contraintes au moyen de son mécanisme d'unification. Le solveur réalise plusieurs tâches (a) il établit l'unification des exigences d'adaptation spécifiées explicitement par les parties prenantes (As-Wished) avec la liste des exigences que le système peut satisfaire (Might-Be), (b) il déduit les exigences implicites (To-Be) cohérentes avec celles déjà spécifiées (As-Wished), tout en (c) éliminant les combinaisons d'exigences incompatibles avec les contraintes. Enfin, le solveur de contraintes peut répondre aux interrogations concernant l'atteinte de l'objectif poursuivi dans la configuration, par exemple: a-t'on atteint un optimum global, quelle est la liste des décisions qu'il reste à prendre, ou à l'inverse quelles nouvelles exigences ne seraient pas compatibles avec celles déjà spécifiées ?

Tant qu'il existe plusieurs systèmes cibles coïncidant à la collection des exigences déjà spécifiées pour définir la configuration, celles-ci doivent être révisées.

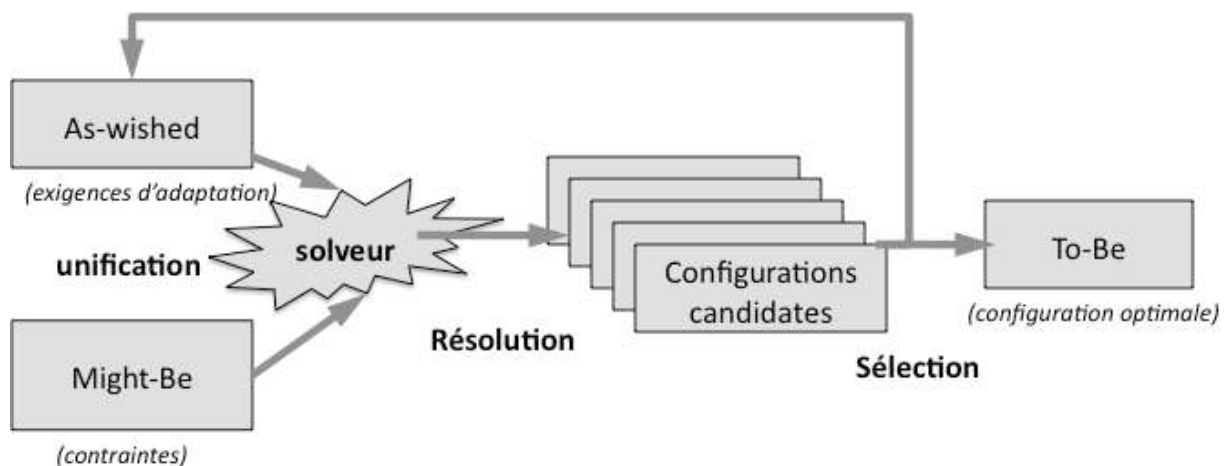


Figure 7: Aperçu de l'approche RED PL

Le programme de contraintes n'impose aucun ordre dans la spécification des exigences. Toutes les décisions peuvent être prises à n'importe quel moment. C'est le solveur de contraintes qui garantit que la collection des

exigences déjà spécifiées pour définir le système cible est conforme aux règles de cohérence définies par le programme de contraintes. La contrepartie de cette approche est que la variabilité n'apparaît plus explicitement, mais implicitement par l'espace des combinaisons permises par les contraintes (l'utilisateur est libre de choisir n'importe quelle exigence n'importe quand).

2.4.c Identification des exigences de configuration par le calcul d'optimum

Comme le montre la boucle à la Figure 7, la résolution de contraintes peut amener à découvrir que plusieurs systèmes cibles sont cohérents avec les exigences déjà spécifiées. La recherche d'optimum impose alors de prendre une décision jusqu'à ce qu'une configuration unique soit définie.

Concrètement, trois possibilités s'offrent alors: la première consiste à choisir parmi les configurations possibles une qui convient suffisamment. Cette stratégie est guidée par le solveur de contrainte lorsque celui-ci est capable de déterminer un ensemble fini de solutions au problème d'optimum, ce qui n'est pas toujours le cas. La deuxième stratégie consiste à spécifier de nouvelles exigences. Dans ce cas, on procède sans savoir si les nouvelles exigences sont cohérentes, ni si elles amènent à la définition d'un système cible unique, ni si elles permettent de se rapprocher de l'optimum souhaité. La réponse à ces questions peut néanmoins être obtenue en interrogeant le solveur de contraintes. La troisième stratégie consiste à modifier les exigences déjà spécifiées. Cette révision des décisions déjà prises peut refléter une remise en cause des arbitrages déjà effectués, par exemple parce qu'ils ne sont pas jugés suffisamment restrictifs, ou parce que l'on juge que les exigences déjà spécifiées empêchaient en fait d'obtenir l'optimum global recherché. Cette boucle peut être stoppée lorsque le solveur indique (i) qu'aucune incohérence n'apparaît entre les exigences du système cible et les contraintes, et (ii) lorsque le système cible qui satisfait l'objectif d'optimisation est bien unique.

4.3. Résultats

Mes travaux sur la mise en coïncidence d'exigences par résolution de contraintes m'ont amené à produire quatre résultats saillants:

- Un état de l'art des méthodes, techniques et outils d'ingénierie des exigences dans le contexte des Lignes de Produit, et une analyse de leur capacité à satisfaire les besoins des entreprises.
- Une approche originale de programmation par contraintes de spécification des exigences associées à une ligne de produits et à un produit configuré.
- Une série de règles de transformation permettant de construire un programme de contraintes à partir de différents modèles de lignes de produits.
- La méthode REDPL de configuration de produits par calcul d'optimum reposant sur la mise en coïncidence des exigences produit avec celles que la ligne de produits est capable de satisfaire.
- Une étude de cas réalisée en collaboration avec la société Biomédicale STAGO pour la retro-conception de sa ligne d'automates d'analyse de plasma sanguin.

Mes travaux de recherche sur ce sujet ont fait l'objet d'une collaboration avec la société STAGO instruments. Cette collaboration m'a apporté une compréhension plus fine des attentes des industriels en matière de méthodes techniques et outils d'ingénierie des exigences dans le contexte des Lignes de Produit. Elle m'a aussi permis de définir la problématique de la mise en coïncidence dans le contexte des Lignes de Produit. Alors qu'on considérait classiquement la question de la configuration comme une succession de choix (souvent pré-ordonnés) au cours desquels les parties prenantes font « leur marché » parmi les éléments de la Ligne de Produit, j'ai défini une approche dans laquelle on s'intéresse en premier lieu aux exigences des parties prenantes. Cette collaboration a enfin permis de démontrer l'efficacité d'une méthode basée sur des techniques de résolution de contraintes pour attaquer ce problème.

Mes travaux sur le thème des Lignes de Produit ont abouti à la publication d'un article à l'intention d'un public d'industriels dans la revue INSIGHT, et de trois articles dans des conférences scientifiques internationales avec comité de lecture: [Djebbi2007a] (International Conference on Advanced information Systems Engineering), [Djebbi2007b] (International IEEE Conference on Requirements Engineering), et [Djebbi2007c] (IEEE Asian Pacific Software Engineering Conference).

J'ai par ailleurs publié trois articles dans des ateliers internationaux: [Djebbi2006a] (International Workshop on Comparative Evaluation in Requirements Engineering), ainsi que [Djebbi2008a] et [Salinesi2009a] (International Workshop on Variability Modelling of Software-intensive Systems).

L'article [Djebbi2007a] présente l'outillage de l'approche proposée au moyen d'un prototype de recherche développé avec Excel. Un nouvel outil plus complet est en cours de développement. Cet outil qui exploitera le solveur de contraintes GNU-PROLOG proposera par ailleurs des fonctionnalités avancées de construction de modèles de Ligne de produit, d'intégration de modèles, de vérification [Salinesi2009a] [Salinesi2010].

Nous proposons une *approche de spécification des exigences de lignes de produits* qui repose intégralement sur les contraintes. Toutes les exigences y sont spécifiées au moyen de contraintes: les exigences de domaines (qui sont capitalisées au niveau des modèles de domaine), aussi bien que les exigences produit (qui définissent les attentes des parties prenantes). Les contraintes sont des expressions arithmétiques simples telles que $A=B$ ou $A<>B$ appliquées à des variables (en l'occurrence A et B). Les variables identifient les éléments de la Ligne de Produit ou les éléments à intégrer dans un produit, et les contraintes spécifient les conditions d'acceptabilité du produit.

Un premier avantage de cette approche est que les collections d'exigences ainsi spécifiées héritent des propriétés intrinsèques aux langages de programmation par contraintes telles que l'unicité, la commutativité (le programme $\{C1, C2\}$ a la même sémantique que le programme $\{C2, C1\}$), et la monotonie (l'espace des solutions se réduit au fur et à mesure de l'ajout de contraintes). Ainsi, dès lors qu'elles sont exprimées sous formes de programmes de contraintes, il est possible de rapprocher les exigences exprimées librement par les parties prenantes (As-Wished) et celles que le système est capable de satisfaire (Might-Be) en les rassemblant sous la forme d'un programme rassemblant les deux ensembles de contraintes. L'utilisation conjointe (en un programme) de ces deux collections de contraintes pour trouver une solution tire avantage de la propriété de monotonie: plus on ajoute de contraintes, plus l'espace des solutions est réduit. La mise en coïncidence est assurée par la production

de solutions qui satisfont simultanément l'ensemble des contraintes: celles de la Ligne de Produits et celles qui correspondent aux exigences des parties prenantes à l'égard du produit.

Notre approche s'inscrit dans un courant récent de travaux qui ont étudié l'apport de l'approche déclarative dans la spécification des systèmes adaptables, plus particulièrement des lignes de produits [Benavides2005], [Mannion2002], [Zhang2004], [Batory2005]. Cependant, le principe généralement mis en avant dans ces approches est de traduire un modèle de variabilité sous la forme de prédicats de la logique du premier ordre. Cette approche a démontré son utilité (un solveur SAT permet de guider le processus de configuration de manière fiable et efficace). Cependant, la mise en coïncidence y est réduite à la portion congrue. En effet, les variables que permettent de traiter ce type de solveur sont des variables booléennes. Dans les programmes de contraintes, elles ne permettent de spécifier qu'un seul type de variabilité: un élément est inclus ou pas dans un produit. Le pouvoir d'expression des parties prenantes s'en trouve considérablement limité: ceux ci n'ont pas d'autre choix que de choisir ou refuser les possibilités offertes par la Ligne de Produit. Ces approches sont donc intrinsèquement « dirigées par la solution ».

L'originalité de notre approche est double: d'une part, nous procédons en programmation par contraintes sur des variables à domaine fini pour construire les modèles d'exigence *As-Wished* et *Might-Be*, d'autre part, nous procédons par une approche de configuration équilibrée (au sens où elle met sur un pied d'égalité les exigences et les solutions possibles) pour construire le système cible (*To-Be*).

La spécification de contraintes sur des variables à domaines finis apporte une certaine richesse à la spécification de la variabilité des Lignes de Produits. Ainsi, il est possible de spécifier de manière efficace des contraintes sur des variables intrinsèquement numériques (telles que «Mémoire > 1024 * Blocs») ou sur des groupes de variables (telles que «NbreClésAdministrateurs+NbreClésUtilisateurs < 10»). La construction d'un modèle de ligne de produit directement comme un programme de programmation par contraintes, n'est pas tout à fait identique à l'approche qui consiste à d'abord construire un modèle de variabilité puis le traduire en un modèle de contraintes. En effet, la programmation par contrainte offre des possibilités qui vont au-delà de ce que les modèles de variabilité permettent de représenter individuellement: contraintes entières, contraintes sur des réels, contraintes sur des ensembles, contraintes mettant en jeu plus de deux éléments (alors que les contraintes spécifiées dans les modèles de variabilité sont pour l'essentiel binaires), etc. Bien entendu, les modèles de variabilité peuvent être une source d'information pour construire les modèles de contraintes. Nous pensons que les limitations inhérentes à ces modèles amènent cependant à restreindre le champ de spécification des contraintes, et que de nombreuses contraintes pourraient être exprimées de manière plus efficace, et parfois plus naturelle, en adoptant directement une approche de programmation par contrainte. D'un point de vue pratique, il est en revanche certain que de nombreux praticiens sont plus familiers avec les langages simples de spécification de Lignes de Produit. La transformation automatique de modèles spécifiés à partir de ces langages en programme de contraintes est donc une préoccupation importante.

La seconde particularité de notre approche réside dans la manière de percevoir la construction du système cible. Les approches traditionnelles procèdent par "dérivation". Lorsque les exigences d'un système cible (*To-Be*) sont

dérivées à partir du modèle des exigences d'un système adaptable (Might-Be), la garantie de conformité est apportée par construction. C'est à dire que c'est le fait de parcourir le modèle de variabilité en choisissant les exigences du système cible parmi celles qui restent disponibles qui permet d'assurer que le système cible est cohérent avec le modèles de Ligne de Produit. Dans cette approche, la cohérence du Produit avec le modèle de Ligne de Produit n'est pas évaluée à partir de modèles déjà construits, ce qui pose problème lorsque le modèle de produit change ou que les parties prenantes ont une idée préconçue des exigences du Produit. Notre approche consiste, au contraire, à évaluer a posteriori la cohérence des exigences du système cible par le biais du mécanisme de résolution de contraintes. Lorsque au moins une solution du programme de contraintes respectant toutes les contraintes peut être trouvée, le Produit peut être considéré comme cohérent avec la Ligne de Produit. Les exigences peuvent être considérées comme correctes si, en plus d'être cohérentes elles sont complètes, c'est à dire qu'il existe une solution unique à l'ensemble des contraintes.

L'étude de cas montre que la méthode REDPL est scalable et peu intrusive. En effet, nous pensons qu'elle ne remet pas en cause l'emploi des modèles de variabilité, car il est possible de la mettre en œuvre sans que les parties prenantes aient à lire un programme de contraintes. On peut par exemple laisser les parties prenantes parcourir les modèles de variabilité pour prendre les décisions d'inclure ou pas les différentes exigences d'adaptation, et d'utiliser le programme de contrainte pour vérifier que les exigences produites sont cohérentes. Les modèles de variabilité ne doivent donc pas être supprimés car ils sont utiles à coordonner les décisions lorsque cela est nécessaire. Cette approche permet de bénéficier du guidage apporté par les modèles à variabilité, tout en bénéficiant du fort degré de liberté introduit par la programmation par contrainte, puisque grâce à elle n'importe quelle exigence d'adaptation peut tout de même être spécifiée n'importe quand.

L'utilisation de la programmation par contrainte ouvre des champs entiers d'investigation difficile à explorer par les techniques de variabilité. Ainsi, dans le domaine des télécommunications, il faut tenir compte de relations d'ordre temporel entre les exigences. Ces contraintes peuvent être exprimées en programmation par contraintes, alors que leur complexité est au-delà du pouvoir d'expression des relations d'inclusion et d'exclusion entre paires d'exigences que proposent les langages de variabilité. Des techniques classiques en programmation par contraintes telles que la réification (qui consiste à associer une variable à une contrainte « $CR1 \Leftrightarrow A > B$ ») sont la base pour exprimer des préférences entre contraintes (« $CR1 > CR2$ »). Un autre champ d'investigation concerne donc la dimension économique des Lignes de Produit: non seulement la question du choix des exigences d'un produit, mais aussi le problème de la construction de la Ligne de Produit peuvent être traités comme une série d'arbitrages visant à un choix optimal tenant compte de préférences et de contraintes (telles que le coût) [Djebbi2007].

La courbe d'apprentissage est plus importante en ce qui concerne la conception des systèmes adaptables. Les premières expérimentation avec un prototype de recherche nous a fait apparaître que les stratégies de guidage peuvent être démultipliées pour surmonter cette difficulté: import à partir d'autres modèles, primitives, analyse de données etc. Les programmes produits sont denses, parfois plus naturels et plus lisibles que des modèles de variabilité. Trois avantages restent déterminants à nos yeux: (i) la possibilité d'utiliser la programmation par contraintes pour exprimer des contraintes qui portent sur des éléments de différents modèles, y compris définis

dans des méta-modèles différents, et (ii) le pouvoir d'expression, et (iii) la disponibilité de plate-forme efficaces pour la résolution de contraintes.

5. Perspectives de recherche

Nos travaux actuels sont organisés selon deux directions: (a) l'amélioration de l'analyse de similarités, en particulier au moyen de techniques de réutilisation, et (b) l'aide à la construction des modèles de systèmes adaptables (Might-Be).

L'évaluation de la méthode MIBE à la SNCF a montré que l'analyse de similarité est une tâche complexe, en particulier en raison des différences dans les choix de modélisation des exigences initiales (As-Wished), et du système adaptable (Might-Be). Les différences dans la manière de définir les périmètres des modèles, dans la perception des niveaux d'abstraction, dans les choix d'affinement, dans les choix terminologiques et dans la manière de formuler les exigences sont autant d'obstacles à la mise en oeuvre des mesures automatiques de similarité. Nous pensons qu'il est possible d'agir à deux endroits: d'une part dans la construction des modèles des exigences initiales (As-Wished), et d'autre part dans l'analyse de similarité elle-même. Notre approche consiste à utiliser des modèles de domaine génériques. Les modèles de domaines qui nous intéressent sont génériques au sens où ils sont indépendants d'un système particulier (contrairement aux modèles de domaines définis pour les lignes de produits). Ils représentent des exigences typiques qu'ont eues (ou pourraient avoir) des parties prenantes lors de différents projets d'intégration de système adaptables. Ces modèles peuvent être non seulement utilisés pour construire les modèles des exigences initiales (As-Wished), mais aussi pour construire des modèles des exigences des systèmes adaptables (Might-Be). Notre hypothèse est que si tous ces modèles spécifiques (respectivement au projet et au système) sont construits à partir des mêmes modèles génériques, alors il est plus facile d'en analyser la similarité. La facilité vient non seulement du fait que les modèles seront plus cohérents en terme de structure, mais aussi du fait qu'il ne sera pas obligatoirement nécessaire de refaire des calculs de similarité dans la mesure où ils auront déjà été faits.

Notre utilisation de la programmation par contraintes s'est pour l'instant limitée à la programmation par contrainte sur les domaines finis (booléens, entiers, énumérations). Or tous les types de contraintes ne peuvent être spécifiés avec ce domaine de contrainte. Nous avons notamment bon espoir d'ouvrir de nouvelles possibilités en étendant notre approche au domaine de contrainte des ensembles, notamment pour définir des contraintes d'inclusion, d'intersection, ou d'exclusion qu'aucun modèle de variabilité ne permet actuellement de définir, mais aussi pour définir des contraintes applicables à tous les éléments d'ensembles, sans avoir à définir à l'avance quels sont ces éléments. Une difficulté que l'on peut d'ores et déjà envisager est l'implémentabilité: différents solveurs sont disponibles pour analyser les programmes construits dans différents domaines. Aucun solveur « général » n'étant actuellement disponible, analyser des contraintes relevant de différents domaines nécessiterait de faire coopérer ces différents solveurs. Cette difficulté est cependant plus technique que conceptuelle, et nous avons bon espoir qu'elle puisse être levée à plus ou moins brève échéance.

Enfin, nous entamons de nouvelles recherches sur le problème de la construction de modèles d'exigences des systèmes. Une difficulté importante réside dans la vérification de ces modèles. Tout comme les modèles d'exigences de systèmes classiques, les modèles d'exigences de systèmes adaptables doivent respecter un certain nombre de propriétés telles que non-ambiguïté, cohérence, complétude, faisabilité ou testabilité. Nous pensons que le fait de considérer des modèles de système adaptables démultiplie la complexité de la vérification des modèles d'exigences, voir les rend inapplicables. Par exemple, la faisabilité se pose non pas pour un système, mais pour l'ensemble de tous les systèmes cibles possibles qui peuvent être produits par adaptation. Or le nombre de systèmes cibles peut être extrêmement important, voir impossible à prédire. Il est donc nécessaire de raisonner de manière différente. L'approche de la programmation par contrainte offre la possibilité d'interroger les modèles. Nous pensons utiliser cette possibilité pour développer des techniques d'exploration des modèles d'exigences des systèmes adaptables afin d'en vérifier certaines propriétés.

CHAPITRE 4

Ingénierie des exigences dans le contexte de l'évolution

1. Introduction et positionnement de mes travaux sur le contexte de l'évolution

La durée de vie des systèmes d'information ne cesse d'augmenter. Nonobstant, les efforts pour les faire évoluer dépassent largement ceux consentis pour les développer. Le problème n'est pas récent: déjà dans les années 70, l'US Air Force avait estimé que 30 \$ étaient dépensés par instruction lors du développement d'un de ses systèmes puis 4000 \$ pour sa maintenance [Boehm1981]. On imagine facilement les enjeux quand on considère les dimensions de certains systèmes: ainsi, le DoD indique maintenir des centaines de systèmes d'information hétérogènes répartis géographiquement sur plus de 1700 sites pour un total de 1,4 milliard de lignes de code [Aiken1994].

De nombreux exemples illustrent l'influence que peuvent avoir les évolutions technologiques et leur intégration rapide aux systèmes d'information sur le fonctionnement des organisations. La numérisation des échanges a par exemple eu un impact très fort chez Radio France et M6 [CIGREF2007]. L'apparition de nouveaux types de systèmes tels que les Data Warehouse, ou les progiciels de CRM, a profondément influencé les transformations récentes des activités de contrôle de gestion et de marketing-vente. Selon une étude menée récemment par CapGémini [CapGémini2008]⁶ 83% des directeurs des systèmes d'information estiment que la capacité des systèmes d'information à évoluer facilement est un pré-requis à l'évolution des métiers.

Inversement, les organisations faisant face à un rythme de changement jamais égalé, leurs systèmes d'information doivent s'adapter très rapidement. De nombreux exemples récents montrent l'enjeu de l'évolution des systèmes d'information lors des changements au niveau des organisations. Par exemple, le changement de statut public / privé et la dérégulation des marchés a eu un impact fort sur des sociétés telles que EDF, La Poste, ou France Telecom, et en particulier sur leurs systèmes d'information. Or l'étude de Cap Gémini citée plus haut montre que 38% des directeurs de systèmes d'information interrogés estiment que "la fonction informatique n'apporte pas aux métiers l'agilité dont ils ont besoin", et 29% pensent qu'elle est "incapable de suivre le rythme des évolutions de leur environnement".

Les méthodes d'ingénierie des systèmes d'information doivent intégrer cette exigence de maîtrise de l'évolution et proposer des démarches autres que celles qui consistent à re-dérouler toutes les étapes du cycle de développement initial. Des concepts fondamentaux adaptés doivent être définis et des démarches proposées. Ces démarches doivent pouvoir être rapidement mises en œuvre et permettre de respecter les délais organisationnels

⁶ Selon la même étude, 45% des budgets sont dépensés à la maintenance, alors que 34% vont au développement.

de changement. Plus important encore, elles doivent tenir compte des spécificités de la problématique d'évolution.

2. Positionnement

De nombreuses approches posent la question de l'évolution à un niveau technique. On trouve par exemple des techniques de gestion de configuration, de refactoring, de migration de code. De nouvelles technologies apparaissent pour développer des systèmes plus aptes à évoluer; on peut par exemple penser aux technologies orientées objet, ou encore plus récemment à la programmation par aspects et aux architectures orientées service. Notre position est de considérer la problématique de l'évolution à un plus haut niveau d'abstraction, sous l'angle de l'ingénierie des exigences. Nos travaux ont porté (a) sur la spécification d'exigences d'évolution, et (b) sur la place des exigences dans le contexte de la coévolution.

Le principal précurseur dans l'étude des phénomènes d'évolution des systèmes à base de logiciels est sans doute Lehman [Lehman2000]. Lehman a principalement cherché à théoriser l'évolution des logiciels afin d'influencer les démarches méthodologiques de développement de logiciels. D'autres travaux tels que [Harker1993], [Anton2001], [Anderson2002] et [Nurmuliani2004] ont exploré les phénomènes d'évolution plus spécifiquement sous l'angle de l'ingénierie des exigences. Des démarches d'ingénierie des exigences ont par ailleurs été développées pour traiter le problème de l'identification des exigences d'évolution. Les approches sont la prédiction [Avison1994], l'analyse de la volatilité [Moreira2006], l'emploi de taxonomies [Anderson2002], ou d'ontologies [Ruiz2004]. Nos travaux s'inscrivent dans le courant [Lamsweerde2004], [Zowghi1996], [Krishna2004] qui s'intéresse en premier lieu à la définition de langages pour la spécification d'exigences d'évolution.

Concernant la spécification, nous avons développé un langage qui permet de spécifier des exigences d'évolution sous la forme d'écarts. Afin de permettre le déploiement de notre approche, nous avons élaboré un langage générique et adaptable, et proposé une démarche systématique pour son adaptation aux méthodes de développement traditionnelles.

Concernant la co-ingénierie, nous avons développé un cadre qui définit systématiquement le rôle que peuvent jouer les exigences dans un projet d'évolution de différents artefacts liés (on parle de coévolution). Ce cadre nous a, par la suite, permis de développer une démarche d'ingénierie des exigences pour l'évolution des systèmes d'information qui est présentée au chapitre suivant.

3. Spécification des exigences d'évolution par écarts – l'approche GAP

Le courant de pensée dans lequel nous nous inscrivons considère qu'avant même de savoir procéder à la modification de référentiels ou de documents de spécification des exigences (comme le propose par exemple [Tanabe2008]), il faut maîtriser la spécification explicite des exigences d'évolution. Notre préoccupation première est donc de disposer d'un langage adapté à la spécification d'exigences d'évolution.

On peut envisager trois approches pour développer un langage de spécification d'exigences d'évolution: de manière ad hoc, de manière globale, ou par transposition. L'approche ad hoc consiste à chercher le langage qui serait le plus adapté au contexte méthodologique dans lequel les exigences devront être spécifiées. Dans l'approche globale, l'idée est au contraire de définir un langage de spécification des exigences qui puisse être utilisé dans le contexte de n'importe quelle méthodologie. La transposition consiste à s'inspirer d'un langage destiné à être utilisé dans un contexte méthodologique particulier pour définir un langage destiné à un autre contexte.

Nous pensons que l'approche ad-hoc n'est pas adaptée car sa mise en œuvre reposerait sur la connaissance et le savoir-faire de quelques personnes, et elle pourrait être influencée par le contexte de projet. Par ailleurs, l'emploi des approches ad hoc et par transposition n'étant pas forcément répétable, ces deux approches peuvent s'avérer peu systématiques. L'approche globale et l'approche par transposition présentent le défaut de partir d'un langage dont la définition ne tient pas compte des spécificités du contexte méthodologique dans lequel l'emploi du langage va s'inscrire. La préoccupation de l'adaptation du langage au contexte méthodologique étant selon nous centrale, la question de recherche que nous avons posée est:

« quel langage permettrait de spécifier des exigences d'évolution dans le contexte de différentes démarches méthodologiques? »

Conformément au cadre de travail de Jarke et Pohl, nous envisageons les exigences d'évolution sous l'angle de la transition d'une situation As-Is à une situation To-Be toutes deux décrites au moyen de modèles conceptuels. Un langage permettant de spécifier des exigences d'évolution doit permettre d'exprimer les écarts requis entre ces deux situations. Par convention, nous parlerons d'*opérateurs d'écart*. Les opérateurs d'écart forment les primitives du langage de spécification des exigences d'évolution. La question au centre du développement d'un langage de spécification d'exigences de changement est donc celle de la construction d'une typologie d'opérateurs d'écart.

Différents travaux proposent diverses typologies d'opérateurs d'écart dans différents domaines. Par exemple, [Banerjee1987], [Zicari1992], [Aj-Jadir1995], [Estublier2000], [Delgado2003] s'intéressent à l'évolution des modèles de bases de données, [Al-Jadir2003] définit des opérateurs d'évolution pour des DTD XML. [Kradolfer2000], [Reichert1998], s'intéressent à l'évolution de modèles de Workflow. [Soffer2004] définit un ensemble d'opérateurs de changement pour faire évoluer un modèle de processus. [Mens2001] gère l'évolution des logiciels en utilisant des opérateurs de transformation. Dans tous les cas, les opérateurs d'écart permettent bien d'exprimer les différences entre deux versions de modèles.

L'étude de ces travaux fait apparaître une grande variété dans les typologies d'opérateurs qui rend leur comparaison difficile. Cependant, malgré de nombreuses différences, ces typologies partagent les caractéristiques suivantes:

- elles peuvent être constituées différemment selon que les opérateurs modifient la structure, i.e. les relations entre éléments, ou les éléments eux-mêmes ;

- elles proposent généralement un opérateur d'ajout et de suppression (que ce soit de liens, d'éléments ou de propriétés), et
- elles satisfont un certain nombre de critères de qualité, en particulier la complétude et la correction.

On remarque cependant que certains opérateurs d'écart n'apparaissent que dans quelques typologies, tels que l'opérateur « changer » qui se distingue de la succession Ajouter - Supprimer car il est sans perte d'information [Al-Jadir2003]. Ainsi, certaines typologies privilégient la recherche d'opérateurs qui soient "primitifs" au détriment de la puissance d'expression. La question des caractéristiques du langage, et donc du choix des opérateurs d'écart, se pose donc. La section suivante définit cette problématique sous l'angle d'une série de qualités que le langage de spécification des exigences d'évolution -et donc de la typologie d'opérateurs- doit selon nous respecter.

3.1. *Problématique de la spécification d'exigences d'évolution*

Comme tous les auteurs cités ci-dessus, nous cherchons à définir une typologie d'opérateurs d'écart qui soit *complète* et *correcte*. Ainsi que le montrent [Teeuw1997] et [Casati1996], d'autres critères de qualité de typologies d'opérateurs d'écart peuvent être considérés, en particulier la *consistance*, la *minimalité* et la *richesse sémantique*. Les quatre problèmes choisis sont relatifs à ces cinq critères de qualité. Il s'agit de l'incomplétude, de l'incorrection, du conflit d'opérateurs, et de la pauvreté sémantique.

i. Incomplétude: Un langage de spécification d'exigences d'évolution est considéré comme *complet* si la typologie d'opérateurs d'écart qui le définit subsume tout type d'exigences d'évolution possible. En d'autres termes, une typologie d'opérateurs d'écart est complète si elle permet de décrire l'évolution de n'importe quel modèle en n'importe quel autre modèle [Kradolfer2000]. Un langage de spécification d'exigences d'évolution incomplet ne permettrait pas de spécifier toutes les exigences d'évolution.

ii. Incorrection: Selon Banerjee [Banerjee1987], une typologie d'opérateurs d'écart est *correcte* si chacun des opérateurs qu'elle propose est correct, c'est à dire s'il n'aboutit pas à un modèle incorrect. La correction d'un modèle peut par exemple être définie par un ensemble d'invariants qui sont des conditions sur le modèle. Les invariants doivent être satisfaits avant et après toute modification du modèle. Inversement, un langage de spécification d'exigences d'évolution qui reposerait sur une typologie d'opérateurs d'écart incorrectes permettrait de spécifier des exigences d'évolution impliquant une situation To-Be qui ne serait pas définie correctement.

iii. Conflits d'opérateurs: Il est important que le langage de spécification ne soulève aucune ambiguïté en ce qui concerne le choix d'un opérateur d'écart pour spécifier une exigence d'évolution donnée. Selon [Teeuw1997], une typologie d'opérateurs d'écart est *consistante* si les définitions des différents opérateurs n'entrent pas en conflit les unes avec les autres, et les définitions des opérateurs d'écart sont en conflit si une même exigence d'évolution peut être spécifiée par plusieurs opérateurs différents. Un langage de spécification d'exigences d'évolution qui présente des conflits d'opérateurs impose des choix de forme (la même exigence pouvant être spécifiée par plusieurs opérateurs sémantiquement équivalents), et peut s'avérer plus difficile à interpréter en raison des variantes de forme introduites artificiellement.

iv. Pauvreté sémantique: Avant de préciser la nature du problème de pauvreté sémantique, il est important de définir le critère de qualité de minimalité. La *minimalité* d'un langage de spécification d'exigences d'évolution fait référence à l'obtention de la propriété de complétude par un ensemble minimal d'opérateurs. En d'autres termes, un ensemble d'opérateurs d'écart est minimal s'il est complet et qu'il ne contient aucun opérateur qui puisse être obtenu par composition d'autres opérateurs [Casati1996]. La minimalité est, d'une certaine manière, corrélée à la richesse sémantique. Un langage de spécification d'exigences d'évolution peut être considéré comme *sémantiquement riche* si toute exigence d'évolution peut être exprimée en n'utilisant qu'un seul de ses opérateurs d'écart. Inversement, la pauvreté sémantique d'un langage amène à combiner plusieurs opérateurs d'écart pour spécifier une seule exigence d'évolution, ce qui génère une déperdition de l'information [Al-Jadir2003]. De plus, une typologie d'opérateurs d'écart sémantiquement riche permet de spécifier chaque exigence d'évolution de manière naturelle sans combiner artificiellement les opérateurs d'écart.

Il est clair que les propriétés de minimalité et de richesse sémantique ne peuvent pas être satisfaites par la même typologie d'opérateurs d'écart dans la mesure où elles se contredisent. Nous proposons donc de préciser dans le langage à définir la partie minimale de notre typologie d'opérateurs, tout en proposant également les opérateurs qui permettent à notre langage de spécification des exigences d'évolution d'être sémantiquement riche.

3.2. Principes de l'approche GAP

L'approche de développement d'un langage de spécification d'exigences d'évolutions par écarts que nous proposons a pour but (i) d'identifier les opérateurs d'écarts formant la base du langage de spécification satisfaisant un certain nombre de propriétés et (ii) de fournir les moyens d'adapter les opérateurs d'écarts en fonction du contexte.

Pour surmonter les difficultés qui découlent des approches « ad hoc » et « par transposition » de développement de langages de spécification d'exigences d'évolution, GAP propose une typologie générique d'opérateurs d'écart. Cette typologie est donc définie indépendamment du formalisme utilisé pour représenter les modèles As-Is et To-Be. Différentes typologies d'écarts, par exemple associées à un méta-modèle orienté-objet ou à un méta-modèle de but, peuvent être générées à partir de la typologie générique.

2.3.a Spécification des exigences d'évolution au moyen d'un langage adapté d'une typologie générique d'écarts

Comme le montre la Figure 8, l'approche est à trois niveaux: le niveau modèle, le niveau méta-modèle et le niveau méta-modèle générique. Ces trois niveaux coïncident respectivement aux niveaux M1, M2 et M3 de [Bézivin2001].

Au *niveau modèle*, sont définis les modèles (aussi désignés produits ou schémas) avant et après évolution. C'est à ce niveau qu'est défini le modèle pivot représentant les processus d'entreprise et les fonctionnalités du système. C'est aussi à ce niveau que sont définies les exigences d'évolution (représentées par la lettre grecque Δ), sous la forme d'écarts exprimant les évolutions à mettre en œuvre entre les deux modèles. Une hypothèse importante est

que les deux modèles As-Is et To-Be sont définis avec le même langage. Nous ne nous intéressons donc pas, comme dans [Terrasse2003] et [Bezivin2001], aux évolutions dans les cas où les modèles As-Is et To-Be sont des instances de deux méta-modèles différents, cas qui soulève de nombreuses problématiques difficiles liées à la transformation de modèles, telles que la discordance conceptuelle.

Au *niveau méta-modèle* apparaissent un méta-modèle spécifique et une typologie d'écarts associée. Le méta-modèle spécifique est celui de la méthodologie réellement employée dans un projet. Il définit le type des éléments utilisés dans les modèles As-Is et To-Be; ces derniers étant des instances du méta-modèle spécifique. De la même manière, la typologie spécifique d'écarts spécifie le langage employé pour définir les exigences d'évolution au niveau modèle. Les écarts identifiés entre le modèle As-Is et le modèle To-Be au niveau modèle sont des instances de la typologie spécifique d'écarts définie au niveau méta-modèle.

Le *niveau méta-modèle générique* propose une typologie générique d'écarts et un méta-modèle générique à partir desquels sont respectivement définis la typologie spécifique et le méta-modèle spécifique. Le méta-modèle générique identifie les concepts génériques nécessaires à la définition d'opérateurs génériques rassemblés au sein de la typologie générique d'opérateurs d'écarts. Le méta-modèle générique permet, par instanciation, de rendre explicites les éléments et la structure des méta-modèles spécifiques. Il s'agit donc d'un langage générique et adaptable de spécification des exigences d'évolution.

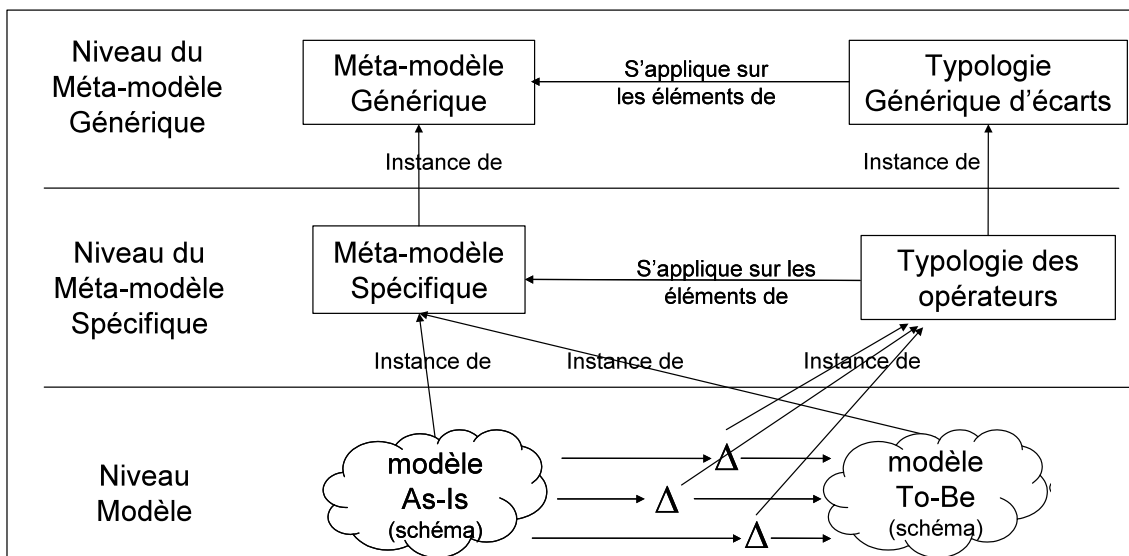


Figure 8: Aperçu de l'approche GAP

2.3.b Définition d'une typologie générique d'écarts au moyen d'un méta modèle générique

Le méta-modèle générique cherche à identifier systématiquement les aspects sémantiques et structurels susceptibles de faire l'objet d'un écart sur les différents types d'éléments qui composent le méta-modèle spécifique.

L'utilisation d'une typologie générique relative à un méta-modèle générique permet d'être indépendant de tout facteur spécifique. Le méta-modèle générique peut être instancié par chacun des méta-modèles utilisés. La typologie générique associée au méta-modèle générique est adaptée pour coïncider à chacun des méta-modèles spécifiques. Une telle approche permet d'identifier systématiquement les aspects sémantiques et structurels susceptibles de faire l'objet d'un écart sur les différents types d'éléments qui composent le méta-modèle spécifique. Par exemple, si on utilise le méta-modèle Entité-Relation pour représenter le modèle d'une base de données, alors les écarts seront exprimés au niveau modèle entre deux modèles Entité-Relation. Les écarts entre les modèles As-Is et To-Be, expriment ce qui doit évoluer entre les deux situations. Ils instancient les opérateurs de la typologie spécifique. On peut par exemple exprimer que l'entité type Réservation doit être éclatée en deux entités types Réservation et Demande et que la relation type 'correspond' qui a pour source Réservation et pour cible Demande doit être ajoutée. Ces types d'écart sont définis avec l'opérateur associé par adaptation de la typologie générique d'écarts en conformité avec le méta-modèle spécifique Entité-Relation.

Le méta-modèle générique considère que tout méta-modèle est constitué d'*Eléments*, qui ont un *Nom* et un *Type*, et sont caractérisés par un ensemble de Propriétés. Il existe deux façons orthogonales de classer les Eléments. La première classification fait la distinction entre éléments *Simple*s et *Composés*. La seconde classification différencie les éléments *Liens* et *Non liens*. Les *Eléments Composés* sont décomposables en éléments plus fins (qui peuvent être simples ou à leur tour composés). Un *Elément Lien* est un connecteur entre deux éléments, l'un jouant le rôle de Source et l'autre celui de Cible.

La typologie générique d'écarts identifie trois types majeurs de changements: les changements de nommage, les changements intrinsèques et les changements structurels. Les changements de nommage affectent seulement la façon dont les organisations veulent se référer à un élément. Ce type ne compte qu'un seul opérateur générique *Renommer*. Les changements intrinsèques affectent les éléments en se limitant aux éléments eux-mêmes sans tenir compte de leur relation avec d'autres éléments: modifier le domaine d'un Attribut est un exemple d'un tel changement localisé. L'approche GAP propose quatre opérateurs pour spécifier les changements au sein même d'un élément, à savoir, *Modifier*, *Joindre*, *Enlever*, qui concernent les propriétés des éléments et *Retyper* qui change le type d'un élément et permet, par exemple, de transformer une Relation Type du As-Is en Entité Type du To-Be. Les changements structurels font intervenir plusieurs éléments et coïncident, non plus à des transformations concernant un élément unique, comme c'est le cas des changements intrinsèques, mais à des modifications de plusieurs, voire de tous les éléments qui composent le méta-modèle. Il y a neuf opérateurs pour spécifier ce type de changement: *Ajouter*, *Supprimer*, *Remplacer*, *Fusionner*, *Diviser*, *ChangerOrigine*, *AjouterComposant*, *SupprimerComposant* et *DéplacerComposant*. Par exemple, ajouter ou supprimer des relations types ou des entités types dans un modèle E/R As-Is pour construire le modèle To-Be sont des changements structurels. AjouterComposant consiste à ajouter un élément dans un élément composé autre que le modèle lui-même. Ainsi, pour ajouter un attribut dans une entité type, il faudra utiliser l'opérateur AjouterComposant dans la mesure où, comme nous l'avons vu précédemment, une Entité Type est un Elément composé d'Attributs.

2.3.c Définition formelle des opérateurs générique au moyen d'invariants

La définition des opérateurs repose sur deux concepts: une signature et un prédicat. La signature identifie le type des éléments impliqués dans les modèles As-Is (c'est-à-dire avant l'exécution de l'opérateur) et To-Be (c'est-à-dire après l'exécution de l'opérateur). Le prédicat est composé de deux éléments: une expression de la logique du premier ordre et éventuellement des paramètres. L'expression n'indique pas comment modifier le modèle As-Is mais spécifie les conditions que doit satisfaire le modèle To-Be. Elle repose sur les concepts du méta-modèle spécifique ; un concept étant un Élément ou une Propriété. Le paramètre fait référence à un concept du méta-modèle ; il identifie le type d'un élément de modèle auquel se rattache l'exigence d'évolution spécifiée.

2.3.d Démarche de définition d'un langage de spécification des exigences de changement par adaptation de la typologie générique d'écart au méta modèle spécifique

Outre un méta-modèle générique et une typologie d'opérateurs, GAP propose un processus en plusieurs étapes qui guide la démarche de développement du langage de spécification d'exigences d'évolution. Ce processus cherche à garantir les critères de qualité évoqués plus haut. La qualité du langage de spécification des exigences résulte donc non seulement de la qualité de la typologie générique d'opérateurs d'écart, mais aussi de la manière de dérouler le processus de dérivation de la typologie spécifique d'opérateurs d'écart à partir de la typologie générique.

3.3. Résultats

Les travaux menés autour de la définition d'un langage de spécification des exigences de changement sont les suivants :

- Proposition et expérimentation de l'emploi d'une approche de spécification des exigences d'évolution par les écarts [Salinesi2003b] [Etien2003c] [Etien2004a] [Salinesi2004d] [Etien2004b]; cette approche a par la suite été formalisée et nommée GAP ;
- Définition d'un langage générique d'écarts [Etien2004c] [Etien2005];
- Evaluation du langage d'écart au moyen de critères de qualité [Etien2003a] [Etien2003b]; et
- Proposition d'une démarche méthodologique pour définir un langage d'écart « spécifique », c'est à dire adapté au langage employé pour spécifier les modèles sources et cibles entre lesquels les écarts sont spécifiés [Rolland2003].

Comme le montrent les références ci dessus, les travaux menés sur le thème de la spécification d'exigences d'évolution ont abouti à la publication de 9 articles scientifiques dans des revues, conférences et ateliers internationaux avec comité de lecture. On distinguera en particulier la publication d'un article dans la revue Springer Requirements Engineering Journal [Rolland2004], ainsi que la publication d'articles dans les conférences CAiSE (International Conference on Advanced information Systems Engineering) [Etien2004c] OOIS (Object

Oriented Information Systems) [Etien2003c] et la European Japanese Conference on Information Systems and Knowledge Bases [Etien2004a].

La diffusion de ces travaux paraît assurée, puisque l'article [Rolland2004] est référencé plus de 36 fois selon Google Scholar. Enfin, ces travaux sont à la base de deux travaux significatifs : une collaboration avec la société Renault DIAC sur le thème de l'évolution de son Système d'Information, et la thèse de doctorat d'Anne Etien, soutenue en 2006.

La démarche méthodologique GAP appartient à un courant de l'ingénierie des exigences qui considère qu'il faut savoir *spécifier les exigences d'évolution*, et pas seulement changer les exigences dans les référentiels et cahiers des charges.

Le premier résultat de la démarche GAP est la démonstration qu'*il est possible de spécifier les exigences d'évolution sous la forme d'écarts* entre modèles As-Is et To-Be. L'intérêt de cette manière de spécifier les exigences d'évolution est qu'elle est précise, et moins ambiguë que l'approche consistant à ne spécifier que les exigences cibles sans référence au changement que ces exigences induisent. Notre approche de spécification d'exigences d'évolution par opérateurs d'écart ne se substitue pas aux des typologies de buts d'évolution [Dardenne1993], [Anton1996], [Nurcan1999], [Prat1997]. Elles peuvent au contraire être employées de manière complémentaire. La typologie d'opérateurs d'écart proposée dans GAP se positionne au niveau des modèles, ce qui n'est pas systématiquement le cas avec les typologies de but d'évolution dont l'emploi se situe à un niveau plus macroscopique.

De plus, GAP se distingue par un cadre de travail original pour la définition de différents langages de spécification des exigences d'évolution. Certaines démarches de spécification de langages d'évolution proposent un langage unique global (comme les typologies de buts d'évolution citées ci-dessus), un langage spécifique défini de manière ad hoc [Soffer2004], ou un langage spécifique défini par transposition [Krishna2004]. Le cadre méthodologique sur lequel repose GAP, se positionne à un niveau générique. Il définit la question du développement d'un langage de spécification d'exigences d'évolution comme intimement liée au contexte méthodologique dans lequel ce langage sera utilisé. Pour ce faire, *le langage proposé est générique et adaptable*. L'adaptation se fait au moyen d'un méta-modèle générique qui peut être instancié par le méta-modèle de produit de n'importe quelle méthode. Outre son utilité dans l'adaptation de notre langage générique, le méta-modèle générique est utile pour garantir le caractère systématique du développement de la typologie générique d'opérateurs d'écart.

L'approche GAP se distingue par son aspect systématique. La plupart des autres approches ne garantissent que quelques-unes, parfois aucune des qualités de complétude, de correction, de consistance, de minimalité ou de richesse sémantique. Nous avons en particulier démontré que *les critères de qualité de complétude, de correction, de consistance et de richesse sémantique sont satisfaits*, soit au niveau générique, soit au niveau spécifique au travers de mécanismes d'instanciation de la typologie spécifique d'opérateurs d'écart qui les garantisse par construction. De cette façon:

- (a) chaque type d'opérateur d'écart spécifie un type d'exigences d'évolution;

- (b) tout écart peut être spécifié au moyen de l'un des opérateurs de la typologie spécifique;
- (c) l'application de chaque opérateur garantit la cohérence des modèles cibles (To-Be) sans introduire de nouvelles erreurs; et
- (d) les définitions des opérateurs d'écart sont systématiques, claires et non ambiguës.

GAP propose un *processus méthodologique qui guide le développement de langages de spécification des exigences d'évolution* qui (a) sont spécifiques aux méta-modèles employés dans les projets et (b) présentent les qualités requises comme indiqué ci-dessus. Ce processus méthodologique a été spécifié au moyen du formalisme MAP. Son déroulement a été illustré par un exemple simple mais réaliste dans le cadre des travaux de Anne Etien [Etien2006].

Enfin, nous avons démontré la scalabilité de notre démarche en l'appliquant réellement au sein de la filiale DIAC de financement de crédits automobiles dans le groupe Renault-Nissan. L'objectif du projet était de déployer le système d'information français de la DIAC dans les implantations étrangères; tout d'abord en Espagne et en Allemagne, puis à plus long terme dans d'autres pays – y compris non Européen. Notre mission a été de recenser les exigences d'évolution du système d'information de la DIAC. L'application de la démarche GAP a en premier lieu permis de valider ses principes élémentaires (spécification des exigences d'évolution par écarts, hypothèse de l'identité des méta-modèles pour modéliser le As-Is et le To-Be, cadre de travail au niveau générique permettant de développer des langages adaptés). Ce travail a aussi permis de confirmer que les qualités que l'approche considère sont bien celles requises. Par la nature réelle du projet, ce travail a enfin démontré la scalabilité de notre démarche a des projets de taille industrielle.

4. Co-évolution

L'évolution est en soi un sujet complexe. Une dimension de cette complexité est liée au fait que les systèmes fonctionnent et évoluent très rarement de manière totalement indépendante ou en autarcie. Chaque évolution d'un système peut avoir des répercussions désirables (ou indésirables) sur son environnement. Inversement, l'environnement évolue lui-même, ce qui peut avoir différentes sortes d'influences sur le système considéré. Les systèmes et organisations étant de plus en plus liées, les évolutions individuelles font place à des évolutions de véritables systèmes de systèmes. Tous les éléments doivent évoluer de manière harmonieuse, mais étant donnée la taille et la complexité des systèmes considérés, il est difficile voir impossible de procéder comme pour des systèmes simples.

Le terme *co-évolution* désigne l'évolution d'un ensemble d'entités, similaires ou différentes, liées les unes autres. Par exemple, Lehman fait apparaître que l'interdépendance croissante des systèmes est de plus en plus déterminante dans l'étude de leur évolution [Lehman2000]: “dans de nombreux cas, le phénomène d'évolution des applications [...] n'est pas autonome mais relève d'un phénomène de co-évolution. Les gouvernements, les entreprises et autres organismes faisant un usage toujours croissant de l'informatique pour les activités d'administration, de communication interne et externe, de marketing, pour garantir la sécurité, pour les activités techniques et autres, les applications deviennent inextricablement interdépendantes, partageant et échangeant des

données, invoquant les services les unes des autres, etc. La tendance indéniable est en direction de l'intégration de services internes et externes [...]. Et l'intégration s'étend graduellement aux systèmes des clients, fournisseurs et aux organismes de services, tels que les banques". On peut donc parler de co-évolution de systèmes interdépendants, tout comme on peut parler de co-évolution de fragments de codes cohérents, ou de co-évolution de spécifications d'un système qui relèvent de différents domaines d'ingénierie.

La recherche sur les problématiques soulevées par la co-évolution est relativement récente dans le monde de l'informatique. Un certain nombre de travaux ont été menés pour comprendre l'évolution conjointe de systèmes logiciels et (éventuellement) d'autres entités telles que des processus métier [Bodhuin2004] ou [Kardasis1998], ou les bases de données [Cleve2005]. [Zaidman2008] propose par exemple une approche pour assurer la co-évolution du code et des données de test. Peu de travaux ont été effectués plus précisément dans le périmètre de l'ingénierie des exigences. Des règles ont été proposées pour s'assurer que différents types d'exigences (formelles/informelles, fonctionnelles/non fonctionnelles, spécifiées par différents langages) restent cohérentes [Han1997], [Zowghi2003], [Krishna2004], [Bhuiyan2006]. [Bodhuin2004] propose des métriques pour évaluer et réduire le risque de dérives des entités impliquées dans la co-évolution. [Seybold2004] propose de guider la découverte d'exigences d'évolution par des techniques de simulation.

L'état des recherches nous a paru très morcelé, et nous avons découvert à la lecture des publications qu'il n'existe ni de vision d'ensemble généralisée de la co-évolution, ni de définitions précises des concepts sous-jacents (par exemple dans l'esprit de [Lämmel2004]) sur lesquelles s'appuyer pour développer de nouvelles méthodes, techniques et outils d'ingénierie des exigences qui s'inscrivent dans la problématique de la co-évolution.

La question de recherche que nous avons posée est donc:

“quels sont les fondements théoriques nécessaires pour développer de nouvelles méthodes techniques et outils d'ingénierie des exigences propres à la co-évolution?”

Considérer la co-évolution de manière spécifique est susceptible de présenter de nombreux avantages pratiques: meilleure analyse d'impact des exigences d'évolution, meilleure analyse des compromis des multiples scénarios d'évolution, meilleure gestion de la cohérence des évolutions, amélioration de la qualité de la documentation de l'architecture des systèmes d'information et de l'entreprise, moins de dérives de « désalignement », et une traçabilité *a posteriori* des composants du système d'information et de l'organisation de l'entreprise.

Sur un plan fondamental, nous espérons (a) mettre en évidence de manière systématique les concepts qui doivent formellement être définis pour développer une théorie de la co-évolution (b) explorer les lois fondamentales de l'évolution sous l'angle de la co-évolution, et (c) identifier de manière systématique l'ensemble des approches possibles d'ingénierie de la co-évolution.

Les recherches que nous avons menées sur ce thème sont donc de type exploratoire. Il s'est agi non pas d'inventer en soit de nouveaux concepts, méthodes, techniques ou outils, mais de poser les fondements théoriques nécessaires à leur définition.

4.1. *Problématique de la co-évolution*

Nous voyons trois obstacles à la bonne compréhension des problématiques liées à la co-évolution: le manque de fondements théoriques, la diversité des domaines d'application, et le manque de données empiriques.

i. Méconnaissance des problématiques spécifiques: la co-évolution est-elle un concept ou un processus? Peut elle être spécifiée, mesurée, est-il possible d'élaborer des modèles prédictifs? Nous ne disposons aujourd'hui pas de définition formelle des concepts fondamentaux à manipuler pour réaliser une ingénierie de la co-évolution, que ce soit par le génie logiciel, l'ingénierie des systèmes ou l'ingénierie des systèmes d'information. Les influences externes sont nombreuses, en particulier en raison de l'étude de l'évolution des groupes, en biologie, en économie, ou en sociologie. Par exemple, le modèle Darwinien d'évolution biologique repose sur un principe de "sélection naturelle" d'individus. Ce principe a été transposé en économie par H. Specer avec l'expression "survie du plus adapté". Or, comme nous l'avons montré dans la section précédente, les évolutions de systèmes sont délibérées: elles résultent d'exigences d'évolution. Les exigences d'évolution ne peuvent être entièrement réduites à de simples suppressions ou créations de propriétés de systèmes, comme peuvent l'être les phénotypes d'individus vivants : elles ont un propos, une raison fondamentale. Le modèle Lamarckien d'évolution biologique est à ce titre plus intéressant puisqu'il fait apparaître l'évolution sous l'angle du besoin des individus de s'adapter à l'environnement. Selon le modèle Lamarckien, l'évolution résulte de la transmissions de caractères acquis par des individus. Comme le montre Jean Molino, le modèle Lamarckien peut être appliqué au contexte de l'évolution culturelle. Il semble cependant inadapté aux systèmes à base de logiciels dans la mesure où -en l'état actuel des choses- il n'existe pas de forme de transmission de la capacité à satisfaire des exigences d'un système à l'autre. Il paraît peu probable que l'évolution des systèmes à base de logiciels puisse être étudiée sous l'angle de ces modèles. La méconnaissance actuelle de spécificités de la co-évolution, en particulier des problématiques spécifiques à la coévolution rend difficile l'élaboration d'un socle théorique permettant de développer de nouvelles méthodes, techniques et outils d'ingénierie.

ii. Diversité des domaines d'application: la coévolution est une préoccupation qui apparaît en ingénierie des systèmes d'information (par exemple lorsque le système d'information, l'organisation et le domaine Co évoluent), tout comme en ingénierie des systèmes (lorsque les spécifications de systèmes relevant de différents domaines d'ingénierie Co évoluent), ou même en génie logiciel (lorsque différents logiciels et composants de logiciels Co évoluent); tous sont des domaines dans lesquels se pratique l'ingénierie des exigences. Cependant, les problématiques soulevées dans ces différents domaines ne sont pas identiques. Par exemple, l'angle en ingénierie des systèmes est celui de la co-ingénierie [Pohl1994], alors qu'en génie logiciel, on se préoccupe plutôt de la gestion de configuration et de l'évolution de code. La diversité des domaines d'application est donc un obstacle important à une compréhension globale et précise de la problématique de coévolution. Nous pensons qu'il est cependant important d'embrasser cette diversité, dans la mesure où elle peut contribuer à mieux définir les fondements théoriques de la coévolution, et à s'assurer de la généralité des concepts définis.

iii. Manque de données empiriques: la coévolution étant un domaine qui n'a que très récemment soulevé l'intérêt des scientifiques, peu de données empiriques ont été pour l'instant publiées la concernant. Comme dans tout cycle de

recherche, la disponibilité de données empiriques est importante puisqu'elles offrent un appui pour diriger, confirmer ou infirmer la validité des théories et concepts. Or, on dispose de très peu d'études empiriques relatives à l'évolution sous l'angle de l'ingénierie des exigences ([Anton2001], [Anderson2001]), et malheureusement, la question de la coévolution n'y est pas abordée. Le manque de données empiriques spécifiques au problème de la coévolution, est cruciale, dans la mesure où, dans un premier temps, ce sont les données empiriques (retours d'expérience, expérimentations, observations in situ de type ethnographiques, etc.) qui offrent le point d'appui nécessaire pour élaborer les premiers modèles et à évaluer la qualité des prédictions qu'ils permettent de produire.

4.2. Principes du cadre de travail de la coévolution

Nous proposons un cadre de travail qui définit les qualités attendues d'une approche d'ingénierie des exigences qui vise à *comprendre la coévolution* et à *la mettre en œuvre*. Ce cadre est structuré autour de cinq dimensions. Chaque dimension correspond à un problème clef d'ingénierie des exigences que nous percevons comme crucial pour comprendre et mettre en œuvre la coévolution. Différentes approches peuvent être prises pour résoudre chaque problème-clef. Ces approches sont présentées comme des valeurs dans les dimensions coïncidentes. Toute méthode d'ingénierie des exigences qui s'intéresse à la coévolution peut être positionnée dans le cadre en identifiant quelle valeur elle prend dans chacune des cinq dimensions.

2.4.a Comprendre la coévolution par les relations des entités qui co-évoluent et l'expression des exigences d'évolution

Un problème clef lorsque l'on entreprend un projet d'évolution de système est la maîtrise du lien par lequel les entités qui co-évoluent sont associés. Un autre problème est d'exprimer les exigences qui définissent spécifiquement les évolutions de ces différentes entités de manière adéquate. Ces problèmes sont respectivement abordés par le cadre dans les dimensions intitulées *comprendre les relations* et *exprimer les exigences d'évolution*. Ces deux dimensions relatives à la compréhension de la coévolution sont respectivement abordées dans le chapitre de ce mémoire qui concerne le thème de l'alignement, et dans l'autre partie du présent chapitre qui présente l'approche GAP.

2.4.b Mettre en œuvre la coévolution par: l'identification des exigences d'évolution, la propagation du changement, et la vérification des relations entre les entités cibles

Trois dimensions ont été définies relativement à la mise en œuvre de la coévolution: l'identification des exigences d'évolution, la propagation du changement, et la vérification des relations entre les entités cibles

La coévolution fait intervenir à la fois une analyse d'impact et une propagation du changement. Le but de la propagation du changement est de transposer les exigences d'évolution dans le To-Be [Han1997]. L'analyse d'impact vise à évaluer comment les exigences d'évolution peuvent affecter la cohérence de chaque entité qui

évolue, que ce soit de manière interne ou par rapport aux autres entités. La principale caractéristique distinctive d'une démarche d'*identification des exigences* spécifique au contexte de la coévolution est l'analyse d'impact: comment identifier des exigences d'évolution tout en tenant compte du fait que plusieurs autres entités doivent évoluer en même temps? La Figure 9 montre que les approches d'analyse d'impact peuvent être divisées en quatre familles, à savoir: indépendance, interdépendance, dépendance, et double-dépendance. Chaque famille est définie selon la direction des relations de dépendance entre les entité qui co-évoluent.

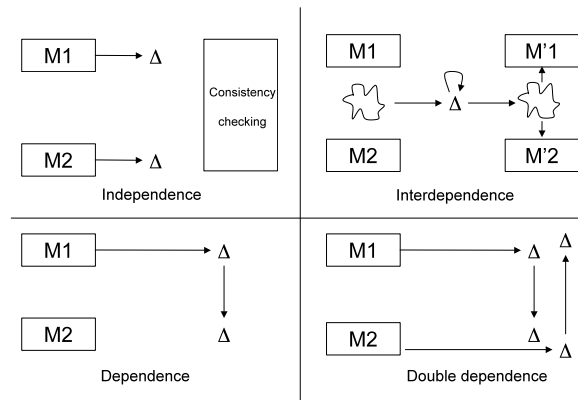


Figure 9: Typologie des démarches d'identification des exigences en fonction de l'approche d'analyse d'impact.

La coévolution est réalisée de manière indépendante (coins supérieur gauche de la Figure 9) quand il n'y a pas de dépendance entre les processus d'ingénierie des différentes entités qui évoluent. C'est typiquement le cas dans les projets soumis à une forte pression: les exigences d'évolution sont implémentées dans le système sans vérifier la cohérence avec l'organisation cible. Cette approche est aussi rencontrée dans les projets d'évolution qui portent sur des systèmes voués à disparaître à court terme, ou bien encore lorsque les exigences d'évolution sont elles-mêmes hautement volatiles et susceptibles d'être rapidement abandonnées.

L'approche indépendante de coévolution va souvent de paire avec les projets dans lesquels l'évolution est implicite, ou bien lorsque le nombre des entités qui co-évoluent est trop grand. Cette situation est bien illustrée par les projets d'ingénierie des systèmes dans lesquels différentes disciplines d'ingénierie interviennent. Non seulement les exigences d'évolution sont souvent implicites (différents documents de spécification des exigences sont parfois livrés à différents moments), mais aussi il arrive qu'il soit techniquement difficile de comprendre comment une évolution requise dans une discipline impacte le système dans la perspective des autres disciplines.

Pour résoudre ce problème, la cohérence peut être contrôlée rétrospectivement afin de vérifier que le nouveau système est aligné avec les autres entités [Bodhuin2004]. L'alignement peut aussi être évalué au moyen de métriques [Mens2001]. Si une rupture d'alignement est observée, d'autres évolutions sont requises et un nouveau cycle d'ingénierie peut être réalisé. C'est typiquement ce qui arrive dans les projets d'intégration d'ERP dans lesquels l'adaptation d'une transaction individuelle doit être implémentée pour pouvoir vérifier au travers de tests de non-régression si il y a des impacts ou pas sur d'autres transactions ou modules.

Dans l'approche dépendante d'identification des exigences d'évolution, les exigences d'évolution d'une entité sont déduites à partir des exigences d'évolution identifiée pour une autre entité qui co-évoque. C'est typiquement le

type d'approche prise dans un projet d'amélioration de processus métier quand les exigences d'évolution du système d'information sont générées pour se conformer aux évolutions requises au niveau du métier. Cette approche est aussi fréquemment prise par les managers de portefeuilles de projets qui définissent des projets maîtres auxquels les autres projets d'évolution du système d'information doivent se conformer. Cette approche est formalisée par les approches d'ingénierie dirigée par les modèles. Au niveau des exigences, des règles sont par exemples proposée par [Krishna2004] pour déduire les exigences d'évolution sur les spécification du systèmes exprimées en Z à partir d'évolutions exprimées au niveau intentionnel par des modèles I*. Dans une approche dépendante, la cohérence est vérifiée comme une propriété externe où les exigences dépendantes doivent se conformer à une entité dominante, par exemple la connaissance de domaine dans [Zowghi2003].

Il y a double dépendance lorsque chaque entité qui co-évolue peut jouer un rôle dominant dans l'analyse d'impact des exigences d'évolution. Par exemple, [Kardasis1998] propose des règles pour évaluer l'impact d'évolutions requises au niveau métier sur les exigences d'évolutions du système d'information et vice versa. Une approche de double dépendance peut être considérée comme la combinaison de deux approches de dépendance mono directionnelles.

A l'inverse, une approche interdépendante peut être considérée comme bi-directionnelle. En effet, dans ce type d'approche, chaque exigence d'évolution spécifie en même temps comment toutes les entités doivent co-évoluer. Dans ce cas, l'analyse d'impact et la propagation sont réalisée par une seule collection d'exigences d'évolution. Cela peut être réalisé en utilisant des invariants [Banerjee1987] ou en employant des règles heuristiques comme le suggère [Han1997].

Le processus de développement du résultat de la coévolution est appelé *propagation du changement* [Han1997]. Une fois que les exigences d'évolution ont été identifiées, les spécifications du nouveau système doivent être produites. Ce problème s'intéresse donc à la manière de concevoir les modèles To-Be. Nous avons trouvé que cette partie du processus de coévolution est rarement décrite dans la littérature ou documentée dans les projets. Deux familles d'approches peuvent néanmoins être distinguées: les approches symétriques et les approches asymétriques.

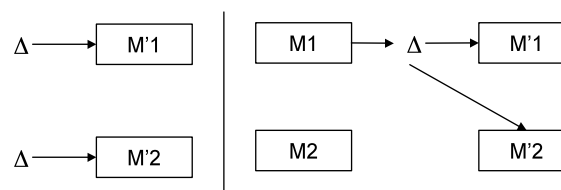


Figure 10: Principe de la propagation symétrique et asymétrique des exigences d'évolution

L'approche la plus courante est illustrée dans la partie gauche de la Figure 10. Elle consiste à combiner les exigences d'évolution avec le modèle As-Is de chaque entité qui évolue, de manière à obtenir les modèles To-Be coïncidants [Krishna2004]. Cette approche est appelée *propagation symétrique* car les exigences d'évolution spécifiques sont combinées symétriquement à chacune des entités qui co-évolue [Lämmel2004].

Dans l'approche de *propagation asymétrique*, les exigences d'évolution destinées à générer les modèles To-Be d'une entité sont aussi utilisées pour les modèles To-Be de l'autre entité en coévolution [Lämmel2004].

Le dernier problème abordé par notre cadre méthodologique est celui de la *vérification des relations* entre les entités cibles. En effet, selon notre cadre méthodologique, la spécificité de la coévolution est de faire évoluer différentes entités en tenant compte des relations qui les associent. Les relations peuvent servir a priori à guider la découverte d'exigences d'évolution, ou a contrario, à vérifier a posteriori, à faire le bilan de la coévolution. Un tel bilan doit être réalisé de manière systématique au moyen de techniques de vérification telles que des inspections, de l'échantillonnage ou des mesures.

4.3. Résultats

Deux des articles que j'ai produits sur le thème de la coévolution peuvent être distingués dans la mesure où ils ont été publiés dans des conférences internationales majeures RE'05 (International IEEE Conference on Requirements Engineering) [Etien2005] et CAiSE'03 (International Conference on Advanced information Systems Engineering) [Salinesi2003].

Ces deux articles présentent respectivement deux résultats majeurs sur ce thème :

- Un cadre méthodologique de la coévolution ; et
- Une typologie des classes de problèmes d'ingénierie que l'on peut associer à la thématique de la coévolution.
- A ces deux résultats vient s'ajouter un troisième sous la forme d'une démarche méthodologique guidant l'identification des exigences par analyse de la coévolution.

Ce dernier est présenté dans la thèse d'Anne Etien, ainsi que dans les Conférence ICSoft (International Conference on Software Engineering and Data Technologies) [Etien2006b] et ACIS (Australasian Conference on Information Systems) [Salinesi2004d].

Le *cadre méthodologique* que nous avons développé permet de *comprendre les éléments fondamentaux de la coévolution en ingénierie des exigences* sous l'angle de cinq dimensions: (i) la compréhension des relations entre les entités qui co-évoluent, (ii) l'identification des exigences d'évolution, (iii) l'expression des exigences d'évolution, (iv) la propagation du changement, et (v) la vérification des relations entre les entités cibles. Chaque dimension nous a permis de définir et d'explorer une problématique spécifique à la coévolution en ingénierie des exigences. L'étude de la littérature nous a permis d'identifier différentes approches de résolution de ces problématiques, ce qui nous a permis d'établir un panorama des positionnements en utilisant le cadre méthodologique comme moyen de référence. Ce cadre méthodologique permet (a) de comparer les approches existantes, (b) de faire apparaître les problèmes fondamentaux que doit considérer une démarche d'ingénierie des exigences tenant compte des problématiques spécifiques de la coévolution, (c) de définir les caractéristiques essentielles d'une démarche lors de son élaboration.

Une étude systématique de la coévolution sous l'angle du lien d'alignement nous a permis de définir une *typologie des classes d'ingénierie des exigences dirigées par la coévolution des métiers et du système d'information* [Salinesi2003].

Relativement à l'expression des exigences d'évolution, notre cadre montre la nécessité de *langages pour spécifier explicitement les exigences d'évolution*. De tels langages sont nécessaire non seulement parce que la spécification est une condition essentielle si l'on veut pouvoir raisonner sur les exigences d'évolution, mais aussi parce que c'est le seul moyen de mettre en place la traçabilité des entités qui co-évoluent.

Notre cadre propose une typologie des démarches d'identification des exigences d'évolution. Nous avons utilisé cette typologie comme socle théorique pour développer la *démarche méthodologique ACEM qui guide la découverte d'exigences d'évolution par l'étude de la relation d'alignement entre les entités qui coévoluent*.

De plus en plus de démarches d'ingénierie des exigences sont développées pour traiter de la question de la coévolution. Cependant, notre étude bibliographique nous a permis de montrer (a) qu'il n'y a pas de manière unique d'aborder la coévolution; le cadre montre bien que des avantages et des désavantages peuvent être rencontrés dans toutes les approches et ce en fonction de la situation des projets dans lesquelles elles sont employées. Nous pensons qu'une intégration de méthode situationnelle pour aider à définir et intégrer l'approche capable de traiter de différentes situations de coévolution. (b) certains aspects de la coévolution ne sont pas bien traités. La raison peut être triple: soit notre état de l'art n'est pas complet, soit la problématique n'est pas si importante que nous le présentions, soit la problématique est effectivement importante mais mal traitée et devrait donc l'être dans de futures recherches. Notre expérience dans plusieurs projets industriels nous amène à penser que c'est probablement le cas. Néanmoins, nous avons observé qu'il n'y a à ce jour pas de fondements théoriques sur lesquelles s'appuyer pour développer de nouvelles méthodes de manière systématique. Les problématiques considérées sont éparées, les concepts mal définis, les terminologies parfois contradictoires, et le lien avec les autres domaine d'ingénierie n'est que partiel.

Notre cadre méthodologique, publié dans [Etien2005] est une première étape vers l'élaboration des fondements théoriques nécessaires pour développer de nouvelles méthodes techniques et outils d'ingénierie des exigences propres à la coévolution. C'est sur ce cadre méthodologique que s'est reposée notre analyse lors de la définition des principes fondamentaux de la démarche méthodologique ACEM. Le cadre nous a en outre permis d'établir le positionnement de la méthode ACEM par rapports aux autres démarches existantes.

5. Perspectives de recherche

En 2005, les discussions menées lors de l'atelier sur les challenges de l'évolution des logiciels organisé par le Software Evolution Working Group de l'ERCIM (ChaSE 2005) ont fait apparaître parmi les 18 problèmes clef de recherche à traiter en priorité [Mens2005]:

- "définir de nouveaux langages de programmation ou de modélisation dans lesquels la notion de changement soit une entité de première classe", et

- "supporter la coévolution de différents types d'artefacts".

Nos travaux sont donc en ligne avec les courants de recherche dans le domaine.

De nombreuses directions de recherche sont à explorer de manière immédiate: définir formellement les liens qui unissent les exigences d'évolution avec les entités qui co-évoluent, enrichir l'approche GAP pour diversifier la typologie des exigences d'évolution, élaborer des modèles prédictifs de la coévolution et les valider empiriquement, développer de nouvelles démarches guidant la découverte des exigences d'évolution et appréhender leur lien avec les multiples artefacts manipulés lors de projets d'ingénierie.

Définir formellement les liens qui unissent les exigences avec les entités qui co-évoluent: La nature fondamentale de la relation qui unit les entités qui co-évoluent est déterminante car c'est de cette relation que dépend en très grande partie les principes mis en avant dans les démarches d'ingénierie des exigences qui s'intéressent à la coévolution. Nous avons commencé à explorer cette piste de recherche en nous intéressant à la relation d'alignement [Salinesi2003]. Nos travaux sur le thème de l'alignement sont présentés au chapitre suivant. La relation d'alignement n'est cependant pas l'unique concept que l'on puisse définir et exploiter. Par exemple, [Lämmel2004] définit les couples d'entités qui co-évoluent par la relation de cohérence qui les unit. Les travaux de A. Persson [Dahlstedt2005] sur les nombreux types d'interdépendances employées en ingénierie des exigences ouvrent un champs d'investigation intéressant et prometteur à explorer sous l'angle de la coévolution.

Enrichissement de l'approche GAP pour diversifier la typologie des exigences d'évolution: l'ingénierie des exigences distingue différents types d'exigences: exigences fonctionnelles, non-fonctionnelles, législatives, contraintes, exigences aspectuelles, etc. [Glinz2005]. La méthode GAP au travers des forces contextuelles ne permet de formuler que des exigences fonctionnelles. La méthode GAP pourrait être enrichie par la définition d'exigences d'évolution exprimées sous forme qualitative ou non fonctionnelle.

Elaboration de modèles prédictifs de la coévolution et validation empirique: une grande difficulté lors de l'identification d'exigences d'évolution est la projection dans le futur. Plus encore dans le cadre de la coévolution, il s'agit d'anticiper comment l'ensemble des entités doit co-évoluer. Il est cependant important de savoir prédire la coévolution. L'élaboration de modèles prédictifs permettrait de guider la découverte d'exigences d'évolution de manière anticipée. Ces modèles pourraient servir de base au développement de méthodologies innovantes. La disponibilité de données empiriques est un facteur clé pour élaborer des modèles prédictifs valides. Un axe de recherche à explorer est donc la recherche de retours d'expériences. Ces retours d'expériences pourront aussi servir à la création d'études de cas qui permettraient de valider les méthodes existantes telles que [Avison1994], [Moreira2006], [Anderson2002], [Ruiz2004], ou celle que nous avons nous même développée. Ces études de cas peuvent aussi servir de base à l'élaboration de protocoles d'expériences empiriques contrôlées.

Elaboration de directives relatives à la propagation des écarts sur les modèles de l'organisation: la répercussion des exigences d'évolution sur les différentes entités qui co-évoluent peut varier en fonction de leur nature. Nous avons élaboré une démarche méthodologique qui propose un certain nombre de directives pour guider la propagation des écarts sur des modèles de système et de processus [Etien2005]. On peut néanmoins se

demander si des règles ou des directives génériques ne pourraient pas être identifiées permettant ainsi de définir des principes fondamentaux de l'ingénierie de la coévolution.

L'intérêt des chercheurs pour les problématiques de coévolution ne cesse de croître. En génie logiciel, le problème est apparu dès que l'on a cherché à garantir la cohérence des évolutions de codes disparates. En ingénierie des systèmes, il a été évoqué de manière récurrente dans le contexte de ce que l'on appelle la co-ingénierie. En ingénierie des systèmes d'information, le problème de coévolution apparaît au cœur des préoccupations de directeurs de systèmes d'information qui veulent que les organisation et les systèmes d'information puissent évoluer rapidement et de manière synergique. Encore très récemment, l'étude de projets de développement et de maintenance de logiciels open source [Scacchi2006] a démontré que la problématique de la coévolution est au cœur de l'organisation de ce type de projet. Plus généralement, on peut s'attendre à ce que l'expansion des technologies émergentes telles que les architectures orientées service, ou la programmation par aspects aillent dans le sens de développements de types nouveaux dans lesquels les problèmes de coévolution apparaîtront dès lors que l'évolution des systèmes est tirée par plusieurs parties prenantes. Nous avons donc bon espoir que la problématique de la coévolution continue d'attirer l'attention pendant quelques temps.

CHAPITRE 5

Alignement

1. Introduction et positionnement de mes travaux sur l'alignement

La distinction du "quoi" et du "pourquoi" qui est au centre des approches d'ingénierie des exigences a naturellement amené la communauté à s'intéresser à l'alignement entre les systèmes d'information et la stratégie ou le business d'entreprise.

Le rapport [CSC2001] montre que la problématique de l'alignement du systèmes d'information représente 54.2 % des préoccupations exprimées des directeurs de systèmes d'information et que l'alignement du système d'information figure en deuxième place parmi les facteurs qui contribuent le plus au succès du système d'information dans les entreprises.

Bien que l'intérêt de l'alignement soit largement reconnu, sa mise en œuvre reste trop souvent limitée dans les faits. Comme le montre l'étude récente de PAC, peu de dirigeants considèrent que le système d'information et les processus de leur organisation sont alignés [Pujol2008]. [Luftman2007] identifie deux causes principales: (i) les acteurs de l'organisation ne savent pas ce qu'est l'alignement et (ii) il y a une absence de communication et de compréhension entre le monde du "business" et celui des technologies de l'information. Cette analyse est confirmée par les études telles que le rapport du [CIGREF2002] qui met en évidence un écart entre les attentes des directions (générales, métiers et SI) en matière de contribution des SI à la création de valeur pour l'entreprise. Cet écart s'explique par des problèmes de relation entre les parties prenantes: (a) manque de coordination et de communication, (b) difficultés à formuler les besoins métiers et à les traduire sous une forme exploitable par les équipes SI et (c) faible capacité à engager les changements nécessaires dans les métiers afin de tirer les bénéfices escomptés des SI.

2. Positionnement

Un nombre croissant de recherches s'intéressent à l'alignement [Henderson1993], [Luftman2007], [Wegmann2007], [Krishna04], [Bodhuin04]. Plusieurs ateliers de recherche ont été consacrés à ce thème, comme par exemple BPMDS 2004 ou REBNITA 2005. Les approches méthodologiques⁷ abordent le plus souvent l'alignement comme un processus de transformations successives qui convertissent les modèles de processus organisationnels ou l'expression de la stratégie en modèles du SI. L'alignement est alors obtenu implicitement au travers de l'application de règles méthodologiques de transformation.

⁷ Beaucoup de travaux qui relèvent du domaine du management des systèmes d'information se contentent en fait de définir la notion d'alignement et de l'étudier, sans proposer de méthodologie d'ingénierie de système d'information qui en fasse un exploitation systématique. Les démarches méthodologiques comme celle de [Wieringa] ou de [XXX] sont beaucoup moins nombreuses que les études empiriques.

Ma position est d'*étudier l'alignement comme un concept et donc de le modéliser*. Dans cette perspective, il est nécessaire de trouver des langages d'alignement, formalisés par des méta-modèles d'alignement. De tels méta-modèles doivent permettre de spécifier des alignements complexes, par exemple partiels, à différents niveaux d'abstraction et de détail, et/ou entre des éléments conceptuellement différents.

D'un point de vue pratique, il y a plusieurs intérêts d'appréhender l'alignement au moyen de modèles conceptuels. Tout d'abord, cela permet de mieux vérifier la cohérence des nombreuses vues à gérer: celle de la stratégie avec le SI, celle du métier avec le SI, celle des exigences et de l'architecture, etc. De plus, l'approche conceptuelle permet d'introduire des automatismes dans l'analyse des différents modèles employés, et par la suite de réduire le besoin de coordination manuelle et de résolution de conflits.

D'un point de vue théorique, l'élaboration de méta-modèles d'alignement présente un triple intérêt.

- Tout d'abord, cela permet de donner une définition formelle du concept d'alignement. Dans beaucoup d'approches existantes, l'alignement est défini en des termes vagues et sujets à interprétation. Par exemple, dans certain cas l'alignement est défini comme une forme de cohérence (« A et B sont alignés s'ils ne sont pas en contradiction »), alors que dans d'autres, il s'agit d'une forme de contribution avec une notion de directionnalité ou de dominance (« A est aligné avec B s'il fait ce que B commande de faire »). Les différentes définitions font apparaître des différences de point de vue sur ce qu'est l'alignement. Seuls des méta-modèles peuvent permettre de faire apparaître de manière précise la spécificité de chacun de ces points de vue.
- Le deuxième intérêt est directement lié au premier. Puisque les méta-modèles font apparaître plusieurs définitions de l'alignement, ils peuvent permettre d'élaborer une véritable typologie de l'alignement. Une telle typologie est utile à la compréhension des différences et similarités entre les différentes formes d'alignement. Elle peut aider à choisir la forme d'alignement désirée dans une situation particulière, ou pour l'intégrer à une méthodologie d'ingénierie des systèmes d'information.
- Le troisième intérêt réside dans le fait que les méta-modèles sont une base nécessaire à l'élaboration de techniques et d'outils de raisonnement systématique. Par exemple, il paraît difficile d'aborder la question de l'alignement sans outil pour le mesurer. Nous pensons que disposer de méta-modèles d'alignement est utile pour définir des métriques qui permettent de mesurer des degrés d'alignement.

Les deux sections ci-après présentent deux approches de modélisation que nous avons développées:

- ACEM s'intéresse à la modélisation entre des couples d'entités. Dans la démarche méthodologique ACEM, l'alignement est vu comme un lien entre les paires d'entités concernées. Un modèle d'alignement est un modèle qui fait le pivot entre les éléments de deux modèles. Des mesures ont par ailleurs été proposées afin d'enrichir la connaissance de la nature du lien d'alignement avec des données quantitatives. Le méta-modèle d'alignement proposé est générique. Le méta-modèle ACEM définit formellement les liens d'alignement entre des couples d'entités positionnées dans des ontologies. ACEM propose une démarche d'adaptation qui permet d'intégrer le méta-modèle d'alignement proposé dans

une méthode, par identification des concepts de la méthode, avec les concepts identifiés dans les ontologies. D'un point de vue applicatif, nous nous sommes en particulier intéressés à l'alignement entre un système d'information et un processus d'entreprise. Outre le méta-modèle d'alignement, ACEM propose une démarche méthodologique qui guide la découverte d'exigences d'évolution du système d'information par l'analyse de son alignement avec les processus métier de l'entreprise.

- Dans la démarche méthodologique INSTAL, un modèle d'alignement est non pas un pivot constitué de liens entre les éléments de deux modèles, mais un modèle global qui subsume les entités alignées. La position prise dans l'approche INSTAL est que deux entités sont alignées si elles partagent une intention commune d'alignement. Le méta-modèle d'alignement INSTAL est donc un méta-modèle intentionnel qui a la particularité de permettre de représenter les intentions de différentes entités. D'un point de vue applicatif, le méta-modèle INSTAL a été utilisé pour modéliser l'alignement stratégique du système d'information. Outre le méta-modèle d'alignement intentionnel, INSTAL propose une démarche méthodologique qui permet de guider la découverte d'exigences d'évolution du système d'information par l'analyse de modèles d'alignement stratégique.

3. Modélisation et mesure de l'alignement – l'approche ACEM

A ce jour, la littérature ne fait pas apparaître de consensus sur la définition de l'alignement. En fait, une analyse fine des différentes définitions montre que celles-ci sont équivoques, parfois tautologiques ou contradictoires, et que certains concepts sont totalement théorique et sans lien avec la réalité de terrain [Maes2000].

En fait, plus d'une dizaine de termes sont employés dans la littérature pour parler du lien entre le système d'information et les organisation [Wegman2007]. Ainsi, on nomme ce concept aussi bien: “fitness relationship” [Potts1997] et “matching” (qui sont des expressions anglaises difficilement traduisibles en français), que “alignement” [Wegmann2005], “pont” [Ciborra1997], “harmonie” [Luftman1996] ou “fusion” [Smaczny01], “lien” [Henderson1993], mais aussi “coïncidence” [Knoll1994] ou “congruence” [Bergeron1999].

Les différentes définitions de l'alignement présentent tout de même un certain nombre de similitudes:

- Dans tous les cas, la notion même d'alignement implique l'existence de plusieurs entités liées entre elles. Le plus souvent, la relation d'alignement concerne deux entités, bien que certaines approches, en particulier dans le courant de l'urbanisme des Systèmes d'Information, étudient l'alignement entre plus de deux entités (par exemple cinq dans [Wieringa2003]).
- En général, l'une des entités considérées est relative à un système à base de logiciel: composant logiciel, Système d'Information, ou même stratégie des technologies de l'information.
- La préoccupation des parties prenantes (en tout cas côté Système d'Information) est de gérer ensemble ces entités: lors de leur construction, lors de leur gestion au quotidien (pour éventuellement réajuster cette relation), lors de projets d'évolution ou tout simplement pour les en préserver la connaissance.

Connaître ces points communs ne suffit cependant pas pour définir des démarches méthodologiques d'ingénierie de Systèmes d'Information basées sur une analyse fine de l'alignement. La question de recherche que nous avons soulevée est donc:

"quelle définition formelle de l'alignement permettrait d'en faire une modélisation conceptuelle utile à l'ingénierie des Systèmes d'Information?"

Plutôt que de nous intéresser à toutes les formes d'alignement, je me suis en premier lieu penchés spécifiquement sur l'alignement entre un processus d'entreprise et un Système d'Information. Dans cette section, j'emploie donc le terme « alignement » pour parler de l'alignement entre « processus » et « système ».

Dès le départ, mon approche a été de considérer l'alignement comme une activité d'ingénierie à part entière. Cela nécessite (i) une définition claire de la notion d'alignement, (ii) des langages pour le représenter, (iii) des méthodes pour gérer l'alignement (c'est-à-dire construire des processus et les systèmes qui les supportent de façon alignée, permettre l'évolution de ces entités sans rompre l'alignement...)

3.1. Problématique liée à la modélisation et à la mesure de l'alignement

La modélisation de l'alignement présente selon moi six problématiques fondamentale que je réparties en deux grands groupes. Le premier groupe relève du concept d'alignement lui-même. Il compte trois problématiques:

- i. Absence de définition précise de l'alignement ;
- ii. Difficulté de distinguer l'alignement du non-alignement ; et
- iii. Absence de représentation de l'alignement.

Le second groupe rassemble des problématiques liées à l'ingénierie de l'alignement:

- iv. Difficulté à construire des entités alignées ;
- v. Difficulté à rétablir l'alignement ; et
- vi. Absence de méthode de maintien de l'alignement face aux changements.

i. Absence de définition précise de l'alignement: Il n'existe à ce jour pas de définition communément acceptée de l'alignement. En 1989, Henderson considérait que les caractéristiques du concept d'alignement n'étaient pas clairement définies [Henderson1989]. Quinze ans plus tard, en 2004, Avison estime que la notion d'alignement reste floue, tant au niveau stratégique (entre stratégies d'entreprise et stratégies des technologies de l'information) qu'au niveau opérationnel (entre système et processus) [Avison2004]. L'absence de définition précise de ce concept explique peut être aussi l'absence de consensus sur le terme utilisé, les contradictions, et les ambiguïtés de certaines définitions [Maes2000]. Cette situation est cependant un obstacle à l'élaboration de fondements théoriques de l'alignement qui permettent de développer de véritables démarches méthodologiques d'ingénierie fondées sur ce concept.

ii. Difficulté à distinguer l'alignement du non-alignement: Les chercheurs, les dirigeants d'entreprise, les consultants et les utilisateurs s'accordent à dire qu'une rupture de l'alignement entraîne une baisse de la performance de l'organisation [Henderson1989]. L'état actuel de la recherche sur l'alignement ne permet pas de déterminer pour un projet donné, à un instant précis, si les processus et le système sont alignés. Pourtant, cette information pourrait (i) servir aux décideurs de critère de décision au même titre que le coût ou la flexibilité pour choisir parmi les cibles alternatives de conception ou (ii) les guider dans l'obtention d'une telle situation.

iii. Absence de représentation de l'alignement: Très peu d'approches considèrent la relation d'alignement comme un concept en soi [Maes2000], [Salinesi2003]. Elle est donc rarement représentée. Cette absence de représentation de la relation d'alignement montre (i) d'un point de vue conceptuel que les chercheurs ne pensent pas en termes d'ingénierie de l'alignement et (ii) d'un point de vue pragmatique, que les modèles de processus et les modèles du système sont traditionnellement exprimés dans des langages différents, à des niveaux d'abstraction distincts et dans des documents séparés. Les modèles de processus utilisent des concepts tel que but, processus, acteur et rôle, alors que les modèles du système décrivent des objets, des opérations, des événements, etc. Les fonctionnalités du système sont spécifiées de façon précise et détaillée à un bas niveau d'abstraction, le niveau opérationnel. Les processus sont représentés à un haut niveau d'abstraction, le niveau intentionnel.

Sans moyen pour représenter l'alignement, il est difficile de raisonner sur ce concept, de construire ou de faire évoluer un système et un processus alignés.

iv. Difficulté à construire des entités alignées: De nombreux directeurs des Systèmes d'Information considèrent que la stratégie d'entreprise et la stratégie relative aux technologies de l'information de leur organisation ne font qu'un. Pourtant, ils sont également nombreux à considérer qu'au niveau opérationnel, systèmes et processus ne sont pas alignés [Pujol2008]. Ceci relève pour partie du fait que ces deux entités n'ont pas été construites de manière alignée.

La difficulté à construire un système et un processus alignés résulte en grande partie de l'utilisation de langages différents, de niveaux d'abstraction différents et de modèles structurés différemment pour documenter chacune de ces deux entités [Salinesi2003].

v. Difficulté à rétablir l'alignement: On constate fréquemment qu'au cours de leurs cycles de vie, le système et les processus ne sont plus alignés. Cela arrive par exemple, quand les processus s'adaptent pour répondre aux objectifs des dirigeants, mais ne sont plus gérés convenablement par le système. Il est donc nécessaire de rétablir l'alignement en modifiant le système et éventuellement les processus afin de mieux faire coïncider ces deux entités. Peu d'approches proposent de guider les décideurs, les experts systèmes et les utilisateurs dans une telle démarche.

vi. Absence de méthode de maintien de l'alignement face aux changements: Les entreprises sont de plus en plus soumises à des changements. Si elles veulent rester compétitives, elles doivent faire évoluer leur système et/ou leur processus pour atteindre une nouvelle situation favorable à la synergie. Maintenir l'alignement dans un contexte d'évolution suppose (a) d'identifier l'impact des changements d'une entité sur l'autre, (b) de savoir propager ces

changements et (c) de déterminer quand a lieu cette propagation. En d'autres termes, l'évolutivité de l'alignement impose d'être capable de faire évoluer conjointement le Système d'Information et les processus.

3.2. *Principes de l'approche ACEM*

ACEM (Alignment Evaluation and Correction Method) est une démarche méthodologique dont l'objectif est de guider la correction et l'évolution de l'alignement entre un Système d'Information et des processus d'entreprise. L'élaboration de cette démarche méthodologique a nécessité une définition précise et formelle du concept d'alignement.

2.3.a **Modélisation de la relation d'alignement au moyen de liens**

Une définition formelle de la notion d'alignement suppose d'explicitier les types de liens qui peuvent exister entre les différents éléments de ces modèles. Quatre typologies de liens [Ralyté2004], [Pohl1996], [Bunge1977] et [Bodhuin2004] proposent une définition du concept d'alignement adaptée au domaine de l'ingénierie des Systèmes d'Information. L'étude de ces typologies a permis de faire apparaître deux types de liens: (i) ceux qui permettent d'exprimer une similarité entre concepts et (ii) ceux spécifiant une dépendance entre concepts. A partir de ces deux types de liens, nous avons défini les liens *correspond* et *représente* que nous utilisons pour définir le concept de *relation d'alignement*.

- Le lien *correspond* exprime une égalité entre des concepts similaires de méta-modèles différents. Le lien correspond (i) est à la fois proche du lien 'similaire' de Pohl et de la notion de similarité de Bunge et (ii) est une spécialisation du lien d'Association défini dans [Ralyté2004]. En effet, comme le lien 'similaire' de Pohl, le lien correspond permet de relier deux concepts similaires. Nous utilisons la notion de similarité de Bunge qui exprime le degré de similarité entre deux choses pour pouvoir dire formellement si deux concepts *coïncident*.
- Le type de lien *représente* spécifie qu'un concept d'un méta-modèle a un impact sur un concept d'un autre méta-modèle. Le lien représente est une spécialisation du lien d'association entre éléments de méta-modèles différents défini dans [Ralyté2004]. Ce lien s'appuie sur la notion de couplage de Bunge, sur les liens 'repose sur' de Pohl, 'dépend_de' et 'utilise' de Bodhuin.
- On appelle *relation d'alignement* entre un modèle de système et un modèle de processus l'ensemble des liens « correspond » entre un élément du modèle de système et un élément du modèle de processus et des liens « représente » entre un élément du modèle de système et un élément du modèle de processus, tels qu'il n'existe pas de lien « correspond » faisant intervenir au moins l'un des concepts du méta-modèle de système ou du méta-modèle de processus que ces élémentsinstancient.

2.3.b Mesure de l'alignement système-processus au moyen d'une typologie générique de métriques

Pour pouvoir corriger l'alignement entre un système et des processus, il est nécessaire de savoir distinguer l'alignement du non-alignement. J'ai observé que dans les faits, système et les processus ne sont souvent que partiellement alignés. Il existe donc en réalité toute une gamme de possibilités entre "parfaitement" et "pas du tout" aligné. Il est possible de saisir ces nuances, au moyen d'un *ensemble de métriques*. Notre proposition pour mesurer l'alignement repose sur un cadre organisé, comme celui de [Cavano1988], autour des notions de *facteurs*, *critères* et *métriques*. Nous avons défini quatre facteurs selon lesquels la relation d'alignement peut être mesurée. Chaque facteur a des critères qui lui sont associés. Ces derniers sont, à leur tour, associés à des métriques qui permettent de mesurer le degré d'alignement entre le modèle de processus d'entreprise et le modèle du système qui les supporte. Dix critères et dix métriques ont été identifiés.

Toutes les métriques sont construites de la même façon:

- elles utilisent (i) les concepts du méta-modèle de processus et du méta-modèle de système et (ii) les liens *correspond* ou *représente*.
- Ce sont des fonctions prenant en paramètre deux modèles, un modèle de processus d'entreprise et un modèle du système. Les métriques du facteur fonctionnel prennent en paramètre un troisième paramètre, l'activité que l'on étudie et pour laquelle on calcule ces métriques.
- Elles sont définies comme Nombre d'éléments de A sur Nombre d'éléments de B, de façon formelle $\text{card}(A)/\text{card}(B)$ où: B est un ensemble d'éléments présents dans le modèle de processus ; A est un sous-ensemble de B. A est l'ensemble des éléments de B qui sont supportés par le système, c'est-à-dire tel que, pour chaque élément de B, il existe un élément du modèle du système qui lui soit lié par un lien *correspond* ou *représente* (suivant la métrique). le mot clé *card* spécifie qu'on calcule le nombre d'éléments de l'ensemble étudié.

Les métriques permettent d'exploiter la nuance qui peut exister entre parfaitement aligné et pas du tout aligné. Afin d'aller plus loin dans cette idée et d'aider les décideurs à mieux apprécier quand des actions correctives s'imposent pour rétablir l'alignement, nous avons introduit deux notions, le *seuil* et le *poids*. Le *seuil* permet de fixer une limite inférieure en deçà de laquelle la rupture de l'alignement n'est plus acceptable. Le *poids* permet d'attribuer un coefficient particulier à chaque élément de l'ensemble étudié afin de modifier la métrique en fonction de leur importance réelle. Les *seuils* et les *poids* sont définis par les décideurs en fonction de leur connaissance et de leur appréciation du projet.

2.3.c Adaptation de la typologie générique de métriques par instantiation de métra modèles génériques

Afin d'aider les ingénieurs à définir les métriques spécifiques aux méta-modèle de processus et de système qu'ils emploient, nous avons défini des métriques génériques qui sont spécifiées entre deux méta-modèles génériques.

Les méta-modèles génériques servent de méta-modèles de référence de la même façon que dans [Rosemann2002]. La Figure 11 schématise notre approche pour définir des métriques spécifiques. La terminologie adoptée est celle de l'OMG.

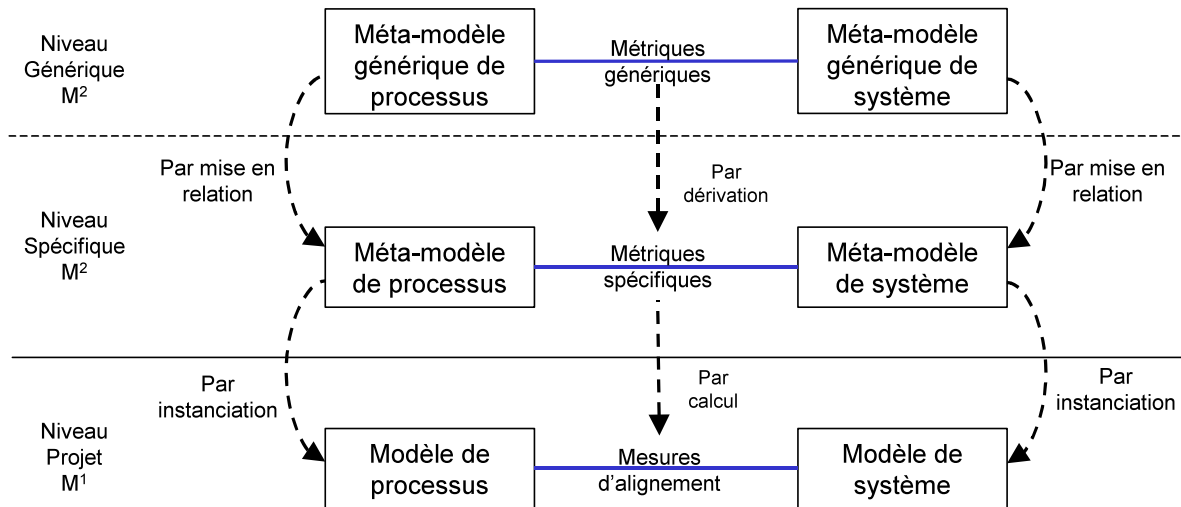


Figure 11: Un système de métriques d'alignement à trois niveaux différents

Les trois niveaux représentés en Figure 11 coïncident aux deux niveaux intermédiaires de l'architecture en couches [MOF].

- Le *niveau Projet* nommé niveau modèle dans [MOF] ou M¹ dans [Bezivin2001] correspond aux méta-données qui permettent de décrire les données du niveau instance (non représenté dans la figure). Ces méta-données sont agrégées sous forme de modèle. A ce niveau, on trouve en particulier un modèle de processus et un modèle de système entre lesquels nous cherchons à évaluer, à un instant donné, l'alignement.
- Le *niveau Spécifique* nommé niveau méta-modèle dans [MOF] ou M² dans [Bezivin2001] contient la description (c'est-à-dire les méta-méta-données) de la structure et de la sémantique des méta-données. Ce niveau correspond au méta-modèle de processus et au méta-modèle de système qui indiquent le type des éléments utilisés dans le modèle de processus et dans le modèle de système. Ces modèles sont des instances de leur méta-modèle respectif. Les méta-modèles *i** [Yu1997], diagramme d'activités UML [UML], WIDE [Casati1996], ou Entité-Relation [Chen1976] sont à ce niveau. L'évaluation de l'alignement entre les modèles de processus et de système du niveau projet est obtenue en appliquant les métriques spécifiques définies entre le méta-modèle de processus et le méta-modèle du système.
- Le *niveau Générique* se situe au même niveau que le niveau spécifique dans l'architecture en couches de l'OMG. Néanmoins, les méta-modèles définis à ce niveau servent de référence pour représenter un système ou un processus d'entreprise. Nous avons défini deux méta-modèles génériques:
 - le méta-modèle BPRAM (**B**usiness **P**rocess **R**epresentation for **A**lignment **M**easurement) qui permet de représenter de façon générique les concepts intervenant dans la modélisation des

processus d'entreprise. Ce méta-modèle est défini à partir de l'ontologie de Soffer et Wand (SW) [Soffer2004] ;

- le méta-modèle SRAM (**S**ystem **R**epresentation for **A**lignment **M**easure) qui permet de représenter de façon générique les concepts permettant de modéliser tout système. Ce méta-modèle est défini à partir de l'ontologie de Bunge, Wand et Weber (BWW) [Wand2004].

Ces méta-modèles génériques identifient les concepts génériques nécessaires à la définition de métriques génériques pour évaluer l'alignement. Ils permettent, par mise en relation, de rendre explicites les éléments et la structure des méta-modèles spécifiques nécessaires à l'évaluation de l'alignement. Ils ne sont pas universels (comme le MOF), c'est-à-dire qu'ils ne représentent pas un système ou des processus de façon exhaustive. Au contraire, ces méta-modèles proposent une description partielle pour un objectif particulier, celui d'évaluer l'alignement entre un modèle de processus et un modèle de système. Certains aspects sont pris en compte alors que d'autres sont omis. Ils sont le résultat de l'application d'un filtre [Breton2002].

Le mécanisme général du processus de génération de métriques spécifiques est celui de la mise en relation entre les concepts des méta-modèles génériques BPRAM et SRAM et ceux des méta-modèles spécifiques. Ce processus se compose de trois étapes:

- Associer les concepts du méta-modèle de processus choisi à ceux du méta-modèle BPRAM.
- Associer les concepts du méta-modèle de système choisi à ceux du méta-modèle SRAM.
- Adapter les métriques génériques en conséquence.

2.3.d Découverte des exigences d'évolution par la modélisation et la mesure de l'alignement au moyen d'un modèle pivot

La démarche méthodologique ACEM *guide la découverte d'exigences d'évolution du Système d'Information et des processus métiers de l'organisation par analyse de leur alignement*. ACEM se place dans le cadre de changement de Jarke [Jarke1994] qui définit le passage d'une organisation d'une situation présente à une situation future. La situation présente est caractérisée par les modèles As-Is BM (Business Model) et As-Is SFM (System Functionality Model) qui représentent les processus d'entreprise et les fonctionnalités du système. La situation future est caractérisée par les modèles To-Be BM et To-Be SFM représentant la situation du métier et du système respectivement, après évolution. Dans ACEM, l'évolution a lieu simultanément au niveau du processus-système défini au moyen de la relation d'alignement. A ce titre, ACEM peut être comme une démarche méthodologique qui guide la coévolution de processus et de systèmes, et à ce titre rentre dans notre cadre méthodologique de la coévolution.

Comme le montre la Figure 12, la spécificité d'ACEM réside dans l'utilisation d'un modèle pivot pour matérialiser la relation d'alignement. Ce modèle modifie la structure générale du cadre de changement définie ci-dessus. Il permet de représenter à la fois le système et les processus d'entreprise au sein d'un même modèle. L'utilisation d'un modèle pivot est motivée par la discordance conceptuelles dues aux langages utilisés pour

représenter les systèmes et les processus. En effet, les modèles de systèmes utilisent des concepts tels que “objets”, “états”, “opérations” ou “événements”. Les modèles de processus d’entreprise manipulent les concepts “d’acteurs”, de “rôles”, “d’activités”. Nous ajoutons donc au cadre de changement général un niveau intermédiaire qui permet de représenter conjointement, au sein d’un même modèle, le système et les processus. De cette façon, on différencie le niveau de l’organisation, le niveau du système et le niveau commun. Le niveau commun est formalisé au moyen du méta-modèle MAP. Le choix de ce formalisme est crucial pour deux raisons (a) il repose sur un méta-modèle intentionnel, offrant ainsi des mécanismes d’abstraction utiles à la représentation synthétique de concepts variés, et (b) il permet de modéliser aussi bien les éléments communs (alignés) que les éléments variables (partiellement ou pas alignés) du processus et du système. Les cartes produites au moyen du méta-modèles MAP intègrent donc l’ensemble des différentes intentions et stratégies sous-jacentes au processus et au système.

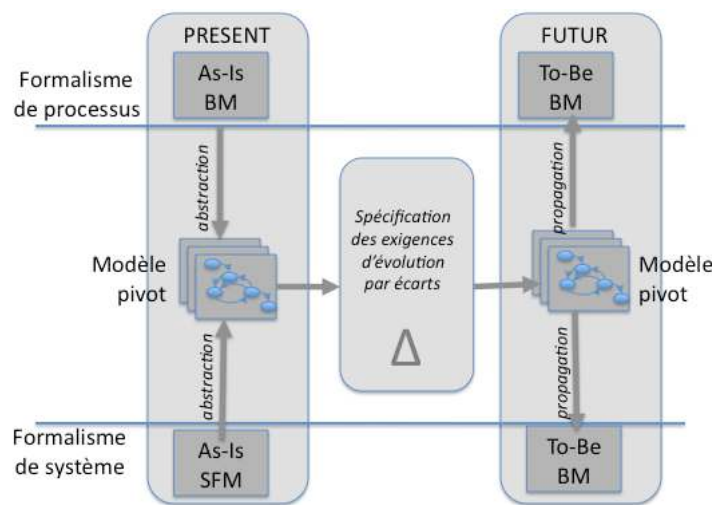


Figure 12: Aperçu de la démarche méthodologique ACEM

La démarche méthodologique ACEM repose sur un processus en trois étapes:

- *Obtention du modèle pivot.* Cette étape est réalisée à partir des modèles de processus et de système. ACEM propose pour la guider (1) d’abstraire les fonctionnalités du système et les activités métiers dans deux ensembles de cartes séparés et (2) de fusionner ces deux ensembles en précisant le degré d’alignement entre les différentes intentions et stratégies représentatives des modèles du processus et du système. Les degrés d’alignements sont évalués par la prise de mesures au moyen des métriques.
- *Evolution du modèle pivot.* Cette étape repose sur l’identification d’écarts coïncidant aux exigences d’évolution et exprimés à partir d’opérateurs d’écart définis selon l’approche GAP. L’identification des exigences d’évolution est guidée par l’analyse de l’alignement. Cette analyse repose sur la mise en perspective des mesures d’alignement relativement aux seuils. Une valeur de métrique inférieure au seuil signifie que des actions correctives doivent être entreprises sur les différents éléments impliqués dans la métrique. Il est possible d’exprimer ces actions correctives sous forme d’exigences d’évolution traduisant une volonté d’adaptation pour rétablir l’alignement. Pour rétablir l’alignement, on peut agir de deux

façons: (i) sur les processus en recherchant si les éléments non supportés par le système doivent être conservés dans les processus ou (ii) sur le système afin qu'il prenne en charge des éléments des processus qui ne l'étaient pas jusqu'alors. On peut entreprendre conjointement ces deux types de correction.

- *Propagation des écarts.* Conformément à notre cadre de coévolution, cette étape correspond à la propagation des écarts identifiés sur le modèle pivot dans les modèles du système et des processus. Cette partie du processus s'appuie sur la traçabilité des liens de coïncidence entre les éléments du modèle pivot et ceux du modèle de processus (ou du système).

L'approche ACEM applique donc le cadre de travail de la coévolution par interdépendance.

3.3. Résultats

Mes travaux sur le thème de l'alignement sont basés sur une collaboration avec de la société Renault DIAC et publié dans la revue Springer Requirements Engineering [Rolland2004]. J'ai par la suite dirigé la thèse de Anne Etien, soutenue en 2006, sur ce thème. Le mémoire de thèse présente en particulier une typologie de métriques d'alignement, ainsi que le processus de spécialisation permettant d'utiliser les métriques avec des méta modèles spécifiques. En outre, j'ai publié cinq articles dans des Conférences Internationales :

- CAISE'03 (Conference on Advanced Information Systems engineering) discute du lien d'alignement en général, de sa définition formelle, des différentes façons de l'appréhender, et des problématiques d'ingénierie qu'il soulève [Salinesi2003].
- Les articles publiés aux conférences EDOC'02 (International IEEE Conference on Enterprise Distributed Object Components) [Salinesi2002a] et OOIS'02 (Object Oriented Information Systems) présentent la typologie de liens d'alignement que je propose [Salinesi2002b].
- Les articles publiés dans les conférences internationales IRMA'04 (Information Resource Management Association) [Zoukar2004d] et AOIS'04 (Agent Oriented Information Systems) [Zoukar2004c], ainsi qu'à l'atelier AWRE'05 (Australasian Workshop on Requirements Engineering) [Salinesi2005] présentent notre approche d'emploi de la modélisation intentionnelle pour résoudre le problème de la discordance conceptuelle lors de l'analyse de l'alignement.

Ma contribution est triple. J'ai: (i) défini formellement l'alignement (ii) élaboré une typologie de métriques pour mesurer l'alignement, et (iii) développé une démarche méthodologique qui guide la découverte d'exigences d'évolution par une analyse de l'alignement. Le rôle de la modélisation intentionnelle dans la modélisation de l'alignement a été appréhendé lors de mes premiers travaux sur l'alignement, puis formellement défini lors de mes travaux sur l'alignement stratégique présentés dans la section suivante. L'ensemble des références afférentes y est présenté de manière plus complète.

La définition formelle de l'alignement que nous proposons repose sur la *typologie des liens d'alignement entre modèles de Système d'Information et de processus métier*. Cette définition de l'alignement s'appuie sur les

principes de la modélisation et de la méta-modélisation et s'inscrit dans la lignée des travaux de l'OMG sur UML et le MOF [MOF]. Cette définition présente deux caractéristiques principales:

- Elle est établie de façon générique, indépendamment de tout projet et de tout modèle spécifique, entre les éléments de deux méta-modèles génériques. Les méta-modèles que nous employons sont le méta-modèle des processus métier (BPRAM) et le méta-modèle du système (SRAM). Ces méta-modèles ont été définis au niveau générique à partir d'adaptations de l'ontologie de Bunge.
- Elle s'appuie sur trois éléments clé: les concepts nécessaires à la modélisation des processus métier (les éléments du méta-modèle de processus), ceux permettant de représenter le système (les éléments du méta-modèle du système) et des liens types formellement définis entre ces concepts.

Nous proposons une collection de *dix métriques qui permettent d'évaluer le degré d'alignement entre processus métier et Système d'Information* [Etien2006]. Cette collection de métriques a été élaborée de manière systématique en considérant les trois éléments clé de la définition de la relation d'alignement. Chaque métrique mesure plus particulièrement un aspect de la relation d'alignement. Une métrique se présente sous la forme d'un ratio qui prend les concepts des processus en référence (dénominateur) et les compare à ceux pris en charge par le système (numérateur). Ainsi, plus la valeur obtenue pour chaque métrique est élevée, plus le système et les processus sont alignés. Les métriques permettent donc d'évaluer le degré d'alignement dans un spectre de nuances entre parfaitement et pas du tout aligné.

Les métriques que nous proposons sont définies à un niveau générique entre les méta-modèles BPRAM et SRAM. Ces métriques génériques servent de support à un *processus de génération de métriques spécifiques associées à deux méta-modèles spécifiques permettant de respectivement représenter des processus d'entreprise et un système* [Etien2006]. La définition de ces métriques spécifiques est néanmoins conditionnée par l'existence dans les méta-modèles spécifiques de concepts spécialisant les concepts BPRAM et SRAM intervenant dans la définition des métriques génériques. Nous avons illustré le processus de génération et montré l'utilité de chacune des métriques pouvant être définies pour différents méta-modèles tels que le diagramme d'activités UML et le méta-modèle de classes O*.

Nous pensons qu'il existe plusieurs avantages à utiliser une approche en niveaux d'abstraction et à définir les méta-modèles au niveau générique à partir d'adaptations de l'ontologie de Bunge:

- les métriques génériques reposent sur une base théorique fournie par l'ontologie de Bunge ;
- les métriques génériques servent de guide pour définir les métriques spécifiques: ces dernières étant une spécialisation des premières ;
- le processus pour construire les métriques spécifiques est simple et facile à mettre en œuvre. Il génère moins d'erreurs que des constructions ad hoc et, enfin,
- les ensembles spécifiques de métriques d'alignement sont cohérents entre eux car ils sont générés à partir du même moule.

Nous avons proposé une démarche méthodologique d'ingénierie des exigences, nommée ACEM, et l'avons définie formellement au moyen d'un modèle de processus spécifié avec le formalisme MAP. *ACEM guide la découverte d'exigences d'évolution d'un Système d'Information et de processus métier par la modélisation et l'analyse de l'alignement* [Sainesi2002a] [Salinesi2002b] [Rolland2004] [Salinesi2005]. Le processus méthodologique est constitué de trois étapes visant à: (i) construire un modèle pivot qui matérialise la relation d'alignement, (ii), le faire évoluer afin de définir les besoins de changement, et (iii) propager les évolutions définies afin de spécifier les modèles de la situation future.

ACEM est une véritable démarche de coévolution. L'approche employée est plus précisément une *approche de coévolution par interdépendance, centrée sur un modèle pivot qui représente conjointement des éléments du processus et du système* [Salinesi2003] [Zoukar2004c] [Zoukar2004d] [Etien2005]. Nous avons expérimenté empiriquement la démarche ACEM avec le cas d'étude du Financement Unitaire à la société DIAC [Rolland2004] Dans ACEM, les exigences d'évolution sont spécifiées par des opérateurs d'écart formalisés au moyen de l'approche GAP. Le méta-modèle utilisé pour spécifier le modèle pivot adopte une approche intentionnelle (adaptée du formalisme MAP) qui permet d'abstraire les détails techniques pour se concentrer sur l'essentiel. La représentation des éléments du système et des processus métiers apparaît conjointement dans le même modèle, cette représentation est effectuée dans les mêmes termes pour tous les types d'éléments alignés. La relation d'alignement apparaît dans le modèle pivot grâce à un système de coloration. Grâce à ce modèle, les experts métiers et les ingénieurs peuvent communiquer et se comprendre en adoptant le même langage. En outre, l'emploi du modèle pivot permet à notre approche: de faire évoluer des processus et le système associé même quand les modèles de la situation présente sont inexploitable (en définissant les besoins d'alignement relatifs à cette situation) et de construire des processus d'entreprise et un système alignés.

4. Modélisation de l'alignement stratégique – l'approche INSTAL

Aujourd'hui, il ne suffit plus de construire des Systèmes d'Information performants ou de définir des processus métier coïncidant aux « bonnes pratiques » et de développer les Systèmes d'Information qui les supportent. Pour que l'entreprise soit performante, ses systèmes et ses processus doivent en permanence être alignés avec sa stratégie, son modèle économique. Les directeurs de Systèmes d'Information font donc face à un nouvel enjeu: « être parties prenantes du processus de changement et de transformation de l'entreprise, c'est à dire avoir les capacités nécessaires pour parler du nouveau modèle économique de l'entreprise »[CIGREF2007]. En fait, l'alignement stratégique est devenu en dix ans leur priorité absolue [Luftman07].

Or cette exigence nouvelle semble particulièrement difficile à satisfaire dans la pratique. Une illustration frappante du problème est la difficulté à maîtriser le rôle de l'informatique dans l'innovation des entreprises. Ainsi, les résultats d'une enquête publiée très récemment par Cap Gemini [CapGemini2008] montrent que bien que 2/3 des DSI (4/5 dans le secteur public) estiment que l'informatique joue un rôle crucial en matière d'innovation des produits et services de l'entreprise, seul 1/4 se considère en mesure de jouer un véritable rôle sur ce sujet.

Les études empiriques de l'alignement stratégique [Chan2007], font clairement apparaître les bénéfices potentiels que celui-ci pourrait procurer. Tout d'abord, ces recherches montrent que l'alignement stratégique accroît la performance de l'organisation [Khefi2006]. En outre, une enquête que nous avons menée nous a montré la diversité des attentes des professionnels [Salinesi2008b]. Pour 80 % d'entre eux, l'alignement est susceptible d'apporter une meilleure évolutivité du Système d'Information et du métier. 70% estiment qu'il permettrait un meilleur contrôle des coûts. 70 % encore pensent que l'alignement stratégique est susceptible d'apporter une meilleure visibilité sur la valeur ajoutée du Système d'Information. Inversement, lorsque l'alignement est mauvais, les processus métier ne profitent pas pleinement des moyens technologiques investis [Henderson1993].

Selon Luftman [Luftman2007], un problème crucial lorsqu'on traite de l'alignement stratégique est le manque de compréhension et de communication entre le monde de la stratégie, du management, et le monde du Système d'Information et des processus métier. En effet, les acteurs qui définissent la stratégie de l'entreprise (i) ne parlent pas le même langage que les acteurs du niveau opérationnel (tels que chefs de projets, responsables d'applications informatiques) et, (ii) ils n'ont pas la même vision de l'entreprise. [Fimbel2006] met en évidence le besoin d'une vision commune du SI partagée par les dirigeants et les acteurs opérationnels comme une condition de la réussite des manœuvres d'alignement, ainsi que la nécessité d'une meilleure compréhension des contraintes de faisabilité par les parties prenantes hors directions des SI.

Il existe bien des modèles d'alignement stratégique. Le plus connu est sans doute le Strategic Alignment Model [Henderson1993] dont l'intérêt est de fournir une définition très générale de l'alignement stratégique. Ces modèles relèvent du domaine du management plus que de l'ingénierie des Systèmes d'Information. S'inscrivant dans une approche causale, ils permettent l'étude de phénomènes liés à l'alignement, comme par exemple son influence sur la performance des entreprises [Kefi2006], et l'élaboration de modèles prédictifs [Moody2005]. Cependant, pour une organisation donnée, ce type de modèle ne fournit qu'une vision très macroscopique de l'alignement stratégique. En outre ces modèles d'alignement stratégique ne présentent pas la dimension conceptuelle qui est indispensable en ingénierie des Systèmes d'Information. Il est donc difficile d'adopter une approche systématique lorsqu'on les analyse. L'absence de modèle conceptuel de l'alignement stratégique rend donc difficile sa prise en compte dans des méthodes d'ingénierie.

Ces diverses considérations m'ont amené à soulever la question de recherche suivante:

« Un langage de modélisation conceptuelle de l'alignement stratégique supporterait-il une analyse systémique autour du rôle joué par les Systèmes d'Information et les processus dans le modèle économique de l'entreprise? »

4.1. Problématiques liées à la modélisation de l'alignement stratégique

L'alignement stratégique est une forme d'alignement très particulière. La différence entre l'alignement tel que nous l'avons abordé dans la section précédente et l'alignement stratégique est double: d'une part à cause la nature particulière de la stratégie, et d'autre part en raison de la nature propre de l'alignement stratégique. Les problématiques évoquées pour l'alignement restent valables pour l'alignement stratégique. Cependant, la singularité de l'alignement stratégique fait apparaître un certain nombre de problématiques supplémentaires: (i)

discordance conceptuelle, (ii) décalage dans les niveaux d'abstraction, (iii) absence de définition formelle conceptuelle de la stratégie, (iv) antagonisme temporel de l'alignement stratégique, (v) explosion combinatoire, (vi) transversalité, et (vii) manque d'indicateurs spécifiques. Plusieurs de ces problématiques spécifiques remettent en cause l'emploi de l'approche ACEM (ou toute autre approche qui lui serait semblable) pour modéliser l'alignement stratégique.

i. Discordance conceptuelle: Une organisation établit une stratégie pour définir le modèle économique qui assure sa pérennité. En entreprise, le concept de stratégie est intimement liée aux concepts de mission, de valeur, d'objectif stratégique, ou de performance. D'un point de vue conceptuel, ces notions sont très distantes de celles qui servent à concevoir les Systèmes d'Information ou à modéliser les processus d'entreprise (par exemple: objet, agent, action, séquence, événement, etc). Les deux niveaux, le niveau stratégique et le niveau opérationnel, sont donc conceptuellement discordants, et ce de manière inhérente. En raison du problème de discordance conceptuelle, on ne peut espérer définir de lien de coïncidence entre les concepts des deux niveaux. La définition de l'alignement telle que proposée par l'approche ACEM (par les liens "correspond" et les liens "représente") n'est donc pas adaptée à l'alignement stratégique. Le premier enjeu est par conséquent de trouver un langage de modélisation conceptuel qui permette de subsumer les deux niveaux opérationnel et stratégique.

ii. Décalage dans les niveaux d'abstraction: Une autre forme de discordance apparaît au travers des différents niveaux de détails auxquels sont définis la stratégie, les systèmes et les processus. La stratégie, tout comme les modèles du niveau opérationnel peut être définie à différents niveaux d'abstraction et de détail allant du macroscopique au grain fin. Or les niveaux d'abstraction de descriptions de la stratégie ne sont pas comparables à ceux des modèles de système ou de processus. L'emploi d'une somme de liens simples à la manière de la méthode ACEM pose la question de quel niveau d'abstraction des modèles opérationnels lier avec quels niveau d'abstraction de la stratégie. Le langage de modélisation conceptuel de l'alignement stratégique doit donc offrir un mécanisme d'abstraction permettant de lier les éléments stratégiques et opérationnels à des niveaux de détail conjoints.

iii. Absence de définition formelle conceptuelle de la stratégie: Bien que la littérature du management abonde de techniques d'aide à la définition de la stratégie (par exemple la matrice BCG, la méthode MACTOR, l'analyse SWOT, les 7S Mc Kinsey, les chaînes de valeur interne, ou ClarkeDL) la notion même de stratégie reste souvent très abstraite. En fait, nous n'avons pas trouvé de définition de la stratégie qui (a) soit formelle, (b) soit universellement reconnue, et (c) supporte une modélisation conceptuelle. Pire, le principe d'une modélisation conceptuelle de la stratégie n'est pas forcément souhaitée par les responsables ou par les cabinets de conseil en stratégie qui préfèrent utiliser les techniques citées ci-dessus et continuer de définir la stratégie en langage naturel.

Même si elle permet de rapprocher les niveaux stratégiques et opérationnels, une définition abstraite de la stratégie s'avère peu utile à l'élaboration d'un modèle conceptuel de l'alignement stratégique selon l'approche ACEM. En effet, une hypothèse fondamentale d'ACEM est que pour définir un modèle d'alignement entre deux concepts donnés, il faut savoir instancier les méta-modèles génériques au niveau duquel les liens d'alignement génériques ont été définis. Ne souhaitant pas définir de méta-modèle de la stratégie, nous ne pouvons utiliser l'approche ACEM pour définir un méta-modèle d'alignement stratégique.

iv. Antagonisme temporel de l'alignement stratégique: d'un côté, la stratégie est de nature progressive, elle définit une évolution souhaitée de l'organisation. De l'autre côté les modèles du niveau opérationnel décrivent des situations qui coïncident à des instants dans le temps. Il y a donc un antagonisme entre sur la nature temporelle de l'alignement stratégique. Un modèle conceptuel d'alignement stratégique devrait permettre de représenter à la fois la progression souhaitée et les différents états de la progression (situations As-Is et To-Be décrits par les modèles du niveau opérationnel).

v. Explosion combinatoire: La nature complexe de l'alignement stratégique est aussi liée au fait que ni la stratégie ni les Systèmes d'Information ne sont des éléments simples (ou qui pourraient être considérés comme une somme d'éléments simples). Représenter l'alignement stratégique par une somme de liens simples générerait une explosion combinatoire du nombre de liens. Cette explosion combinatoire apparaît par exemple dans des techniques d'alignement stratégiques de type matrice d'alignement [Longépé2002] qui, en fait, ne permet que de représenter des liens simples (identifiés dans les cellules de la matrice).

vi. Transversalité: La stratégie a un caractère transverse qui lui est inhérent. En effet, il est quasi systématique qu'un élément de stratégie impacte plusieurs (parfois de nombreux) éléments au niveau opérationnel. L'alignement stratégique est forcément un concept plus complexe que le couplage d'éléments de niveaux différents tels que le définit l'approche ACEM. Un modèle conceptuel de l'alignement stratégique devrait donc être capable de représenter l'alignement stratégique avec un degré de complexité que des ensembles de liens simples ne permettent pas d'atteindre.

vii. Manque d'indicateurs spécifiques: les projets d'ingénierie de Systèmes d'Information peuvent être bâtis en cherchant à "optimiser" l'alignement stratégique au sens du "mieux", "plus vite", "plus efficace", mais si les indicateurs avant projet ne sont pas définis, ceux qu'on mesure après projet n'ont pas de sens. Les indicateurs d'alignement stratégique doivent être porteurs d'information quantitative sur les bénéfices effectifs (constatés ou estimés) de l'alignement. Or les indicateurs d'alignement tels que définis dans ACEM n'apportent une information que sur la qualité de l'alignement. L'enjeu est donc de définir des indicateurs spécifiques qui permettent d'évaluer de manière fiable les bénéfices potentiels ou effectifs de l'alignement stratégique par leur mesure.

Les bénéfices potentiels d'un langage de modélisation conceptuelle sont nombreux. Le but est bien de développer une démarche méthodologique d'ingénierie des exigences qui guide la découverte d'exigences de coévolution de Systèmes d'Information et de processus métier par analyse de l'alignement stratégique. L'enquête que j'ai menée auprès d'un panel d'industriels [Thevenet2008a] a montré que les praticiens estiment que disposer de modèles de l'alignement stratégiques pourraient leur permettre de gérer les portefeuilles de projets, d'aider dans les prises de décisions, ou même tracer les évolutions du Systèmes d'Information et des processus métier.

4.2. Principes de l'approche INSTAL

L'approche INSTAL repose sur l'idée qu'il y a alignement lorsque les éléments alignés ont une intention commune. L'approche intentionnelle paraît particulièrement adaptée parce que: (a) elle offre les mécanismes

d'abstraction utiles lorsque l'on représente des phénomènes complexes, (b) son efficacité a déjà été démontrée aussi bien pour spécifier la stratégie [Longépé2002], [Bleistein2006], que des processus métier [Loucopoulos1999], [Kueng1999], ou des systèmes [Lamsweerde2004], [Rolland2005], et surtout (c) les intentions permettent de formuler ce que l'on souhaite, ce qui est une caractéristique distinctive de la stratégie.

2.4.a Spécification de l'exigence d'alignement stratégique au moyen d'un formalisme intentionnel

Le langage d'alignement stratégique proposé dans l'approche INSTAL repose donc sur un méta-modèle intentionnel. La stratégie y est décrite sous la forme de buts stratégiques dont la description est enrichie par une information sur les choix qui ont été faits et qui sont relatifs aux différentes manières d'atteindre ces buts. Le langage de modélisation employé pour spécifier les modèles d'alignement stratégique est formalisé par le méta-modèle MAP. Les trois concepts centraux du formalisme MAP sont: le but, la tactique (dénommée ainsi au lieu de stratégie afin d'éviter toute confusion avec la stratégie d'entreprise), et la section. Chaque tactique est associée à un but cible; une tactique décrit une manière d'atteindre le but cible. Les sections sont des triplets <but source – tactique – but cible> tels que le but source est une pré condition à l'emploi de la tactique pour chercher à satisfaire le but cible. Les modèles produits avec le formalisme MAP sont appelés des cartes.

Le formalisme MAP a été choisi pour plusieurs raisons:

- C'est un formalisme intentionnel. Comme nous l'avons vu ci-dessus, les formalismes intentionnels sont des candidats de première classe pour formaliser l'alignement stratégiques. De ce point de vue, nous nous positionnons dans le courant de pensée de [Bleistein2006] et [Singh2007]. Les formalismes intentionnels subsument les concepts du niveau opérationnel et du niveau stratégique, ce qui permet de résoudre le problème de la discordance conceptuelle.
- La représentation des tactiques comme différentes manières de satisfaire un but permet de surmonter le problème de l'antagonisme temporel. En effet, les intentions formalisent ce qui est souhaité par la stratégie, et les manières de satisfaire les intentions font le lien avec les niveau opérationnel en indiquant comment ce qui est souhaité serait réalisé. D'autre part, il permet de faire le lien entre les buts stratégiques et les nombreux éléments du niveau opérationnel.
- Les buts et les tactiques sont définis de manière abstraite. Chaque but et chaque tactique peut donc coïncider à plusieurs éléments des niveaux stratégiques et opérationnel.
- Le formalisme MAP offre un mécanisme d'affinement. Le fait que l'on puisse spécifier avec le formalisme MAP les diverses manières de satisfaire une intention est utile à la prise en compte de la nature complexe de la stratégie. MAP offre aussi un mécanisme d'affinement. La nature complexe de l'alignement stratégique est donc prise en compte en partant d'intentions abstraites et en rentrant progressivement dans les détails.

Dans l'approche INSTAL, les modèles, nommés cartes, spécifient l'alignement stratégique. On parle donc de *cartes d'alignement stratégique*. Dans chaque section d'une carte d'alignement stratégique, le but cible et la tactique sont partagés par le niveau stratégique *et* par le niveau opérationnel. Les cartes d'alignement stratégiques forment une sorte de pivot au sens où elles se placent de manière intermédiaire entre les niveaux opérationnel et stratégique. Contrairement aux cartes pivot de l'approche ACEM, les cartes d'alignement stratégique ne représentent pas l'union des éléments définis aux deux niveaux alignés. Au contraire, elles spécifient ce qui est en lien entre les documents de la stratégie, les modèles de Système d'Information et les modèles de processus.

2.4.b Définition de liens structurellement complexes à partir du modèle intentionnel

Dans l'approche INSTAL, les liens d'alignement stratégique sont des liens complexes. Les liens d'alignement stratégique sont définis avec un cœur et avec des extrémités.

- Le *cœur* de chaque lien d'alignement stratégique est une section dans une carte d'alignement stratégique. Cette section spécifie l'intention d'alignement stratégique et le moyen prévu pour la réaliser.
- Les *extrémités* des liens d'alignement stratégique identifient les éléments alignés au niveau stratégique et au niveau opérationnel et définissent le rôle joué par chacun de ces éléments dans l'alignement stratégique.

L'approche INSTAL propose une typologie de liens en fonction de la forme de leurs extrémités. Comme le montre la Figure 13, quatre formes de liens d'alignement stratégique sont proposés dans la typologie INSTAL: un lien de forme simple et trois liens de forme complexe nommés râteau, pinceau et toile d'araignée.

- Un *lien simple* lie un élément source et un élément cible. Cette forme de lien est couramment utilisée dans les approches qui préconisent des matrices d'alignement [Longépé2002]. Les liens de coïncidence définis dans l'approche ACEM sont aussi des liens simples.
- Les trois formes de *liens complexes* font intervenir plusieurs éléments sources et/ou éléments cibles: (i) le lien dît *râteau* lie un élément source et un ensemble d'élément cible (i.e. agrégat d'éléments cibles), (ii) le lien dît *pinceau* lie un ensemble d'éléments sources et un élément cible puis (ii) le lien dît *toile d'araignée* lie un ensemble d'éléments sources et un ensemble d'éléments cibles.

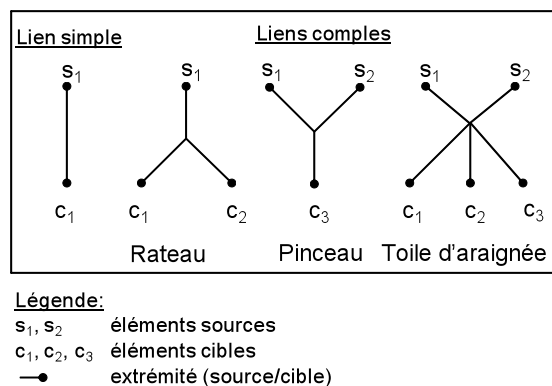


Figure 13: Typologie des structures de liens d'alignement stratégique dans InStAl

La structure des liens d'alignement stratégique est formalisée au moyen d'un méta-modèle. D'après le méta-modèle INSTAL, un lien d'alignement stratégique a pour source et pour cible des extrémités qui peuvent être des extrémités simples ou des extrémités agrégats. Chaque extrémité simple peut être associée à un et un seul élément appartenant soit au niveau opérationnel soit au niveau stratégique. Une extrémité agrégat est associée à plusieurs éléments. Les différents éléments d'une extrémité agrégat de lien d'alignement stratégique doivent appartenir au même niveau, soit opérationnel, soit stratégique. Une extrémité agrégat peut donc être considérée comme une composition d'extrémités simples dirigées vers le même niveau.

L'emploi de liens complexes permet de résoudre le problème de la transversalité. Un lien d'alignement stratégique complexe peut par exemple associer un élément de la stratégie à plusieurs éléments du niveau opérationnel.

Les liens complexes permettent aussi de faire face au problème de l'explosion combinatoire rencontré lorsque l'on emploie des liens d'alignement stratégique simples. Au lieu de définir $(m \times n)$ liens simples entre n éléments de la stratégie et m éléments du niveau opérationnel, l'approche INSTAL permet de définir un lien unique entre l'agrégat des m éléments stratégiques et l'agrégat des n éléments opérationnels. De plus, la section qui apparaît au cœur du lien complexe, avec son but cible et sa tactique, permet de définir le propos commun des différents éléments du niveau opérationnel et du niveau stratégique mis en relation d'alignement.

2.4.c Définition du rôle joué par les entités alignées

Le méta-modèle InStAl indique que la connexion de chaque élément à un lien d'alignement stratégique par une extrémité est caractérisée par un rôle. Cela permet de définir de manière individuelle le rôle spécifique de chaque élément dans un alignement stratégique. Avec les liens complexes, il est possible de voir qu'un élément intervient dans différents liens d'alignement et de bien distinguer le rôle de l'élément au sein d'un lien d'alignement donné, ce qui n'est pas possible avec les liens simples. J'ai mené avec mon équipe une étude empirique des rôles que jouent les éléments dans l'alignement stratégique. Cette étude nous a permis de constituer une typologie faisant apparaître différents types de rôles: nécessaire, utile, suffisant, contraint, et contradictoire.

Ces cinq types de rôles peuvent être définis de la manière suivante:

- Une extrémité est nécessaire dans le lien d'alignement stratégique quand l'alignement ne peut être parfait si l'extrémité n'est pas réalisée.
- Le lien d'utilité indique une dépendance de nature plus faible que le lien nécessaire. Une extrémité est utile au lien d'alignement quand celle-ci aide à réaliser/satisfaire le lien d'alignement sans être pour autant obligatoire.
- Une extrémité est suffisante au lien d'alignement quand la réalisation de l'extrémité suffit à réaliser l'alignement. Ce rôle peut être combiné avec le rôle nécessaire pour former le rôle nécessaire et suffisant.
- Une extrémité peut être contrainte par d'autres extrémités à l'opposé du lien d'alignement (e.g. une extrémité source est contrainte par une extrémité cible, et vice-versa) quand elle a plus de capacités que

ce pour quoi elle est référencé dans le lien d'alignement, et que l'utilisation partielle de cette capacité est due à l'extrémité opposée.

- Une extrémité est contradictoire avec les extrémités opposées quand leurs présences ensemble créent un non (ou un mauvais) alignement. Ce lien met en évidence les cas où l'alignement est mal assuré dans les faits.

2.4.d Mesure du degré effectif d'alignement au moyen de métriques

L'élaboration de tableaux de bord est une préoccupation importante des décideurs d'entreprises qui veulent avoir une vision quantifiée de l'alignement stratégique. Comme l'approche ACEM, l'approche INSTAL permet la quantification de l'alignement stratégique au moyen de *métriques* et de *mesures*.

Les *métriques* apportent une vision synthétique de la manière d'évaluer la satisfaction du but qui est au cœur du lien d'alignement stratégique. Par exemple une métrique du type "taux d'incidents de sécurité" peut être associée à un lien d'alignement stratégique identifiée par la section "maintenir la réputation de l'entreprise et son leadership en s'assurant que les services informatiques sont capables de résister à des attaques et d'en surmonter les effets". Dans la pratique courante, les métriques sont synthétiques. Par exemple, la métrique du taux d'incidents de sécurité permet de synthétiser l'ensemble des données relatives à la sécurité en un seul chiffre.

Les *mesures* définissent la manière de calculer la valeur effective de la capacité à atteindre les buts qui sont au cœur des liens d'alignement stratégique, selon les principes du tableau de bord équilibré de Robert Kaplan et David Norton [Kaplan2006]. Par exemple, la métrique du taux d'incidents frauduleux peut être mesurée par le nombre d'accès frauduleux aux services de l'entreprise, par le nombre d'incidents informatiques réels qui ont eu un impact sur l'activité de l'entreprise, ou par le nombre d'incidents qui ont mis l'entreprise en difficulté vis à vis des tiers.

A une métrique peuvent donc être associées différentes mesures qui (a) sont calculables par l'analyse des éléments du niveau opérationnel, et (b) apportent des facettes différentes de la métrique. Le méta-modèle d'alignement stratégique INSTAL fait apparaître les métriques et les mesures de manière distincte: les métriques relèvent du niveau stratégique, alors que les mesures sont rattachées au niveau opérationnel. Une spécificité importante des mesures proposées par INSTAL est que leur calcul est effectué à partir des instances du niveau opérationnel. Cette approche, différente de celle proposée dans ACEM, permet de mesurer la satisfaction effective de l'alignement stratégique, et non pas projetée à partir de modèles. Par ailleurs, les métriques et mesures sont attachées aux liens et non aux extrémités du lien, ce qui permet de les définir même quand le lien d'alignement stratégique n'est que partiellement spécifié (c'est à dire que toutes ses extrémités n'ont pas encore été identifiées).

Par ailleurs, le méta-modèle fait apparaître un concept de *seuil*. Les seuils sont des valeurs cibles qui sont associées aux métriques. L'alignement stratégique est quantifié par des valeurs cibles qui doivent être atteintes à différentes dates. Chaque seuil est donc associé à une date qui définit l'horizon auquel l'alignement stratégique doit avoir quantitativement atteint le but cible.

Enfin, les métriques ne sont pas définies de manière absolue. Chacun de points de vue coïncidant aux entités alignées peut définir à la fois:

- les métriques et seuils qui selon ce point de vue doivent être employés pour mesurer sa propre contribution à la satisfaction de l'exigence d'alignement stratégique, et
- les métriques et seuils qui, selon ce point de vue, doivent être employés pour mesurer la contribution de l'autre point de vue à la satisfaction de l'exigence d'alignement stratégique.

Pour que l'exigence d'alignement stratégique soit satisfaite, il faut donc (a) que les contributions nécessaires et suffisantes d'éléments appartenant aux différents points de vue alignés soient identifiées, (b) qu'il n'existe aucun lien de contribution qui fasse apparaître une contradiction avec l'exigence d'alignement stratégique, (c) que les métriques et seuils des différents points de vue soient cohérents, et enfin (d) que les mesures effectives des métriques respecte les valeurs des seuils associés.

2.4.e Découverte d'exigences d'évolution par modélisation et mesure des exigences d'alignement stratégique

La démarche méthodologique INSTAL *guide la découverte d'exigences d'évolution du Système d'Information et des processus métiers de l'organisation par analyse de l'alignement stratégique*. INSTAL se place dans le cadre de changement de Jarke [Jarke1994] qui définit le passage d'une organisation d'une situation présente à une situation future. La situation présente est caractérisée par les modèles As-Is des éléments stratégiques des éléments opérationnels qui représentent d'une part la stratégie de l'entreprise, et d'autre part les processus d'entreprise et les fonctionnalités du Système d'Information. La situation future est caractérisée par les modèles To-Be représentant la situation de la stratégie, du métier et du système, après évolution.

Une hypothèse essentielle de la méthode est que nous ne maîtrisons pas l'évolution de la stratégie. Celle-ci est définie par des décideurs auxquels nous ne prétendons pas indiquer comment faire évoluer le modèle économique de leur entreprise. En revanche, l'évolution guidée par INSTAL a lieu au niveau opérationnel, et concerne simultanément les processus métier et le Système d'Information. A ce titre, INSTAL peut être vue comme une démarche méthodologique qui guide la coévolution de processus et de systèmes, et rentre dans notre cadre méthodologique de la coévolution.

Comme le montre la Figure 14, le point de départ de la méthode INSTAL réside en une collection d'exigences d'alignement stratégique. Les exigences d'alignement stratégique sont spécifiées au moyen de cartes d'alignement stratégique. Chaque section de ces cartes définit un lien d'alignement stratégique et est documentée par des mesures, métriques et seuils qui indiquent les bénéfices attendus de l'alignement stratégique. Les cartes d'alignement stratégique permettent (a) de mettre en coïncidence les éléments des niveaux stratégiques et du niveau opérationnel, (b) de modéliser l'alignement stratégique et d'établir un diagnostic sur la situation courante (As-Is), (c) de guider la découverte d'exigences d'évolution, et (d) de donner un aperçu de l'alignement stratégique futur (To-Be).

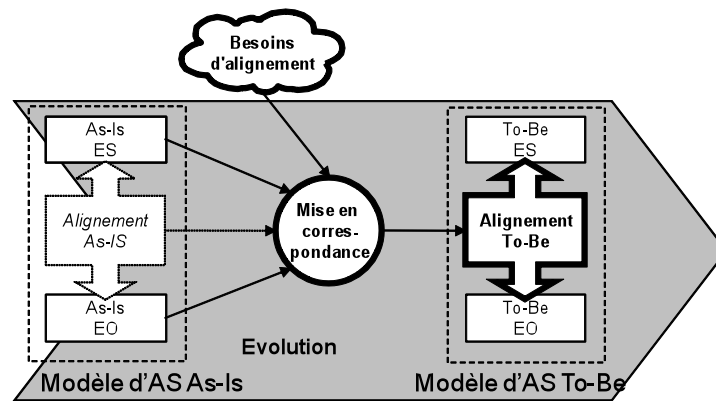


Figure 14: Aperçu de la démarche méthodologique InStAl

La démarche méthodologique INSTAL est constituée de trois étapes: (i) diagnostique de l'alignement stratégique, (ii) découverte des exigences d'évolution (iii) propagation et validation des exigences d'évolution.

- *Diagnostic de l'alignement stratégique:* INSTAL propose de diagnostiquer la situation par rapport à cet alignement. Tout d'abord, INSTAL établit une mise en relation qui vise à produire un modèle d'alignement stratégique conforme aux exigences définies par les cartes d'alignement stratégique. Chaque lien d'alignement stratégique est complété en identifiant les éléments concernés aux niveaux stratégique et opérationnel.
- *Découverte des exigences d'évolution:* L'obtention d'un alignement To-Be conforme aux besoins d'alignement peut requérir un certain nombre de changements au niveau opérationnel. Des exigences d'évolution peuvent donc apparaître au moment de l'élaboration des modèles de liens d'alignement stratégique. Une fois l'ensemble des extrémités définies, la qualité de l'alignement stratégique courant peut être calculée au moyen des métriques, mesures et seuils. Ceux-ci jouent le rôle de critère de satisfaction qui formalise la satisfaction des exigences d'alignement stratégique, et donc ils indiquent si de nouvelles évolutions doivent être envisagées.
- *Propagation et validation des exigences d'évolution:* Les exigences d'évolution sont spécifiées au moyen de l'approche GAP. Différentes exigences d'évolutions peuvent être produites à l'issue de l'analyse des différents liens d'alignement stratégique. Ces exigences d'évolution pouvant concerner les mêmes éléments, elles peuvent se combiner ou être contradictoire. INSTAL propose donc de prioriser les exigences d'évolution, et de les intégrer. Les métriques, mesures et seuils peuvent être utilisé pour valider les exigences d'évolution. Par exemple, la nécessité d'exigences d'évolution qui impactent des éléments concernés par des liens d'alignement stratégique jugés satisfaisants du point de vue des métriques pourrait être remise en cause.

INSTAL est une démarche progressive destinée à être employée de manière continue et non pas de manière discrète pour un projet donné. L'étape de diagnostique doit donc en permanence être refaite, afin d'évaluer si les bénéfices attendus de l'alignement stratégique sont bien constatés, et dans le cas où des dérives seraient constatées, proposer des mesures correctives de l'alignement stratégique.

4.3. Résultats

Nos travaux sur l'alignement stratégique s'inscrivent dans un courant qui cherche à intégrer l'alignement stratégique aux méthodes d'ingénierie de Systèmes d'Information.

Y. Pigneur [Osterwalder2002] propose une ontologie de modèles e-business dans laquelle des exigences de haut niveau des Systèmes d'Information sont explorées du point de vue du support que ces systèmes apportent à la stratégie e-business de l'entreprise.

La méthode e3-value [Pijpers2007] guide la modélisation d'un système d'entreprises sous l'angle d'un réseau d'acteur échangeant des valeurs. L'alignement stratégique apparaît implicitement au travers des flux de valeurs qui déterminent les bénéfices que tire chaque acteur de la collaboration.

La méthode i* [Yu1997] propose une technique de modélisation dans laquelle un réseau d'acteurs (par exemple des entreprises, une entreprise et son système) peut être modélisée au moyen de diagrammes de relations stratégiques. Ce type de diagramme montre comment les acteurs dépendent les uns des autres pour la satisfaction de leurs buts individuels.

Dans la méthode SEAM [Wegman2007] des diagrammes d'activité sont utilisés pour représenter la mise en œuvre commune d'une stratégie de constellation d'entreprises. Un modèle des relations fournisseur/adopteur documente l'alignement au travers de dépendances entre les entreprises qui proposent des valeurs au travers de leurs Systèmes d'Information et celles qui en bénéficient.

Comme ces différentes approches, nous cherchons à utiliser la modélisation conceptuelle comme support à l'ingénierie des Systèmes d'Information par analyse de l'alignement stratégique. L'originalité de notre approche réside dans la modélisation explicite de l'alignement stratégique.

Ma contribution sur le thème de l'alignement stratégique est triple : (i) proposition d'un méta modèle pour une modélisation conceptuelle de l'alignement stratégique, (ii) métriques pour guider la mesure de l'alignement stratégique, et (iii) proposition d'une démarche méthodologique pour guider la découverte d'exigences d'évolution du système d'information par analyse de l'alignement stratégique.

Les résultats de cette recherche ont fait l'objet de plusieurs publications dans :

- les revues telles ISI (Ingénierie des Systèmes d'Information) [Salinesi2008a] [Thevenet2008b] [Thevenet2009a] [Gam2007], INSIGHT [Thevenet2008c], et SRF (Systems Research Forum Journal) [Thevenet2009c],
- les conférences internationales telles que CAiSE (International Conference on Advanced Information Systems Engineering) [Thevenet2007b] [Viscusi2008a], et RCIS (Research Challenges in Information Systems) [Thevenet2007a] [Salinesi2007a]
- divers ateliers (AWRE, RIGIM, ECI) [Thevenet2006a] [Thevenet2007c] [Thevenet2008b].

Ces 13 publications (dont 6 dans des revues et 4 dans des conférences internationales) sont le fruit de deux collaborations :

L'une avec la société BNP Paribas. Au cours de cette collaboration j'ai pu analyser la problématique de l'alignement stratégique sur le terrain, identifier les besoins en termes de méthode, et réaliser une étude de cas en appliquant la méthode INSTAL sur un cas réel chez BNP Paribas. Cette collaboration a enfin été le cadre de la thèse CIFRE de Laure Hélène Jean Baptiste - Thevenet que j'ai encadré avec Colette Rolland [Thevenet 2009].

La seconde avec L'Université Polytechnique de Milan. Cette seconde collaboration a été l'occasion d'élaborer la typologie des métriques d'analyse stratégique, et de les tester sur une étude de cas [Viscusi2008a].

Outre l'étude de cas menée avec BNP Paribas [Thevenet2007b] [Thevenet2008b] [Thevenet2009], et celle des e-Services développés par le gouvernement italien [Viscusi2008a] j'ai évalué la méthode INSTAL sur le cas SEVEN Eleven Japan [Gam2007a], [Thevenet2006], [Thevenet2007c].

Ma principale contribution réside dans la proposition d'un méta-modèle qui définit formellement les concepts de l'alignement stratégique. Le méta-modèle intentionnel MAP est au cœur du méta-modèle d'alignement stratégique. L'emploi de ce méta-modèle pour spécifier l'alignement stratégique des entreprises Seven Eleven Japan [Gam2007a] [Thevenet2006] [Thevenet2007c] et BNP Paribas [Thevenet2007b] [Thevenet2008b] [Thevenet2009] m'ont permis de démontrer la faisabilité d'une *modélisation conceptuelle de l'alignement stratégique par des cartes intentionnelles*. Cette approche a visiblement attiré l'attention de la communauté puisque mes travaux sont cités par des experts du domaine tels que J Gordjin (principal contributeur de la méthode e3Value), E Dubois (animateur de l'atelier BUSITAL spécialisé sur le thème de l'alignement stratégique), C. Woo, ou encore K Cox et S Bleistein.

Dans les approches citées plus haut, l'alignement stratégique apparaît au travers d'une série de liens simples. Au contraire, notre approche de la modélisation de l'alignement stratégique repose sur une représentation de liens structurellement complexes [Thevenet2008]. INSTAL permet de modéliser *quatre structures de liens complexes et cinq types de rôles de l'alignement stratégique*. La sémantique de la relation d'alignement entre un couple d'entités peut être extrêmement variée en fonction de la situation. Notre typologie de cinq rôles offre un langage sémantiquement riche qui permet de représenter l'ensemble de ces situations. Par ailleurs, la modélisation de liens complexes permet de résoudre la problématique de l'explosion combinatoire rencontrée lorsque l'on veut représenter l'alignement stratégique par des collections de liens simples. Elle permet aussi de résoudre le problème de la transversalité de la stratégie: l'alignement d'un élément de la stratégie transverse à plusieurs éléments du niveau opérationnel peut être modélisé par un unique lien complexe d'alignement stratégique.

L'approche INSTAL adopte le paradigme intentionnel et permet de *modéliser l'alignement stratégique en terme de buts partagés par les entités alignées*. Cette approche est conforme à la définition de l'alignement stratégique selon Luftman et McLean selon laquelle l'alignement stratégique est "une application appropriée des technologies de l'information, en harmonie avec les stratégies métier, les buts et les besoins" [Luftman2004]. Dans les approches i* et SEAM citées ci-dessus, les intentions (ou buts) sont individuelles. L'alignement entre les intentions individuelles est représentés par des liens de dépendance (i*), ou par le fait que ces intentions sont

opérationnalisées par des fragments de processus métier communs (SEAM). L'idée d'utiliser les intentions partagées d'alignement pour représenter celui-ci, et non pas des dépendances entre des intentions individuelles est tout à fait distinctive de l'approche INSTAL. Les bénéfices de cette approche sont multiples. Tout d'abord, l'emploi du paradigme intentionnel permet de réduire les problèmes de discordance conceptuelle. En effet, les intentions modélisées subsument simultanément les intentions d'alignement des niveaux opérationnel et stratégiques. Par ailleurs, ces intentions peuvent être simultanément interprétés sous l'angle d'une finalité à atteindre (To-Be) et sous l'angle de ce qui devrait être (As-Is), ce qui permet de réduire les effets de l'antagonisme temporel.

Nous avons développé une *démarche méthodologique qui guide la découverte d'exigences d'évolution du Système d'Information et des processus métier par l'analyse de l'alignement stratégique*. INSTAL est une véritable démarche de coévolution des processus métier et du Système d'Information (une hypothèse fondamentale de la méthode INSTAL est INSTAL ne doit pas entraîner de modification de la stratégie). Deux types d'exigences sont spécifiées: les exigences d'alignement stratégique et les exigences d'évolution. Les exigences d'alignement stratégique sont spécifiées au moyen de cartes, comme selon le formalisme indiqué par le méta-modèle d'alignement stratégique INSTAL. Les exigences d'évolution sont spécifiées au moyen du formalisme GAP proposé par la méthode ACEM. Le principe de la méthode INSTAL est (a) de faire le bilan de la satisfaction des exigences d'alignement stratégique dans la situation actuelle (As-Is), (b) d'identifier les exigences d'évolution nécessaires à la mise en place d'une situation (To-Be) dans laquelle les bénéfices attendus de l'alignement stratégique seront obtenus (c) de propager les exigences d'évolution et de les valider. Le bilan de la satisfaction des exigences d'alignement est tiré de deux manières: par l'analyse du rôle des entités qui participent à l'alignement, et par l'emploi de métriques, mesures et seuils. Lorsque les critères de satisfaction des exigences d'alignements stratégique ne sont pas satisfaits, des exigences d'évolution doivent être définies. La propagation des exigences d'évolution est faite par l'élaboration de modèles To-Be au niveau opérationnel au moyen des exigences d'évolution. L'analyse des rôles et des métriques pour la situation To-Be permet de valider les exigences d'évolution, et de les réviser dans le cas où les exigences d'alignement stratégique ne sont toujours pas satisfaites. La méthode INSTAL peut être appliquée de manière continue: l'alignement stratégique est remis en cause à chaque fois que la stratégie change, ou l'un des éléments du niveau opérationnel évolue, ou que les mesures d'alignement stratégique font apparaître des dérives. Un nouveau cycle d'application de la méthode INSTAL peut être déroulé jusqu'à rétablissement de l'alignement stratégique.

5. Perspectives de recherche

On peut observer qu'un véritable courant de recherche s'est développé ces dernières années sur la question de la prise en compte de l'alignement dans les méthodologies d'ingénierie de Système d'Information. Mes travaux m'ont permis de constater que l'alignement soulève des problématiques spécifiques qui amènent à développer des concepts nouveaux et des démarches originales d'ingénierie des Systèmes d'Information.

Je suis convaincu que l'on peut même parler *d'ingénierie de l'alignement* comme sous-domaine de l'ingénierie des Systèmes d'Information, au même titre que l'ingénierie des méthodes ou que l'ingénierie des exigences.

L'ingénierie de l'alignement se définit comme l'activité qui consiste à construire, rétablir et faire évoluer l'alignement entre plusieurs entités.

L'ingénierie de l'alignement s'intéresse à des problèmes spécifiques, différents de ceux des autres sous-domaines de l'ingénierie des Systèmes d'Information. L'ingénierie de l'alignement doit donc trouver sa propre autonomie en proposant des outils et des méthodes pour étudier l'alignement sous tous ses aspects, tels que la définition, la représentation, la construction, ou l'évolution.

Mes travaux actuels sont menés dans deux directions: (a) le développement de systèmes de métriques et de moyens outillés pour les évaluer, et (b) la diversification des expérimentations empiriques des méthodes ACEM et INSTAL.

Notre définition de métriques, mesures et seuils à un niveau générique conduit à utiliser un processus pour générer des métriques spécifiques propres aux méta-modèles utilisés par les organisations. Un environnement informatisé générant des métriques spécifiques et calculant des mesures pour un projet donné faciliterait le travail de l'ingénieur d'alignement. Dans ce sens, nous sommes en train de développer une démarche d'ingénierie des systèmes d'aide à la décision. Notre démarche méthodologique, nommée CADWA, guide la découverte des exigences de systèmes d'aide à la décision et leur opérationnalisation au moyen de Data Warhouses [Gam2006a]. Nous pensons que CADWA peut être couplée à nos approches d'ingénierie de l'alignement ACEM et INSTAL, fournissant ainsi un environnement riche alliant l'étude de l'alignement sous l'angle de l'ingénierie à son étude sous l'angle de la performance d'entreprise [Gam2006b]. J'espère en particulier que les systèmes d'aide à la décision produits par CADWA soient utiles au diagnostic de l'alignement et au choix des exigences d'évolution découvertes au moyen d'ACEM et INSTAL. Par ailleurs, je souhaite que le rapprochement entre les domaines de la décision et de l'ingénierie de l'alignement permette d'enrichir ma bibliothèque de métriques par de nouvelles métriques conformes aux standards de l'entreprises, aux bonnes pratiques, et aux moyens offerts par les environnement de type Data Warehouse.

J'ai pratiqué de nombreuses études empiriques afin d'évaluer nos démarches d'ingénierie de l'alignement [Rolland2004], [Gam2006b], [Thevenet2007], [Viscusi2008]. Ces études ont non seulement permis de découvrir les problématiques fondamentales de l'alignement, et de valider notre approche de ces problématiques, mais aussi de comprendre que différents types d'alignement pouvaient générer différentes problématiques spécifiques, et donc requérir différentes méthodes. Je crois qu'il serait intéressant de poursuivre mes travaux en considérant deux catégories d'alignement supplémentaire: l'alignement avec la législation, et l'alignement de réseaux d'entreprises.

Les approches d'alignement telles que e3-value et SEAM ont démontré qu'il était possible de traiter du problème de l'alignement au sein d'un réseau d'entreprises par l'approche des constellations de valeurs. L'intérêt de ces approches est de rendre explicite la notion de valeur et ce faisant de permettre une étude plus approfondie de l'étude la faisabilité et des bénéfices du modèle économique de chaque entreprise et du réseau dans son ensemble. Aujourd'hui, ni l'approche ACEM, ni l'approche INSTAL ne permettent de rendre explicite le modèle économique d'un alignement (même stratégique) en terme de valeur pour des entreprises qui coopèrent. Les

bénéfices de l'alignement stratégique y sont en effet définis à titre individuel (pour chaque entreprise, son Système d'Information et/ou de ses processus métier). La problématique de la dynamique des bénéfices en termes d'échanges de valeurs doit, à mon avis être explorée pour un modèle d'alignement stratégique utilisable par les décideurs dans le cadre de leur réflexion sur les alliances stratégiques.

Les travaux sur l'alignement législatif [Breux2006], [Rifaut2008], [Ghanavati2007] ont déjà démontré que celui-ci présente certaines caractéristiques de l'alignement système-processus (par exemple il est possible de produire des modèles conceptuels de la législation), et des caractéristiques de l'alignement stratégique (par exemple la transversalité). Ce contexte d'application offre une bonne opportunité de se baser sur les méthodes ACEM et INSTAL pour développer une approche plus générale qui permettrait de traiter toutes les situations d'alignement, dont l'alignement législatif.

Enfin, d'autres travaux devront être entrepris, peut être à plus lointaine échéance pour explorer le problème de la dispersion générée par les questions d'alignement et d'évolution des Systèmes d'Information. Pour l'instant, les démarches d'ingénierie de l'alignement que nous avons développées m'ont servi à guider la découverte d'exigences d'évolution des Systèmes d'Information et des processus d'entreprise. Or dans la pratique, les directeurs de Système d'Information sont préoccupés par le coût qu'entraîne les évolutions des Systèmes d'Information et ils ont profondément besoin d'outils leur permettant de diagnostiquer l'aptitude des Systèmes d'Information à évoluer, ou mieux encore à susciter l'évolution de l'entreprise par la création de valeur qu'elles permettent.

CHAPITRE 6

Conclusions Générales

L'équipe "Ingénierie des Exigences" que je dirige actuellement au sein du Centre de Recherche en Informatique cherche à maintenir le niveau d'excellence scientifique initié par Colette Rolland au niveau international en publiant intensivement dans les conférences les plus sélectives du domaine (taux d'acceptation < 20%). Cet objectif est atteint puisque depuis 2004, l'équipe a régulièrement publié aux conférences RE (3 articles publiés depuis 2004), CAiSE (5 articles), REFSQ (3 articles), ou dans la revue REJ (2 articles).

Mes travaux de recherche sur l'ingénierie des exigences ont toujours été motivés par des problématiques rencontrées à la fois sur le terrain au travers de mes contacts avec les industriels, et au sein de la communauté scientifique internationale dans le cadre de collaborations avec des chercheurs européens, ou lors de ma participation à des congrès. En particulier, ma très forte implication dans le projet Européen CREWS, dans l'animation de l'atelier REFSQ, et dans les collaborations industrielles (France Télécom, Renault DIAC, Renault groupe, EDF, BNP Paribas, SNCF, STAGO Instruments INCOSE) m'ont imposé une rigueur scientifique et une créativité que j'ai pu valoriser dans ma production scientifique personnelle ainsi que dans mes activités d'enseignement, et d'encadrement doctoral. Cette rigueur et cette créativité me semblent toutes deux essentielles pour asseoir une activité de recherche sur des bases qualitatives non réfutables.

Le contexte de la communauté internationale, l'encadrement doctoral, la collaboration avec le milieu industriel et la valorisation par la formation sont des éléments très motivants de mon activité. Je désire les conserver pour acquérir une reconnaissance internationale et nationale, mais surtout avec le souhait profond de développer une véritable communauté d'ingénierie des exigences en France, comme on peut en rencontrer au Royaume Uni, en Allemagne, ou en Suède.

ANNEXES

Références

- [Aiken1994] P. Aiken, AL Munz, R. Richards. *DoD Legacy Systems: Reverse Engineering Data Requirements*. Communications of the ACM, Vol. 37, No. 5, pp26-41. Mai 1994.
- [Al-Jadir03] Al-Jadir L. *Once Upon a Time a DTD Evolved into Another DTD*. Object Oriented Information Systems, Lecture Notes in Computer Science, Vol.2817, pp226-237. 2003.
- [Al-Jadir95] L. Al-Jadir, T. Estier, G. Falquet, M. Léonard. *Evolution Features of the F2 OODBMS*. Actes de 4th Int. Conf. on Database Systems for Advanced Applications (DASFAA'95), World Scientific Press, Advanced Database Research and Development, Vol. 5. Singapour, 1995.
- [Anderson2001] S. Anderson, M. Felici. *Requirements Evolution From Process to Product Oriented Management*. Actes de 3rd International Conference on Product Focused Software Process Improvement, LNCS 2188, Springer-Verlag, pp 27-41. Kaiserslautern, Allemagne, Septembre 10-13, 2001.
- [Anderson2002] S. Anderson M. Felici. *Quantitative Aspects of Requirements Evolution*. Actes de 26th Computer Software and Application Conference (COMPSAC'02). Oxford, Angleterre, Août 2002.
- [Anthony1965] R.N. Anthony. *Planning and Control Systems: A framework for Analysis*. Harvard University, Boston UNSW. 1965.
- [Anton1996] A. Antón. Goal-Based Requirements Analysis. Actes de 2nd International Conference on Requirements Engineering (ICRE'96). Colorado, USA. Avril 1996.
- [Antón2001] A.I. Antón, C. Potts. *Functional Paleontology: System Evolution as the User Sees It*. Actes de 23rd IEEE International Conference on Software Engineering (ICSE'01), pp 421-430. Toronto, Canada, Mai 2001.
- [Arsanjani2001] A. Arsanjani, J. Alpigini. *Using Grammar-oriented Object Design to Seamlessly Map Business Models to Component-based Software Architectures*. Actes de International Symposium of Modelling and Simulation, pp 186-191. Pittsburgh, PA, USA. Mai 2001
- [Avison1994] D. E. Avison, P. L. Powel and C. Adams. *Identifying and Incorporating Change in Information Systems*. System Practice, Vol. 7, No. 2, pp143-159. 1994.
- [Avison2004] D. Avison, J. Jones, P. Powell, D. Wilson. *Using and Validating the Strategic Alignment Model*. Journal of Strategic Information Systems, Vol. 13, 2pp. 23-246. 2004.
- [Banerjee1987] J. Banerjee, W. Kim, H.J. Kim H.F. Korth. *Semantics and Implementation of Schema Evolution in OODB*. Actes de la Conférence ACM SIGMOD on Management of Data, ACM, pp 311-322. 1987.
- [Batory2005] D. Batory. *Feature Models, Grammars, and Propositional Formulas*. Actes de 9th International Software Product Lines Conference (SPLC'05), pp 7–20. 2005.

- [Benachour1997] C. Ben Achour. *Linguistic Instruments for the Integration of Scenarios in Requirement Engineering*. Actes de Requirements Engineering:Foundation for Software Quality (REFSQ'97). Espagne. Juin 1997.
- [Benachour1998a] C. Ben Achour, C. Rolland, C. Souveyet. *A Proposal for Improving the Quality of the Organisation of Scenario Collections*. Actes de Requirements Engineering:Foundation for Software Quality (REFSQ'98). Italie, Juin 1998.
- [Benachour1998b] C. Ben Achour. *Guiding Scenario Authoring*. Actes de European-Japanese Conference on Information Modelling and Knowledge Bases (EuroJap'98). Finland, Mai 1998.
- [Benachour1998c] C. Ben Achour. *Writing and Correcting Textual Scenarios for System Design*. Natural Language and Information Systems (NLIS'98). Autriche, Août 1998.
- [Benachour1999a] C. Ben Achour. *Extraction des Besoins par Analyse des Scenarios Textuels*. Thèse de Doctorat. Université Paris 6 – Pierre et Marie Curie. 1999.
- [Benachour1999b] C. Ben Achour, C. Souveyet, M. Tawbi. *Bridging the Gap between Users and Requirements Engineering: the Scenario-Based Approach*. Journal of Computer Systems Science & Engineering (JCSSE), Special issue: Object-Oriented Information Systems, Vol.14 No.6, pp 112 – 122. Février 1999.
- [Benachour1999c] C. Ben Achour, C. Rolland, N. M. Maiden, C. Souveyet. *Use Case Authoring Guidance: Results of an Empirical Study*. Actes de 4th IEEE International Symposium on Requirements Engineering (RE'99). Limerick, Irlande, Juin 1999.
- [Benavides2005] D. Benavides, P. Trinidad A. Ruiz-Cortés. *Using Constraint Programming to Reason on Feature Models*. Actes de 17th International Conference on Software Engineering and Knowledge Engineering (SEKE'05). Taipei, China. 2005
- [Berander2005] P. Beraner, A. Andrews. *Requirements Prioritization*. Chapitre du Livre: Engineering and Managing Software Requirements, A. Aurum, C. Wohlin (editeurs), Springer Verlag. 2005.
- [Bergeron1999] F. Bergeron, L. Raymond, S. Rivard. *Conceptualizing and Analyzing Fit in Information Systems Research: An Empirical Comparison of Perspectives*. Cahier du GreSI No. 99-03. Montreal, Canada. 1999.
- [Besançon1999] R. Besançon, M. Rajman, J.C. Chappelier. *Textual Similarities on a Distributional Approach*. Actes de 10th International Workshop on Database and Expert Systems Applications (DEXA'99). Florence, Italie, Septembre 1999.
- [Bezivin2001] J. Bezivin. *From Object Composition to Model Transformation with the MDA*. Actes de 39th International Conference on Technology of Object-Oriented Languages and Systems (TOOLS'01). Santa Barbara, USA, Juillet-Août 2001
- [Bhuiyan2006] M. Bhuiyan, M. Islam, A. Krishna, A. Ghose. *Co-evolution of Agent Oriented Conceptual Models and Use Case Diagrams*. Actes 6th International Conference on Quality Software (QSIC'06). 2006.

- [Bianco1999] G. Bianco, V. De Antónellis, S. Castano, M. Melchiori. *A Markov Random Field Approach for Querying and Reconciling Heterogeneous Databases*. Actes de 10th International Workshop on Database and Expert Systems Applications (DEXA'99). Florence, Italie, Septembre 1999.
- [Bleistein2006] S. Bleistein, K. Cox, J. Verner, K. Phalp. *B-SCP: A Requirements Analysis Framework for Validating Strategic Alignment of Organizational IT Based on Strategy, Context, and Process*. Revue Information and Software Technology, Vol. 46. 2006.
- [Bodhuin2004] T. Bodhuin, R. Esposito, C. Pacelli M. Tortorella. *Impact Analysis for Supporting the Co-Evolution of Business Processes and Supporting Software Systems*. Actes de Workshop on Creating and Maintaining the Fit between Business Processes and Support Systems (BPMDS'04) associé à la conférence CAiSE'04. Riga, Lituanie, 2004.
- [Boehm1981] B. Boehm. *Software Engineering Economics*. Prentice Hall. 1981.
- [Boehm2001] B. Boehm, V. R. Basili. *Software Defect Reduction Top 10 List*. IEEE Computer, Vol.34 No.1, pp 135-137. Janvier 2001.
- [Breaux2006] T. Breaux, A. Antón. *Analyzing Goal Semantics for Rights, Permissions, and Obligations*. Actes de 13th IEEE International Conference on Requirements Engineering (RE'05), pp 177-188. Paris, France, Septembre 2005.
- [Breton2002] E. Breton. *Contribution à la Représentation de Processus par des Techniques de Métamodélisation*. Thèse de Doctorat de l'Université de Nantes. N°0366-066. Juin 2002.
- [Bubenko1994] J. Bubenko, C. Rolland, P. Loucopoulos, V. De Antonellis. *Facilitating Fuzzy to Formal Requirements Modelling*. Actes de 1st International Conference on Requirements Engineering (ICRE'94). Colorado, USA. 1994
- [Bunge1977] M. Bunge. *Treatise on Basic Philosophy: Ontology I. The Furniture of the World*, Reidel. 1977.
- [CapGémini2008] CapGemini. *CIO Survey 2008*. <http://www.cio-online.com/actualites/lire-les-dsi-entre-innovation-et-impuissance-1248.html>
- [Casati1996] F. Casati, S. Ceri, B. Pernici, G. Pozzi. *Workflow Evolution*. Actes de 15th International Conference on Conceptual Modeling (ER'96), pp 438-455. Cottbus, Allemagne 1996.
- [Castano1993] S. Castano, V. De Antónellis. *A Constructive Approach to Reuse of Conceptual Components*. Actes de 2nd IEEE International Workshop on Software Reusability. R. Prieto-Diaz, W.B. Frakes (Eds). Lucca, Italie, 1993.
- [Cavano1988] J.P. Cavano, J.A. McCall. *A framework for the management of quality*. Actes de ACM Software Assurance Quality Assurance Workshop, pp 133-139. 1988.
- [Chan2007] Y. Chan, B. Reich. *IT alignment: an Annotated Bibliography*. Journal of Information Technology, N° 22, pp316–396. Octobre 2007.

- [Chen76] P. Chen. *The Entity-Relation Model - Towarda Unified View of Data*. ACM Transactions on Database System, Vol. 1, No. 1, pp 9-36. Mars 1976.
- [Ciborra1997] C. Ciborra., *De profundis? Deconstructing the Concept of Strategic Alignment*. Scandinavian Journal of Information Systems Vol. 9 n°1, pp 67–82. 1997
- [CIGREF2002] CIGREF. *Alignement Stratégique du Système d'Information – Comment faire du Système d'Information un Atout pour l'Entreprise?* Rapport du CIGREF. Septembre 2002.
- [CIGREF2007] CIGREF. *Changement et transformation du SI*. Note de synthèse. 2007
- [Cleve2005] A. Cleve, J. Henrard, J.L. Hainaut. *Co-Transformations in Information System Reengineering*. Revue Theoretical Computer Science Vol5, No. 15. 2005.
- [Cockburn1996] A. Cockburn. *Structuring Use Cases with Goals*. Technical report, Human and Technology, 1996.
- [CSC2001] Computer Sciences Corporation. *Critical Issues of IS Management*. 2001
- [Dahlstedt2005] A Dahlstedt, A. Persson. *Requirements Interdependencies: State of the Art and Future Challenges*. Chapitre du livre "Engineering and Managing Software Requirements". Aybüke Aurum, Claes Wohlin (editeurs), Springer Berlin Heidelberg. 2005.
- [Dardenne1993] A. Dardenne, A. van Lamsweerde, S. Fickas. *Goal-Directed Requirements Acquisition*. Science of Computer Programming, Vol.20, pp 3-50. 1993
- [Delgado2003] C. Delgado, J. Samos, M.Torres. *Primitive Operations for Schema Evolution in ODMG Databases*. Object Oriented Information Systems, Lecture Notes in Computer Science, Vol.2817, pp226-237. 2003.
- [Diamantini] C. Diamantini, M. Panti. *A Conceptual Indexing Method for Content-Based Retrieval*. Actes de 10th International Workshop on Database and Expert Systems Applications (DEXA'99). Florence, Italie, Septembre 1999.
- [Djebbi2007] O. Djebbi, C. Salinesi. *RED-PL, a Method for Deriving Product Requirements from a Product Line Requirements Model*. Actes de International Conference on Advanced information Systems Engineering (CAISE'07), Springer Verlag. Trondheim, Novège, Juin 2007.
- [Easterbrook1991] S. Easterbrook. *Handling Conflict Between Domain Descriptions With Computer-Supported Negotiation*. Revue Knowledge Acquisition, Vol. 3, pp.255-289. 1991.
- [Esteves2001] J. Esteves, J. Pastor. *Enterprise Resource Planning Systems Research: an Annotated Bibliography*. Revue Communications of the Association for Information Systems (CAIS), Vol. 7, No. 8. 2001.
- [Esteves2002] J. Esteves, J. Pastor, J. Casanovas. *Monitoring Business Process Redesign in ERP Implementation Projects*. Actes de Americas Conference on Information Systems. Dallas, USA, 2002.
- [Estublier2000] J. Estublier, M. Nacer. *Schema Evolution in Software Engineering Databases -- A new Approach in Adele environment*. CAI Computer and Artificial Intelligence Journal. Vol. 19. pp. 183-203. Juin 2000.

- [Etien2003a] A. Etien, C. Salinesi. *Towards a Systematic Definition of Requirements for Software Evolution: A Case-study Driven Investigation*. Actes de Evaluation of Modeling Methods in Systems Analysis and Design (EMMSAD), (en association avec CAISE'03), Velden, Autriche, Juin 2003.
- [Etien2003b] A. Etien, R. Deneckère, C. Salinesi. *Extending Methods to Express Change Requirements*. Actes de Engineering Methods to Support Information Systems Evolution (EMSISE), associée à la conférence OOIS'03, Genève, Suisse, Septembre 2003.
- [Etien2004a] A. Etien, C. Salinesi. *Evolution Patterns to Elicit Compliance Requirements*. Actes de European-Japanese Conference on Information Modelling and Knowledge Bases (EuroJap'04), Skövde, Suède, Juin 2004.
- [Etien2004b] A. Etien, C. Rolland, C. Salinesi. *Overview of a Gap-driven Evolution Process*. Actes de Australian Workshop on Requirements Engineering (AWRE'04), Adelaïd, Australie, Decembre 2004.
- [Etien2005] A. Etien, C. Salinesi. *Managing Requirements in a Co-evolution Context*. Actes de IEEE International Conference on Requirement Engineering (RE'05), pp 125 – 134. Paris, France, Septembre 2005.
- [Etien2006a] A. Etien. *ACEM Analyse et Correction des Exigences d'Evolutions*. Thèse de Doctorat de l'Université Paris 1. Paris, France, Mars 2006.
- [Etien2006b] A. Etien, C. Rolland, C. Salinesi. *A Meta-modelling Approach to Express Change Requirements*, Actes de International Conference on Software Engineering and Data Technologies (ICSOFT), Session spéciale Meta-modelling. Setubal, Portugal, Septembre 2006.
- [Evans1997] R. Evans, S. Park, H. Alberts. *Decisions not Requirements: Decision centered Engineering of Computer-Based Systems*. Actes de International Conference on Engineering and Computer-Based Systems, pp 435-442. 1997.
- [Feather1987] M. Feather. *Language Support for the Specification and Development of Composite Systems*. ACM Transactions on Programming Languages and Systems Vol. 9, No. 2, pp 198-234. Avril 1987.
- [Fimbel2006] E. Fimbel. *DGAM (Dynamic Global Alignment Model): a New Alignment Model in Response to the Limits of the SAM Model and the Shifting Expectations and Processes of Practitioners*. Actes du 11ème Colloque de l'AIM. Luxembourg, Juin 2006.
- [Finkelstein2002] A. Finkelstein, D. Bush. *Requirements Elicitation for Complex Safety Related Systems*. Actes de London Communication Symposium. Londres, Royaume Uni, Septembre 2002.
- [Gam2006a] I. Gam, C. Salinesi. *A Requirement-driven Approach for Designing Data Warehouses*. Actes de Requirements Engineering:Foundation for Software Quality (REFSQ'06) associée à la conférence CAiSE'06. Luxembourg, Luxembourg. Juin 2006.
- [Gam2006b] I. Gam, LH Thevenet, C. Salinesi. *Documenter l'alignement d'un ED avec la stratégie d'entreprise en vue de mieux satisfaire les exigences des décideurs*. Revue Ingénierie des Systèmes d'Information, Vol. 11No. 6 pp83-114. 2006.

- [Garlan1995] D. Garlan, R. Allen, J. Ockerbloom. *Architectural Mismatch: Why Reuse is so Hard*. Revue IEEE Software, Col.12, No3, pp17-26. Novembre 1995.
- [Ghanavati2007] S. Ghanavati, D. Amyot, L. Peyton. *Towards a Framework for Tracking Legal Compliance in Healthcare*. Actes de International Conference on Advanced Information Systems Engineering (CAiSE'07), pp218-232. Trondheim, Norvège, Juin 2007.
- [Glass2006] R. Glass. *The Standish Report: Does It Really Describe a Software Crisis?* Revue Communications of the ACM. Vol49, No. 8, pp15-16. Août 2006.
- [Glinz2005] M. Glinz. *Rethinking the Notion of Non-Functional Requirements*. Proceedings of the Third World Congress for Software Quality (3WCSQ 2005), Vol. II, 55-64. Munich, Allemagne, 2005.
- [Goguen2002] J. Goguen, G. Rosu. *Institution Morphisms*. Revue Formal Aspects of Computing, Vol13, No3-5. Juillet 2002.
- [Han1997] J. Han. *Supporting Impact Analysis and Change Propagation in Software Engineering Environments*?. Actes du Workshop on Software Technology and Engineering Practice (STEP'97 / CASE'97), IEEE Computer Society Press, pp. 172-182. London, UK, Juillet 1997.
- [Harker1993] S. D. P. Harker, K. D. Eason, J. E. Dobson. *The Change and Evolution of Requirements as a Challenge to the Practice of Software Engineering*. Actes de 1st IEEE Symposium on Requirements Engineering (RE'93), 1993, pp 266-272. Californie, USA, 1993.
- [Henderson1989] Henderson, J., Venkatramen, N., 1989. *Strategic Alignment: A Model for Organisational Transformation*. Dans Kochan, T., Unseem, M. (Eds.), 1992. Transforming Organisations. OUP, New York. 1989.
- [Henderson1993] J.C. Henderson, N. Venkatraman. *Strategic Alignment: Leveraging Information Technology for Transforming Organizations*. IBM Systems Journal, Vol. 32, No1, pp 4-16, 1993. (Ré-imprimé dans IBM Systems Journal, Vol. 38, No2, 1999)
- [Hoare1978] C.A.R. Hoare. *Communicating Sequential Processes*. Communications of the ACM Vol. 21 No. 8 pp 666-677, Août 1978.
- [IEEE1998] Institute for Electrical and Electronics Engineering. *IEEE recommended practice for software requirements specifications*. Standard IEEE 830-1998. Octobre 1998.
- [Ivankina2004a] E. Ivankina, C. Salinesi. *Adapting Ishikawa's Cause-Effect Diagrams to Understand Requirements Threats*. Actes de Conference on Business Information Systems (BIS'04). Pologne, Avril 2004.
- [Ivankina2004b] E. Ivankina, C. Salinesi. *An Approach to Guide Requirement Elicitation by Analysing the Causes and Consequences of Threats*. Actes de European-Japanese Conference on Information Modelling and Knowledge Bases (EuroJap'04). Suède, Juin 2004.

- [Ivankina2004c] C. Salinesi, E. Ivankina. *Guiding Use Case Discovery By Analysing Threats*. Actes du Forum de International Conference on Advanced Information Systems Engineering (CAISE'04 Forum), Springer Verlag. Riga, Lituanie, Juin 2004.
- [Ivankina2005] E. Ivankina, C. Salinesi. *Eliciting Requirements by Analysing Threats Caused by Users*. Actes de Software Engineering Research, Management and Applications (SERA), IEEE Computer Society Press, Mont. Plaisance. USA, Août 2005.
- [Jacobs1993] S. Jacobs, M. Jarke, K. Pohl. *Report on the First International IEEE Symposium on Requirements Engineering (RE93)*. Requirements Engineering Newsletter No. 6. Juin 1993.
- [Jarke1994] M. Jarke, K. Pohl. *Requirements Engineering in 2001: (virtually) managing a changing reality*. Software Engineering Journal, Vol.9, No.6, pp 257-266. Novembre 1994.
- [Jarke1997] M. Jarke, K. Pohl, P. Haumer, K. Weidenhaupt, E. Dubois, P. Heymans, C. Rolland, C. Ben Achour, C. Cauvet, J. Ralyté, A. Sutcliffe, N.A.M. Maiden, S. Minocha. *Scenario Use in European Software Organizations – Results from Site Visits and Questionnaires*. Rapport CREWS n° 97-10. 1997.
- [Jiliani1997] L. Jilani, R. Mili, A. Mili. *Approximate Component Retrieval: An Academic Exercise or a Practical Concern?*. Actes de 8th Workshop on Institutionalising Software Reuse (WISR8). Columbus, Ohio, USA. Mars 1997.
- [Jorgensen2006] M. Jorgensen, K. Molokken. *How Large are Software Cost Overruns? Review of the 1994 Chaos Report*. Revue Information and Software Technology. Vol. 48, No. 4. Avril 2006.
- [Kaplan2006] R. Kaplan, D. Norton. *Alignment – Using the Balanced Scorecard to Create Corporate Strategy*. Harvard Business School Press. 2006.
- [Kardasis1998] P. Kardasis, P. Loucopoulos. *Aligning Legacy Information Systems to Business Processes*. Actes de International Conference on Advances in Information Systems (CAiSE'98), pp. 25 – 40. Pise, Italie. Juin 1998.
- [Kefi2006] H. Kéfi, A. Shwarz, M. Kalika. *Modèle Basé sur les Processus Versus Aligement Stratégique: Quels Facteurs Explicatifs de la Performance*. Actes du Colloque de l'AIM, Luxembourg, Juin 2006.
- [Knoll1994] K. Knoll, S. L. Jarvenpaa. *Information Technology Alignment or “fit” in Highly Turbulent Environments: the Concept of Flexibility*. Actes de Computer Personnel Research Conference on Reinventing IS. Alexandria, Virginia, USA. 1994.
- [Kornyshova2007a] E. Kornyshova. *Développement d'une Démarse Visant à Introduire l'Aide à la Décision Multicritère dans l'Ingénierie des Systèmes*. Poster au Forum de l'Association Française d'Ingénierie Système (AFIS'07). France, 2007.
- [Kornyshova2007b] E. Kornyshova, C. Salinesi. *Business Process Priorisation with Multicriteria Methods: Case of Business Process Reengineering*. Actes de International Conference on Enterprise Information Systems (ICEIS'07). Portugal, Juin 2007.

- [Kornyshova2007c] E. Kornyshova, C. Salinesi. *MCDM Techniques Selection Approaches: State of the Art*. Actes de IEEE Symposium on Computational Intelligence in Multicriteria Decision Making (MCDM'07), Honolulu, HI, USA, Avril 2007.
- [Kornyshova2007d] E. Kornyshova, R. Deneckère, C. Salinesi. *Method Chunks Selection by Multicriteria Techniques: an Extension of the Assembly-based Approach*. Actes de Situational Method Engineering (SME'07). Genève, Suisse. Septembre 2007.
- [Kornyshova2008] E. Kornyshova, C. Salinesi. *Selecting MCDM Techniques: State of the Art*. International Journal of Information Technology and Intelligent Computing (IT&IC). Février 2008.
- [Kradolfer2000] M. Kradolfer. *A Workflow Metamodel Supporting Dynamic, Reuse-based Model Evolution*. Thèse de Doctorat, Department of Information Technology, University of Zurich, Switzerland, chap. 4, pp 59-73. Mai 2000.
- [Krishna2004] A. Krishna, A.K. Ghose, S. Vilkomir. *Co-Evolution of Complementary Formal and Informal Requirements*. Actes de International Workshop on Principles of Software Evolution (IWPE'04), pp 159-164. Kyoto, Japan. Septembre, 2004.
- [Krumbholz00] M. Krumbholz, N.A.M. Maiden. *How Culture Might Impact on the Implementation of Enterprise Resource Planning Packages*. Actes de International Conference on Advances in Information Systems (CAISE'00), Springer Verlag. Stokholm, Suède. Juin 2000.
- [Kueng1997] P. Kueng, P. Kawalek. *Goal-based Business Process Models: Creation and Evaluation*. Business Process Management Journal, Vol. 3, pp 17-38. 1997.
- [Lämmel2004] R. Lämmel. *Coupled Software Transformations*. Actes de International Workshop on Software Evolution Transformation (SET'04), pp31-35. 2004.
- [Lamsweerde2004] A. van Lamsweerde. *Goal Oriented Requirements Engineering: a Roundtrip from Research to Practice*. Invited Keynote. Actes de 12th IEEE Joint International Requirements Engineering Conference (RE'04), pp2-4. Kyoto, Japon. Septembre 2004.
- [Larman2003] C. Larman, V. Basili. *Iterative and Incremental Development: A Brief History*. Revue IEEE Computer, Vol. 36, no. 6, pp 47-56. Juin. 2003
- [Lehman2000] M.M Lehman, J.F Ramil and G Kahen. *Evolution as a Noun and Evolution as a Verb*. Actes de Workshop on Software and Organisation Co-evolution. Londres, Royaume Uni. 2000.
- [Longépé2002] C. Longépé. *The Enterprise Architecture IT Project – the Urbanisation Paradigm*. Penton. 2002.
- [Loucopoulos1999] P. Loucopoulos, V. Kavakli. *Goal-driven Business Process Analysis Application in Electricity Deregulation*. Revue Information Systems Journal, Vol. 24, No. 3, pp 187-207. Mai 1999.
- [Luftman1996] J. Luftman. *Competing in the Information Age*. Oxford University Press, 1996.
- [Luftman2004] J Luftman, ER McLean. *Key issues for IT executives*. MIS Quarterly Executive, Vol.6. 2004

- [Luftman2007] J. Luftman, R. Kempaiah. *An Update on Business-IT Alignment: « a Line Has Been Drawn »*. MIS Quarterly Executive, Vol. 6, No. 3. Septembre 2007.
- [Lutz2005] R. Lutz, S. Nelson, A. Patterson-Hine, C. Frost D. Tal. *Identifying Contingency Requirements Using Obstacle Analysis*. Actes de 13th IEEE Requirements Engineering Conference (RE'05), pp 263-272. Paris, France. Septembre 2005
- [Maes2000] R. Maes, D. Rijsenbrij, O. Truijens, H. Goedvolk. *Redefining Business – IT alignment through a Unified Framework*. Landelijk Architectuur Congres. Amsterdam. Pays Bas. 2000.
- [Mannion2002] M. Mannion. *Using First-Order Logic for Product Line Model Validation*. Actes de 2nd Software Product Line Conference (SPLC2), LNCS 2379, pages 176–187. San Diego, Californie USA. 2002.
- [McCracken1982] D. McCracken, M. Jackson. *Lifecycle Concepts Considered Harmful*. ACM SIGSOFT Software Engineering, Vol. 7, No. 2, pp 29 – 32. 1982.
- [Mens2001] T. Mens and S. Demeyer. *Evolution metrics*. Actes de 4th International Workshop on Principles of Software Evolution (IWPSE '01), en association avec ESEC/FSE'01. Vienne, Autriche. Septembre 2001.
- [Mens2005] T. Mens, M. Wermelinger, S. Ducasse, S. Demeyer, R. Hirschfeld, M. Jazayeri. *Challenges in Software Evolution*. Actes de International Workshop on Principles of Software Evolution (IWPSE'05), en association avec ESEC/FSE'05. Lisbonne, Portugal. Septembre 2005.
- [Milner1980] R.A. Milner. *Calculus of Communicating Systems*. Lecture Notes in Computer Science, 92. Springer-Verlag, New York, 1980.
- [Mitleton-Kelly2000] E. Mitleton-Kelly, M-C Papaefthimiou. *Co-evolution & an enabling infrastructure: a solution to legacy?* Chapitre 4 du livre « Systems engineering for business process change », Peter Henderson, Springer-Verlag. 2000.
- [MOF] <http://www.omg.org/cgi-bin/apps/doc?formal/02-04-03.pdf>
- [Moody2005] D. Moody. *Theoretical and Practical Issues in Evaluating the Quality of Conceptual Models: Current State and Future Directions*. Revue Data & Knowledge Engineering Vol. 55, No. 3, pp 243-276. Décembre 2005.
- [Moreira2006] A. Moreira, J. Araújo, J. Whittle. *Modeling Volatile Concerns as Aspects*. Actes de International Conference on Advanced information Systems Engineering (CAiSE'06), pp 544-558. Luxembourg, Juin 2006.
- [Mylopoulos1992] J. Mylopoulos, L. Chung, B. Nixon. *Representing and Using Nonfunctional Requirements: A Process-Oriented Approach*. Revue IEEE Transactions on Software. Engineering, Vol. 18, No. 6, Juin 1992.
- [NattOchDag2001] J. Natt och Dag, B. Regnell, P. Carlshamre, A. Andersson, J. Karlsson. *Evaluating Automated Support for Requirements Similarity Analysis in Market-driven Development*. Actes de 7th International Workshop on Requirements Engineering: Foundations of Software Quality, (REFSQ'01). Interlaken, Suisse. Juin 2001.

- [Ncube00] C. Ncube, N. Maiden. *COTS Software Selection: The Need to Make Tradeoffs Between System Requirements, Architectures and COTS/Components*. Actes de COTS*2000 Workshop (associé à la conférence ACSE'00). Limerick, Ireland, 2000.
- [Nurcan1999] S. Nurcan, J. Barrios, G. Grosz, C. Rolland. *Change Process Modelling Using the EKD-Change Management Method*. Actes de European Conference on Information Systems (ECIS'99). Copenhagen, Danemark. Juin 1999.
- [Nurmuliani2004] N. Nurmuliani, D. Zowghi and S. Fowell. *Analysis of Requirements Volatility during Software Development Life Cycle*. Actes de Australian Software Engineering Conference (ASWEC'04). Melbourne, Australie. Avril 2004
- [Osterwalder2002] A. Osterwalder, Y. Pigneur. *An e-Business Model Ontology for Modeling e-Business*. Actes de 15th Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy. Bled, Slovenie. Juin 2002.
- [Papadacci2005a] E. Papadacci Stephanopoli, C. Salinesi. *Arbitrating between Enterprise Objectives. NeNo. – an Enterprise Architecture Approach developed at Renault*. Actes de Requirements Engineering Decision Support (REDECS'05) associé à la Conférence RE'05. Paris, France. Août 2005.
- [Papadacci2005b] E. Papadacci Stephanopoli, C. Salinesi. *NENO: Une Approche d'Aide à l'Arbitrage par l'Evaluation Qualitative et Quantitative de la Valeur Métier de SI durant la Phase Amont du Processus d'Urbanisation*. Actes de Ingénierie et gestion des processus d'entreprise (ECI), à l'initiative du GT «Entreprise Communicante et Interopérabilité» CNRS GDR I3 et GDR MACS. France, 2005.
- [Papadacci2006] E. Papadacci Stephanopoli, C. Salinesi, C. Rolland. *NENO process: Information systems arbitration process in Enterprise Architecture Project*. Actes de IEEE Conference on Information and Communication Technologies: from Theory to Applications (ICTTA'06). Damascus, Syrie, pp 105 – 106. Avril 2006.
- [Papadopoulos1999] A. Papadopoulos, Y. Manolopoulos. *Structure-Based Similarity Search with Graph Histograms*. Actes de 10th International Workshop on Database and Expert Systems Applications (DEXA'99). Florence, Italie. Septembre 1999.
- [Pijpers2007] V. Pijpers, J. Gordijn. *e3forces: Understanding Strategies of Networked e3value Constellations by Analyzing Environmental Forces*. Actes de International Conference on Advanced Information Systems Engineering (CAiSE07), pp188-202. Trondheim, Norvège. Juin 2007.
- [Poels2000] G. Poels, S. Viaene, G. Dedene. *Distance Measure for Information System Reengineering*. Actes de 12th Conference on Advanced Information Systems Engineering CAISE'00. Stockholm, Suède, Juin 2000.
- [Pohl1994] K. Pohl, S. Jacobs. *Concurrent Engineering: Enabling Traceability and Mutual Understanding*. Journal of Concurrent Engineering Research and Application, Special Issue on Concurrent Engineering and Artificial Intelligence, Vol.2, No.4, pp 279-290. 1994
- [Pohl96] K. Pohl. *Process-Centered Requirements Engineering*. John Wiley & Sons Inc. 1996.

- [Potts1994] C. Potts, K. Takahashi, A. Antón. *Inquiry-based requirements analysis*. Revue IEEE Software Vol. 11 No. 2, pp 21-32. 1994.
- [Potts97] C. Potts. *Fitness for Use: The System Quality that Matters Most*. Actes de 3rd International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'97) associé à la conférence CAiSE'97. Barcelone, Espagne. Juin1997
- [Prat97] N. Prat. *Goal Formalisation and Classification for Requirements Engineering*. Actes de 3rd International Workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'97) associé à la conférence CAiSE'97, pp 145-156. Barcelone, Espagne, 1997.
- [Prieto-Diaz1987] R. Prieto-Diaz, P. Freeman. *Classifying Software for Reusability*. Revue IEEE Software, Vol. 4 No. 1. 1987.
- [Pujol2008] M. Pujol. *L'Alignement IT/Métier en France*. Livre Blanc Pierre Audoin Consultant. 2008.
- [Ralyté2004] J. Ralyté, C. Rolland, R. Deneckère. *Towards a Meta-Tool for Change-Centric Method Engineering: a Typology of Generic Operators*. Actes de International Conference on Advances of Information Systems (CAISE'04). Riga, Lithuanie. Juin 2004.
- [Rifaut2008] A. Rifaut, E. Dubois. *Using Goal Oriented Requirements Engineering for Improving the Quality of ISO/IEC 15504 based Compliance Assessment Frameworks*. Actes de 16th IEEE International Requirements Engineering Conference (RE'08), pp33-42. Barcelone, Espagne, Septembre 2008.
- [Robey02] D. Robey, J.W. Ross, M-C. Boudreau. *Learning to Implement Enterprise Systems: An Exploratory Study of the Dialectics of Change*. Journal of Management Information Systems. Eté 2002
- [Rolland1998a] C. Rolland, C. Ben Achour, C. Cauvet, J. Ralyte, A. Sutcliffe, N. M. Maiden, M. Jarke, P. Haumer, K. Pohl, E. Dubois, P. Heymans. *A Proposal for a Scenario Classification Framework*. Requirements Engineering Journal (REJ), Vol.3, No.1. 1998.
- [Rolland1998b] C. Rolland, C. Souveyet, C. Ben Achour. *Guiding Goal Modelling Using Scenarios*. Revue IEEE Transactions on Software Engineering (IEEE TSE), Special Issue on Scenario Management, Vol.24, No.12, pp 1055 - 1071, 1998.
- [Rolland1998c] C. Rolland, C. Ben Achour. *Guiding the Construction of Textual Use Case Specification*. Data & Knowledge Engineering Journal, (ed. P. Chen, R.P. van de Riet), Elsevier Science Publishers, North Holland, Vol.25, No.1, pp 125 – 160. 1998.
- [Rolland1999a] C. Rolland, G. Grosz, R. Kla. *Experience with Goal-Scenario Coupling in Requirements Engineering*. Actes de 4th IEEE International Symposium on Requirement Engineering (RE'99). Limerick, Irlande. Juin 1999.
- [Rolland1999b] C. Rolland, C. Souveyet, C. Ben Achour, Mustapha Tawbi, Fernando Velez. *Report on the CREWS-C2 Evaluations*. Rapport Interne N° 99-18, projet ESPRIT CREWS N°21.903. 1999.

- [Rolland2004] C. Rolland, C. Salinesi, A. Etien. *Eliciting Gaps in Requirements Change*. Requirements Engineering Journal , Vol. 9, No. 1, pp 1 – 15. 2004.
- [Rolland2005] C. Rolland, C. Salinesi. *Modeling Goals and Reasoning with Them*. Chapitre du livre « Engineering and Managing Software Requirements », A. Aurum, C. Wohlin (editeurs), Springer Verlag, 2005.
- [Rosemann2002] M. Rosemann, P. Green. *Developing a Meta Model for the Bunge-Wand-Weber Ontological Constructs*. Revue Information Systems, Vol. 27, N°2, pp 75-91. 2002
- [Ross1977] D.T. Ross, K.E. Schoman. *Structured Analysis for Requirements Definition*. Revue IEEE Transactions on Software Eng., Vol. 3, No. 1. 1977.
- [Royce1987] W.W. Royce. *Managing the Development of Large Software Systems: Concepts and Techniques*. Actes de 9th International Conference on Software Engineering (ICSE'87), pp 328- 338. Monterey, California, United States, 1987
- [Ruiz2004] F. Ruiz, A. Vizcaíno, M. Piattini. *An Ontology for the Management of Software Maintenance Projects*. International Journal Of Software Engineering and Knowledge Engineering, Vol. 14, No. 3, pp 323-349. 2004
- [Sabetzadeh2005] M. Sabetzadeh, S. Easterbrook. *An Algebraic Framework for Merging Incomplete and Inconsistent Views*. Actes de 13th IEEE International Conference on Requirements Engineering (RE'05), pp306-315. France, Septembre 2005.
- [Salinesi2002a] C. Salinesi, M. J. Presso. *A Method to Analyse Changes in the Realisation of Business Intentions and Strategies for Information System Adaptation*. Actes de IEEE International Enterprise Distributed Object Computing Conference (EDOC'02), Lausanne, Suisse, Septembre 2002.
- [Salinesi2002b] C. Salinesi, J. Wäyrynen. *A Methodological Framework for Understanding IS Adaptation through Enterprise Change*. Object Oriented Information Systems (OOIS'02), Montpellier, France, Septembre 2002.
- [Salinesi2003a] C. Salinesi C. Rolland. *Fitting Business Models to Systems Functionality Exploring the Fitness Relationship*. Actes de International Conference on Advanced information Systems Engineering (CAiSE'03). Klagenfurt, Velden, Autriche. Juin 2003.
- [Salinesi2003b] A. Etien, C. Salinesi. *Compliance Gaps: a Requirement Elicitation Approach in the Context of System Evolution*. Object Oriented Information Systems (OOIS'03). Genève, Suisse. Septembre 2003.
- [Salinesi2004a] C. Salinesi. *Authoring Use Cases*. Chapitre du livre « Scenarios, Stories, Use Cases Trough the Systems Lifecycle ». Edité par I. Alexander et N. A. M. Maiden. JohnWiley. Août 2004.
- [Salinesi2004b] C. Salinesi, A. Etien, I. Zoukar. *Goal / Strategy Maps - Methods, Techniques and Tools to Specify Requirements in Different Evolutionary Contexts*. International Conference on Systems Engineering (INCOSE'04). Toulouse, France, Juin 2004.

- [Salinesi2004c] C. Salinesi, A. Etien, I. Zoukar. *A Systematic Approach to Express IS Evolution Requirements Using Gap Modelling and Similarity Modelling Techniques*. International Conference on Advanced information Systems Engineering (CAISE'04). Riga Lituanie, Juin 2004.
- [Salinesi2004d] C. Salinesi, A. Etien, J. Wäyrynen. *Towards a Systematic Propagation of Evolution Requirements in IS Adaptation Projects*. Actes de Australian Conference on Information Systems (ACIS'04), Hobart, Australie, Décembre 2004.
- [Salinesi2005] C. Salinesi. *Dealing with the Information Systems Urbanisation Issues with a Requirements Engineering Approach*. Actes de Australian Workshop on Requirements Engineering (AWRE'05), Melbourne, Australie, Novembre 2005.
- [Salinesi2007] C. Salinesi. *Enterprise Architecture: from Practice Issues to Research Innovation*. Actes de International Conference on Research Challenges in Information Science (RCIS'07), (C. Rolland, O. Pastor, JL Cavarero, eds), pp. 107 – 120. Ourzazate, Maroc. Avril 2007.
- [Salinesi2008a] C. Salinesi, L. H. Thevenet. *Enterprise Architecture: des Problèmes Pratiques à l'Innovation*. Revue Ingénierie des Systèmes d'Information (ISI), RSTI (Revue des sciences et technologies de l'information) Numéro Spécial « Nouveaux challenges dans les SI », Hermès, France, 1:1, pp 75 – 105. 2008.
- [Salinesi2008a] C. Salinesi, E. Ivankina, W. Angole. *Using the RITA Threat Ontology to Guide Requirements Elicitation: an Empirical Investigation in the Bank Secto*. Actes du 1st International Workshop on Managing Requirements Knowledge (MaRK'08). Barcelone, Espagne. Septembre 2008.
- [Scacchi2006] W. Scacchi, J. Feller, B. Fitzgerald, S. Hissam, K. Lakhani. *Understanding Free/Open Source Software Development Processes*. Guest Editorial. Software Process Improvement and Practice No. 11, pp 95-105. 2006.
- [SCHEER] www.ids-scheer.fr
- [Seybold2004] C. Seybold, S. Meier, M. Glinz. *Evolution of Requirements Models by Simulation*. Actes de International Workshop on Principles of Software Evolution (IWPSE'04), pp43-48. Kyoto, Japon, Septembre 2004.
- [Singh2007] S. Singh, C. Woo. *A Methodology for Understanding Business-IT Alignment: The 3G Framework*. Actes de 6th SIGSAND Symposium on Research in Systems Analysis and Design. Tusla, USA, Mai 2007
- [Smaczny2000] T. Smaczny. *Is an Alignment Between Business and Information Technology the Appropriate Paradigm to manage IT in today's Organisations?*. Management Decision, Volume 39, Number 10, N0 6, pp 797-802. 2001.
- [Soffer2003] P. Soffer, B. Golany, D. Dori. *ERP Modelling: a Comprehensive Approach*. Revue Information Systems, No. 28, pp673-690. 2003.
- [Soffer2004] P. Soffer *Analyzing the Scope of a Change in a Business Process Model*. Actes de Workshop on Creating and Maintaining the Fit between Business Processes and Support Systems (BPMDS'04). Riga, Latvia. 2004.
- [Soffer2005] P. Soffer, B. Golany, D. Dori. *Aligning an ERP System with Enterprise Requirements: an Object Oriented Approach*. Revue Computers in Industry, Vol. 56, No. 6, pp639-662. Août 2005.

- [Sommerville2007] I. Sommerville. *Software Engineering*. Pearson Education, 2007
- [Spanoudakis1993] G. Spanoudakis, P. Constantopoulos. *Similarity for Analogical Software Package Reuse: a Computational Model*. Actes de International Conference on Advanced information Systems Engineering (CAiSE'93). Paris, France. Juin 1993.
- [Tanabe2008] D. Tanabe, K. Uno, K. Akemine, T. Yoshikawa, H. Kaiya, M. Saeki. *Supporting Requirements Change Management in Goal Oriented Analysis*. Actes de IEEE International Conférence on Requirements Engineering, Espagne. Juin 2008.
- [Tawbi12000] M. Tawbi, J. F. Velez, C. Ben Achour, C. Souveyet. Evaluating the CREWS-L'Écritoire Requirements Elicitation Process. Actes de Requirements Engineering: Foundation for Software Quality (REFSQ'00). Stockholm, Suède. Juin 2000.
- [Tawbi1999] M. Tawbi, C. Ben Achour, J. F. Velez. *Modélisation des Buts et Extraction des Besoins*. Journée Jeunes Chercheurs Ingénierie des Systèmes d'Information et Ingénierie des Connaissances (ISI&IC). France, 1999.
- [Teeuw1997] W.B. Teeuw, H. van den Berg. *On the Quality of Conceptual Models*. Actes de 16th International Conference on Conceptual Modeling ER'97. Los Angeles, Californie, USA, Novembre 1997.
- [Terrasse2003] M-N Terrasse, M. Savonnet, G. Becker, E. Leclerc. *UML-based Metamodeling for Information System Engineering and Evolution*. Actes de 9th International Conference on Object-Oriented Information Systems (OOIS'03), pp 83-94. Genève, Suisse. 2003.
- [Thevenet2006] L. H. Thevenet, C. Salinesi, A. Etien, I. Gam, M. Lassoued. *Experimenting a Modeling Approach for Designing Organization's Strategies in the Context of Strategic Alignment*. Actes de Australian Workshop on Requirements Engineering (AWRE), Adelaide, Australie. Décembre 2006.
- [Thevenet2007a] L. H. Thevenet, I. Gam, C. Salinesi. *Strategic Alignment Documentation*. Actes de IEEE International Conference on Research Challenges in Information Science (RCIS'07). Ouarzazate, Maroc. Avril 2007.
- [Thevenet2007b] L. H. Thevenet, C. Salinesi. *Aligning IS to Organization's Strategy: the INSTAL Method*. Actes de International Conference on Advanced information Systems Engineering (CAISE'07), pp 203 – 217. Trondheim, Norvège, Juin 2007.
- [Thevenet2007c] L. H. Thevenet, C. Salinesi. *Documenter l'alignement des objectifs stratégiques de l'entreprise et du SI avec la méthode INSTAL*. Atelier Ingénierie et gestion des processus d'entreprise (ECI), Paris, France. 2007.
- [Thevenet2008b] L. H. Thevenet. *Modeling Strategic Alignment Using INSTAL*. Actes de l'atelier Requirements, Intentions and Goals in Conceptual Modeling (RIGiM), en association avec ER'08, pp. 261 – 271. Barcelone, Espagne. Octobre 2008.
- [Thevenet2008c] L. H. Thevenet, C. Rolland, C. Salinesi. *Modeling Strategic Alignment with the INSTAL Method*. Revue INSIGHT (INSIGHT), INCOSE, 11:3, pp. 28 – 29. 2008.

- [Thevenet2009d] L.H. Thevenet. *Proposition d'une modélisation conceptuelle de l'alignement stratégique: La méthode INSTAL*. Thèse de Doctorat, Université de Paris 1 Panthéon - Sorbonne. Décembre 2009.
- [Thevenet2009a] L. H. Thevenet, C. Salinesi. *Proposition de mesure de l'alignement stratégique du système d'information dans la méthode INSTAL*. Ingénierie d'Entreprise et de Systèmes d'Information (IESI), Hermes Lavoisier, Toulouse, France, 2010. (à paraître)
- [Thevenet2009b] L. H. Thevenet, C. Rolland, C. Salinesi. *Alignement de la stratégie et du Système d'Information: Présentation de la méthode INSTAL*. Revue des sciences et technologies de l'information (RSTI) série Ingénierie des Systèmes d'Information (ISI), Hermès. Numéro spécial « Ingénierie des Evolutions », sous la direction de J. Akoka, I. Comyn-Wattiau, Vol. 14, No 6. 2009.
- [Thevenet2009c] L. H. Thevenet, C. Rolland, C. Salinesi. *Modeling Strategic Alignment with the INSTAL Method*. Systems Research Forum Journal (SRF), World Scientific Publishing Compagny, 3:1. 2009.
- [UML] <http://www.uml.org>
- [Viscusi2008] G. Viscusi, L. H. Thevenet, C. Salinesi. *Strategic Alignment in the Context of e-Services – an Empirical Investigation of the INSTAL Approach with the Italian e-Government Initiative Case Study*. Actes de International Conference on Advanced information Systems Engineering(CAISE'08), pp 163 – 166. Montpellier, France. Juin 2008.
- [Wand2004] Y. Wand, R. Weber. *Reflection: Ontology in Information Systems*. Journal of Database Management, Vol. 15, No. 2. 2004.
- [Wegman2005] A. Wegmann, G. Regev, B. Loison. *Business and IT Alignment with SEAM*. Actes de 1st International Workshop on Requirements Engineering for Business Need and IT Alignment (REBNITA'05), en association avec la conférence RE'05. Paris, France. Septembre 2005.
- [Wegman2007] A. Wegmann, P. Julia, G. Regev, O. Perroud, I. Rychkova. *Early Requirements and Business-IT Alignment with SEAM for Business*. Actes de 15th IEEE International Requirements Engineering Conference (RE'07). Dehli, Inde. Septembre 2007.
- [Weidenhaupt1998] Klaus Weidenhaupt, Klaus Pohl, Matthias Jarke, Peter Haumer, CREWS TEAM. *Scenario Usage in System Development: A Report on Current Practice*. Revue IEEE Software, Vol.15, No.2, pp 34-45. 1998
- [Wieringa2003] R.J. Wieringa, H.M. Blanken, M.M. Fokkinga, and P.W.P.J. Grefen. *Aligning Application Architecture to the Business Context*. Actes de International Conference on Advanced Information System Engineering (CAiSE 03), pages 209–225. Klagenfurt-Velden, Autriche. Juin 2003.
- [Yu1994] E. Yu, J. Mylopoulos. *Using Goals, Rules and Methods to Support Reasoning in Business Process Reengineering*. Actes de 27th Hawaii International Conference System Sciences (HICSS'94), Janvier 4-7, Vol. 4 pp 234-243. Maui, Hawaii. 1994.

- [Yu97] E. Yu. *Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering*. Actes de 3rd IEEE International Symposium on Requirements Engineering (RE'97), pp 226-235. Washington D.C., USA. Janvier 1997,
- [Yue1987] K. Yue. *What Does It Mean to Say that a Specification is Complete?*. Actes de 4th International IEEE Workshop on Software Specification & Design (IWSSD-4). USA. 1987.
- [Zaidman] A. Zaidman, B. Van Rompaey, S. Demeyer, A. Van Deursen. *Mining Software Repositories to Study Co-Evolution of Production & Test Code*. Actes de International Conference on Software Testing, Verification, and Validation (ICST'08) pp220-229. Lillehammer, Norvège. Avril 2008
- [Zave1997] P. Zave, M. Jackson. *Four Dark Corners of Requirements Engineering*. Revue ACM Transactions on Software Engineering and Methodology, Vol. 6, No. 1, pp 1 - 30. Janvier 1997.
- [Zhang2004] W. Zhang, H. Zhao, and H. Mei. *A Propositional Logic-based Method for Verification of Feature Models*. Actes de 10th International Conference on Formal Engineering Methods (ICFEM'08) Vol. 3308, pages 115–130. Springer–Verlag. Kitakyushu-city, Japon. Octobre 2004.
- [Zicari92] R. Zicari. *A Framework for Schema Updates in an Object-Oriented Database System*. Chapitre 7 du livre Building an Object-Oriented Database System—the Story of O2 (F. Bancilhon, C. Delobel, and P. Kanellakis, eds). Morgan Kaufmann Publishers. 1992.
- [Zoukar2004a] I. Zoukar, C. Salinesi. *Matching ERP Functionalities with the Logistic Requirements of French railways - A Similarity Approach*. International Conference on Enterprise Information Systems (ICEIS'04). Porto, Portugal. Juin 2004.
- [Zoukar2004b] I. Zoukar, C. Salinesi, C. Rolland. *Evolution du Système d'Information par l'Implantation d'un Progiciel de Gestion Intégrée PGI - Systématiser la Mise en coïncidence entre le Système et l'Organisation*. Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'04). Biarritz, France. Juin 2004.
- [Zoukar2004c] I. Zoukar, C. Salinesi. *Using Goal / Strategy Maps to Reduce the Language Disparity Issue in ERP Projects*. Agent-Oriented Information Systems (AOIS'04). Riga, Lituanie. Juin 2004.
- [Zoukar2004d] I. Zoukar, C. Salinesi. *Engineering the Fitness Relationship between an ERP and the Supply Chain Process at SNCF*. Information Resources Management Association (IRMA), Business Process Management Tools and Technologies track. Nouvelle Orléans, USA. Mai 2004.
- [Zowghi1996] D. Zowghi, A.K. Ghose, P. Peppas *A Framework for Reasoning about Requirements Evolution*. Actes de 4th Pacific Rim International Conference on Artificial Intelligence (PRICAI'96), Cairns, Australia. Août 1996
- [Zowghi2003] D. Zowghi and V. Gervasi. *On the Interplay Between Consistency, Completeness, and Correctness in Requirements Evolution*. Information and Software Technology, 45, 14 , pp 993-1009. Novembre 2003.

Bibliographie

1. Métriques bibliographiques

Hindex.....	11
Gindex.....	29
Nombre de chapitres de livres.....	3
Nombre d'articles de revue.....	18
Nombre d'articles de conférences internationales	35
Nombre d'articles de conférences nationales.....	3
Autres publications (ateliers internationaux, ateliers nationaux, forums).....	40

2. Principales références

- C. Rolland, C. Souveyet, C. Ben Achour. *Guiding Goal Modelling Using Scenarios*. IEEE Transactions on Software Engineering (IEEE TSE), Special Issue on Scenario Management, 24:12, pp. 1055 - 1071, 1998. (302 références)
- C. Rolland, C. Ben Achour, C. Cauvet, J. Ralyte, A. Sutcliffe, N. M. Maiden, M. Jarke, P. Haumer, K. Pohl, E. Dubois, P. Heymans. *A Proposal for a Scenario Classification Framework*. Requirements Engineering Journal (REJ), 3:1, 1998. (137 références)
- C. Rolland, C. Ben Achour. *Guiding the Construction of Textual Use Case Specification*, Data & Knowledge Engineering Journal (DKJ), (ed. P. Chen, R.P. van de Riet), Elsevier Science Publishers, North Holland, 25:1, pp. 125 - 160, 1998. (109 références)
- C. Ben Achour, C. Rolland, N. M. Maiden, C. Souveyet. *Use Case Authoring: Results of an Empirical Study*. International Symposium on Requirements Engineering (ISRE), Limerick, Ireland, June 1999. (53 références)
- G. Grosz, C. Rolland, S. Schwer, C. Souveyet, V. Plihon, S. Si-Said, C. Ben Achour, C. Gnaho. *Modelling and Engineering the Requirements Engineering Process: An Overview of the NATURE Approach*. Requirements Engineering Journal (REJ), 2, pp. 115 - 131, 1997. (39 références)
- C. Rolland, C. Salinesi, A. Etien. *Eliciting Gaps in Requirements Change*. Requirements Engineering Journal (REJ), 9:1, pp. 1 - 15, 2004. (34 références)
- C. Ben Achour. *Guiding Scenario Authoring*. European-Japanese Conference on Information Modelling and Knowledge Bases (EJC), Finland, May 1998. (28 références)

- C. Salinesi, C. Rolland. *Fitting Business Models to Software Functionality: Exploring the Fitness Relationship*. International Conference on Advanced information Systems Engineering (CAISE), Springer Verlag, 2003. (27 références)
- M. Jarke, K. Pohl, P. Haumer, K. Weidenhaupt, E. Dubois, P. Heymans, C. Rolland, C. Ben Achour, C. Cauvet, J. Ralyté, A. Sutcliffe, N.A.M. Maiden, S. Minocha. *Scenario Use in European Software Organizations – Results from Site Visits and Questionnaires*. Rapport CREWS 97-10. 1997. (18 références). Synthèse publiée sous le titre: *Scenario Usage in System Development: a Report on Current Practice*. Klaus Weidenhaupt, Klaus Pohl, Matthias Jarke, Peter Haumer, CREWS Team. Proceedings of ICRE'98, 3rd International Conference on Requirements Engineering, Colorado Springs USA. 6-10 April 1998. (91 références)
- C. Ben Achour, C. Souveyet, M. Tawbi. *Bridging the Gap between Users and Requirements Engineering: the Scenario-Based Approach*. Journal of Computer Systems Science & Engineering (JCSSE), Special issue: Object-Oriented Information Systems 1999, 14:6, pp. 112 - 122, 1999. (12 références)
- C. Ben Achour. *Linguistic Instruments for the Integration of Scenarios in Requirement Engineering*. Requirements Engineering: Foundation for Software Quality (REFSQ), Barcelona, Catalonia, Spain, June 1997. (12 références)
- C. Salinesi, J. Wärynen. *A Methodological Framework for Understanding IS Adaptation through Enterprise Change*. Object Oriented Information Systems (OOIS), Montpellier, France, September 2002. (11 références)
- C. Salinesi, M. J. Presso. *A Method to Analyse Changes in the Realisation of Business Intentions and Strategies for Information System Adaptation*. International Enterprise Distributed Object Computing Conference (EDOC), Lausanne, Switzerland, September 2002. (10 références)
- C. Rolland, and C. Salinesi. *Modeling Goals and Reasoning with Them*. « Engineering and Managing Software Requirements », ouvrage collectif, A. Aurum, C. Wohlin (éditeurs), Springer Verlag, 2005. (10 références)
- C. Salinesi, A. Etien, I. Zoukar. *A Systematic Approach to Express IS Evolution Requirements Using Gap Modelling and Similarity Modelling Techniques*. International Conference on Advanced information Systems Engineering (CAISE), Springer Verlag, Riga, Latvia, 2004. (8 références)

3. Chapitres de livres

- C. Rolland, C. Salinesi. *Supporting Requirements Elicitation through Goal/Scenario Coupling*. Dans « Conceptual Modeling : Foundations and Applications », A.T. Borgida, V.K. Chaudhri, R. S. Yu (éditeur), Springer. 2009.
- C. Rolland, C. Salinesi. *Modeling Goals and Reasoning with Them*. Dans « Engineering and Managing Software Requirements », A. Aurum, C. Wohlin (éditeurs), Springer Verlag. 2005.
- C. Salinesi. *Authoring Use Cases*. Dans « Scenarios & Use Cases, Stories through the System Life-Cycle, », Ian Alexander, Neil Maiden (éditeurs), John Wiley and Sons. 2004.

4. Articles de revue

- L. H. Thevenet, C. Rolland, C. Salinesi. *Alignement de la stratégie et du Système d'Information: Présentation de la méthode INSTAL*. Revue des sciences et technologies de l'information (RSTI) série Ingénierie des Systèmes d'Information (ISI), Hermès. Numéro spécial « Ingénierie des Evolutions », sous la direction de J. Akoka, I. Comyn-Wattiau, Vol. 14, No 6. 2009.
- E. Kornysheva, C. Salinesi, R. Deneckère. *Which Method to Support Multicriteria Decision Making Systematically in IS Engineering?*. Systems Research Forum Journal, Vol.3, No 1, pp. 15-24. 2009.
- L. H. Thevenet, C. Rolland, C. Salinesi. *Modeling Strategic Alignment with the INSTAL Method*. Systems Research Forum Journal, Vol.3, No 1, pp. 39-49. 2009.
- O. Djebbi, C. Salinesi, C. Rolland. *Product Line Requirements Configuration in the Context of Multiple Models*. Revue INSIGHT, INCOSE. Vol. 11, No 3, pp. 19 – 20. 2008.
- L. H. Thevenet, C. Rolland, C. Salinesi. *Modeling Strategic Alignment with the INSTAL Method*. Revue INSIGHT, INCOSE, Vol. 11, No 3, pp. 28 – 29. 2008.
- E. Kornysheva, C. Salinesi. *Introducing Multicriteria Decision Making into Software Engineering*. Revue INSIGHT, INCOSE, Vol. 11, No 3, p 24. 2008.
- C. Salinesi, L. H. Thevenet. *Enterprise Architecture: des problèmes pratiques à l'innovation*. Revue des sciences et technologies de l'information (RSTI) série Ingénierie des Systèmes d'Information (ISI), Hermès. Numéro spécial « Nouveaux challenges dans les SI », sous la direction de C. Rolland, O. Pastor, J.L. Cavarero, Vol. 13, No 1, pp. 75 – 105. 2008.
- I. Gam, L. H. Thevenet, C. Salinesi. *Documenter l'Alignement d'un Entrepôt de Données avec la Stratégie d'Entreprise pour mieux satisfaire les Exigences des Décideurs*. Revue des sciences et technologies de l'information (RSTI) série Ingénierie des Systèmes d'Information (ISI), Hermès. Numéro spécial « Elaboration des Entrepôts de Données », sous la direction de E. Métais, G. Zurfluch, Vol.11, No6, pp. 183-114. 2006.
- E. Papadacci Stephanopoli, C. Salinesi, L. Sidler. *Panorama des approches d'arbitrage dans le contexte de l'urbanisation du SI. État de l'art et mise en perspective des approches issues du monde de l'ingénierie des exigences*. Revue des sciences et technologies de l'information (RSTI) série Ingénierie des Systèmes d'Information (ISI), Hermès. Numéro spécial « Méthodes Avancées de Développement dans les SI », sous la direction de J.P. Giraudin, D. Rieu, Vol. 10, No 6, pp. 11 – 30. 2005.
- C. Rolland, C. Salinesi, A. Etien. *Eliciting Gaps in Requirements Change*. Requirements Engineering Journal (REJ), Springer. Vol. 9, No1, pp. 1 – 15. 2004.
- C. Salinesi, B. Regnell. *Editorial - Eighth International Workshop on Requirements Engineering: Foundation for Software Quality*. Revue Information & Software Technology (IST). Numéro spécial « Special Issue on REFSQ », Elsevier Science, Vol. 45, No 14, pp. 941 – 943. 2003.

- C. Salinesi, A. Opdhal, M. Rossi. *REFSQ '2001 workshop summary: Seventh International Workshop on Requirements Engineering: Foundations for Software Quality*. Software Engineering Notes (ACM SIGSOFT), Vol. 27, No2. 2002.
- C. Salinesi, A. Opdhal, M. Rossi. *Editorial - Seventh International Workshop on Requirements Engineering: Foundations for Software Quality (REFSQ'01)*. Requirements Engineering Journal (REJ). Numéro spécial « Special Issue on REFSQ », Vol. 7, No 1. 2002.
- C. Ben Achour, C. Souveyet, M. Tawbi. *Bridging the Gap between Users and Requirements Engineering: the Scenario-Based Approach*. Journal of Computer Systems Science & Engineering (JCSSE). Numéro spécial « Object-Oriented Information Systems 1999 », Vol. 14, No 6, pp. 112 – 122. 1999.
- C. Rolland, C. Souveyet, C. Ben Achour. *Guiding Goal Modelling Using Scenarios*. Revue IEEE Transactions on Software Engineering (IEEE TSE). Numéro spécial « Special Issue on Scenario Management », Vol. 24, No 12, pp. 1055 – 1071. 1998.
- C. Rolland, C. Ben Achour. *Guiding the Construction of Textual Use Case Specification*. Data & Knowledge Engineering Journal (DKJ), P. Chen, R.P. van de Riet (éditeurs), Elsevier Science Publishers, North Holland, Vol. 25, No1, pp. 125 – 160. 1998.
- C. Rolland, C. Ben Achour, C. Cauvet, J. Ralyte, A. Sutcliffe, N. M. Maiden, M. Jarke, P. Haumer, K. Pohl, E. Dubois, P. Heymans. *A Proposal for a Scenario Classification Framework*. Requirements Engineering Journal (REJ), Springer, Vol. 3, No 1. 1998.
- G. Grosz, C. Rolland, S. Schwer, C. Souveyet, V. Plihon, S. Si-Said, C. Ben Achour, C. Gnaho. *Modelling and Engineering the Requirements Engineering Process: An Overview of the NATURE Approach*. Requirements Engineering Journal (REJ), Springer, Vol. 2, No3, pp. 115 – 131. 1997.

5. Articles publiés dans des conférences internationales avec comité de lectures et actes

- G. Fanmuy, J.C. Roussel, R. Szczepaniak, C. Salinesi, A. Dauron, L. Picci, O. Hammami. *Requirements Analysis and Modeling Process (RAMP) for the Development of Complex System*. Actes de European Conference on Systems Engineering and Innovation (EuSEC'10). Stockholm, Suède. Mai 2010.
- C. Salinesi, I. Gam. *How Specific should Requirements Engineering be in the Context of Decision Information Systems?*. Actes de 3rd IEEE International Conference of Research Challenges n Information Science (RCIS'09) . Fes, Maroc. Avril 2009.
- Clotilde Rohleder, Christoph Marhold, Camille Salinesi, Jörg Dörr. *Quality Data Model and Quality Control in the Product Lifecycle Management*. Actes de 6th International Conference on Product Lifecycle Management (PLM'09). Bath, Royaume Uni. Juillet 2009.

- C. Salinesi, C. Marhold, C. Rohleder, J. Doerr. *Clarifying Non-Functional Requirements to Improve User Acceptance – Experience at Siemens*. Actes de la Working Conference Requirements Engineering: Foundations of Software Quality (REFSQ'09), Springer LNCS. Amsterdam, Pays Bas. Juin 2009.
- O. Djebbi, C. Salinesi, D. Diaz. *Deriving Product Line Requirements: the RED-PL Guidance Approach*. Actes de Asian Pacific Software Engineering Conference (APSEC'07), IEEE Computer Society. Nagoya, Japon. Décembre 2007.
- C. Salinesi, Ramzi Bouzidif, Esther Elfassy. *An Experience of Reuse Based Requirements in ERP projects*. Actes de 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC'07). Annapolis, USA. Octobre 2007.
- O. Djebbi, C. Salinesi, G. Fanmuy. *Industry Survey of Product Lines Management Tools: Requirements, Qualities and Open Issues*. Actes de 15th IEEE International Conference on Requirement Engineering (RE'07). New Delhi, Inde. Octobre 2007.
- E. Kornysheva, R. Deneckère, C. Salinesi. *Method Chunks Selection by Multicriteria Techniques: an Extension of the Assembly-based Approach*. Actes de IFIP WG8.1 Working Conference Situational Method Engineering: Fundamentals and Experiences (ME'07), Springer. Genève, Suisse. Septembre 2007.
- O. Djebbi, C. Salinesi. *RED-PL, a Method for Deriving Product Requirements from a Product Line Requirements Model*. Actes de International Conference on Advanced information Systems Engineering (CAISE'07), Springer Verlag. Trondheim, Norvège. Juin 2007.
- L. H. Thevenet, C. Salinesi. *Aligning IS to organization's strategy: the INSTAL method*. Actes de International Conference on Advanced information Systems Engineering (CAISE'07), Springer Verlag. pp. 203 - 217 Trondheim, Norvège. Juin 2007.
- E. Kornysheva, C. Salinesi. *Business Process Priorisation with Multicriteria Methods: Case of Business Process Reengineering*. Actes de 9th International Conference on Enterprise Information Systems (ICEIS'07), Springer. Madeira, Portugal, Juin 2007.
- L. H. Thevenet, I. Gam, C. Salinesi. *Strategic Alignment Documentation*. Actes de IEEE International Conference on Research Challenges in Information Science (RCIS'07). Ouarzazate, Maroc. Avril 2007.
- C. Salinesi. *Enterprise Architecture: from Practice Issues to Research Innovation*. Research Actes de IEEE International Conference on Research Challenges in Information Science (RCIS'07), pp. 107 – 120. Ouarzazate, Maroc. Avril 2007
- E. Kornysheva, C. Salinesi. *MCDM Techniques Selection Approaches: State of the Art*. Actes de IEEE Symposium on Computational Intelligence in Multicriteria Decision Making (MCDM). Honolulu, USA. Avril 2007.
- A. Etien, C. Rolland, C. Salinesi. *A Meta-modelling Approach to Express Change Requirements*. Actes de International Conference on Software Engineering and Data Technologies (ICSOFT'06), Special session on Meta-modelling. Setubal, Portugal. Septembre 2006.

- I. Gam, C. Salinesi. *A Requirement-driven Approach for Designing Data Warehouses*. Actes de 12th Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'06). Luxembourg, Luxembourg. Juin 2006.
- E. Papadacci Stephanopoli, C. Salinesi, C. Rolland. *NENO process: Information systems arbitration process in Enterprise Architecture Project*. Actes de Information and Communication Technologies: from Theory to Applications (ICTTA), IEEE, pp. 105 – 106. Damascus, Syrie. Avril 2006.
- A. Etien, C. Salinesi. *Managing Requirements in a Co-evolution Context*. Actes de 13th IEEE International Conference on Requirement Engineering (RE'05), IEEE Computer Society Press, pp. 125 – 134. Paris, France. Septembre 2005.
- E. Ivankina, C. Salinesi. *Eliciting Requirements by Analysing Threats Caused by Users*. Actes de 3rd ACIS International Conference on Software Engineering Research, Management and Applications (SERA'05), IEEE Computer Society Press. Mont. Plaisance, USA. Août 2005.
- C. Salinesi, A. Etien, J. Wäyrynen. *Towards a Systematic Propagation of Evolution Requirements in IS Adaptation Projects*. Actes de Australian Conference on Information Systems (ACIS'04). Hobart, Australia. Decembre 2004.
- C. Salinesi, R. Kla. *Word Lexicon Generation from Textual Scenario Analysis*. Actes de 5th International Conference on Computer, Communication and Control Technologies (CCCT). Austin-Texas, USA. Août 2004.
- A. Etien, C. Salinesi. *Evolution Patterns to Elicit Compliance Requirements*. Actes de 14th European-Japanese Conference on Information Modelling and Knowledge Bases (EuroJap'04), Frontiers in Artificial Intelligence and Applications. Skövde, Suède. Juin 2004.
- E. Ivankina, C. Salinesi. *An Approach to Guide Requirement Elicitation by Analysing the Causes and Consequences of Threats*. Actes de 14th European-Japanese Conference on Information Modelling and Knowledge Bases (EJC'04), IOS Press. Skövde, Suède. Juin 2004.
- C. Salinesi, A. Etien, I. Zoukar. *Goal / Strategy Maps - Methods, Techniques and Tools to Specify Requirements in Different Evolutionary Contexts*. Actes de International Conference on Systems Engineering (INCOSE'04). Toulouse, France. Juin 2004.
- I. Zoukar, C. Salinesi. *Engineering the Fitness Relationship between an ERP and the Supply Chain Process at SNCF*. Actes de Information Resources Management Association (IRMA'04), Business Process Management Tools and Technologies track. New Orleans, USA. Mai 2004.
- E. Ivankina, C. Salinesi. *Adapting Ishikawa's Cause-Effect Diagrams to Understand Requirements Threats*. Actes de 7th International Conference on Business Information Systems (BIS'04). Poznan, Pologne. Avril 2004.
- C. Salinesi, A. Etien, I. Zoukar. *A Systematic Approach to Express IS Evolution Requirements Using Gap Modelling and Similarity Modelling Techniques*. Actes de 16th International Conference on Advanced information Systems Engineering (CAISE'04), Springer Verlag. Riga, Lituanie. Juin 2004.

- I. Zoukar, C. Salinesi. *Matching ERP Functionalities with the Logistic Requirements of French railways - A Similarity Approach*. Actes de 6th International Conference on Enterprise Information Systems (ICEIS'04). Porto, Portugal. Avril 2004.
- C. Salinesi, A. Etien. *Compliance Gaps: a Requirement Elicitation Approach in the Context of System Evolution*. Actes de Object Oriented Information Systems (OOIS'03), Springer. Genève, Suisse. Septembre 2003.
- C. Salinesi, C. Rolland. *Fitting Business Models to Software Functionality: Exploring the Fitness Relationship*. Actes de 16th International Conference on Advanced information Systems Engineering (CAISE'04), Springer Verlag. Klagenfurt/Velden. Juin 2003.
- C. Salinesi, J. Wäyrynen. *A Methodological Framework for Understanding IS Adaptation through Enterprise Change*. Actes de Object Oriented Information Systems (OOIS'02), Springer. Montpellier, France. Septembre 2002.
- C. Salinesi, M. J. Presso. *A Method to Analyse Changes in the Realisation of Business Intentions and Strategies for Information System Adaptation*. Actes de 6th IEEE International Enterprise Distributed Object Computing Conference (EDOC'02). Lausanne, Suisse. Septembre 2002.
- C. Salinesi, R. Kla. *Word Lexicon Generation from Textual Scenario Analysis*. Actes de 6th World Multiconference on Systemics Cybernetics and Informatics (SCI). Orlando, USA. Juillet 2002.
- C. Ben Achour, C. Rolland, N. M. Maiden, C. Souveyet. *Use Case Authoring: Results of an Empirical Study*. Actes de 4th IEEE International Symposium on Requirements Engineering (ISRE). Limerick, Ireland. Juin 1999.
- C. Ben Achour. *Guiding Scenario Authoring*. Actes de 17th European-Japanese Conference on Information Modelling and Knowledge Bases (EJC'98), IOS Press. Jyväskylä, Finland. Mai 1998.

6. Articles publiés dans des conférences nationales avec comités de lecture et actes

- I. Gam, C. Salinesi. *Analyse des exigences pour la conception d'entrepôts de données*. Actes de Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'06), Hammamet, Tunisie, pp. 1023 – 1038. Juin 2006.
- I. Zoukar, C. Salinesi, C. Rolland. *Evolution du Système d'Information par l'implantation d'un Progiciel de Gestion Intégrée - Systématiser la Mise en Correspondance entre le Système et l'Organisation*. Actes de Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'04). Casino Bellevue, Biarritz, France. Juin 2004.
- S. Assar, C. Ben Achour, S. Si-Said. *Un Modèle pour la Spécification des Processus d'Analyse des Systèmes d'Information*. Informatique des Organisations et Systèmes d'Information et de Décision (INFORSID'00). Lyon, France. Mai 2000.

7. Articles publiés lors d'ateliers et forums internationaux avec actes

- Alberto Lora-Michiels, Camille Salinesi and Raul Mazo Peña. *A Method based on Association Rules to Construct Product Line Model*. Actes de 4th International Workshop on Variability Modelling of Software-intensive Systems (VAMOS'10) "Celebrating 20 Years of Feature Models". Linz, Autriche. Janvier 2010.
- Camille Salinesi, Daniel Diaz, Olfa Djebbi, Raul Mazo. *Exploiting the Versatility of Constraint Programming over Finite Domains to Integrate Product Line Models*. Poster. Actes de 17th IEEE International Conference on Requirements Engineering (RE'09) . Atlanta, Georgia, USA. Septembre 2009.
- C. Salinesi, C. Rolland, R. Mazo. *VMWare: Tool Support for Automatic Verification of Structural and Semantic Correctness in Product Line Models*. Actes de 2nd International Workshop on Variability Modelling of Software-intensive Systems (VaMoS'09), pp. 173 – 176. Seville, Espagne. Janvier 2009.
- C. Salinesi, E. Ivankina, W. Angole. *Using the RITA Threat Ontology to Guide Requirements Elicitation: an Empirical Investigation in the Bank Sector*. Actes de 1st International Workshop on Managing Requirements Knowledge (MaRK'08) associé à RE'08. Barcelone, Espagne. Septembre 2008.
- E. Kornysheva, R. Deneckère, C. Salinesi. *Improving Software Development Processes with Multicriteria Methods*. Actes de Model Driven Information Systems Engineering: Enterprise, User and System Models (MoDISE-EUS) associé à CAiSE'08. Montpellier, France. Juin 2008.
- E. Kornysheva, R. Deneckère, C. Salinesi. *Using Multicriteria Decision-Making to Take into Account the Situation in System Engineering*. Actes de Forum of International Conference on Advanced information Systems Engineering (CAISE'08 Forum). Montpellier, France. Juin 2008.
- G. Viscusi, L. H. Thevenet, C. Salinesi. *Strategic Alignment in the Context of e-Services – an Empirical Investigation of the INSTAL Approach with the Italian e-Government Initiative Case Study*. Article court. Actes de 20th International Conference on Advanced information Systems Engineering (CAISE'08), pp. 163 – 166. Montpellier, France. Juin 2008.
- O. Djebbi, C. Salinesi. *Towards an Automatic PL Requirements Configuration through Constraints Reasoning*. Actes de 2nd International Workshop on Variability Modelling of Software-intensive Systems (VaMoS'08). Essen, Allemagne. Janvier 2008.
- L. H. Thevenet, C. Salinesi, A. Etien, I. Gam, M. Lassoued. *Experimenting a Modeling Approach for Designing Organization's Strategies in the Context of Strategic Alignment*. Actes de 11th Australian Workshop on Requirements Engineering (AWRE'06), Adelaide, Australie. Decembre 2006.
- O. Djebbi, C. Salinesi. *Criteria for Comparing Requirements Variability Modeling Notations for Product Lines*. Actes de 4th International Workshop on Comparative Evaluation in Requirements Engineering (CERE'06), associé à RE'06, pp. 20 – 35. Minneapolis, USA. Septembre 2006.

- C. Salinesi, E. Kornysheva. *Choosing a Prioritization Method – Case of IS Security Improvement*. Actes du Forum of the International Conference on Advanced information Systems Engineering (CAISE'06 forum), pp. 51 - 55. Luxembourg, Luxembourg. Juin 2006.
- C. Salinesi. *Dealing with the Information Systems Urbanisation Issues with a Requirements Engineering Approach*. Actes de 10th Australian Workshop on Requirements Engineering (AWRE'05). Melbourne, Australie. Novembre 2005.
- L. H. Thevenet, G. Fanmuy, C. Salinesi. *Sharing Methodological Knowledge with REGAL*. Poster. Actes de 13th IEEE International Conference on Requirement Engineering (RE'05), IEEE Computer Society Press, pp. 461-462. Paris, France. Septembre 2005.
- E. Papadacci Stephanopoli, C. Salinesi. *Arbitrating between Enterprise Objectives. Neno – an Enterprise Architecture Approach developed at Renault*. Actes de l'atelier Requirements Engineering Decision Support (REDECS'05), associé à RE'05. Paris, France. Août 2005.
- A. Etien, C. Rolland, C. Salinesi. *Measuring the Business / System Alignment*. Actes de 1st International Workshop on Requirements Engineering for Business Need and IT Alignment (REBNITA'05), Associé à RE'05. Paris, France. Août 2005.
- A. Etien, C. Rolland, C. Salinesi. *Overview of a Gap-driven Evolution Process*. Actes de 9th Australian Workshop on Requirements Engineering (AWRE), Adelaide, Australia, December 2004.
- I. Zoukar, C. Salinesi. *Using goal / strategy maps to reduce the language disparity issue in ERP projects*. Actes de l'atelier Agent-Oriented Information Systems (AOIS), Riga, Latvia, June 2004.
- C. Salinesi, E. Ivankina. *Guiding Use Case Discovery By Analysing Threats*. Actes de Forum of the International Conference on Advanced information Systems Engineering (CAISE'04 forum), Springer Verlag. Riga, Latvia. Juin 2004.
- V. Gervasi, E. Kamsties, B. Regnell, C. Salinesi. *Ten Years of REFSQ: a Quantitative Analysis*. Actes de 10th International workshop on Requirements Engineering:Foundation for Software Quality (REFSQ'04), associé à CAiSE'04. Riga, Lituanie. Juin2004.
- E. Papadacci Stephanopoli, C. Salinesi, C. Rolland. *Payoff Analysis of Business Systems in Goal-Oriented Requirements Engineering*. Actes de 10th International workshop on Requirements Engineering:Foundation for Software Quality (REFSQ'04). Riga, Latvia. Juin 2004.
- A. Etien, R. Deneckère, C. Salinesi. *Extending Methods to Express Change Requirements*. Actes de Engineering Methods to Support Information Systems Evolution (EMSISE'03), en association avec OOIS'03. Genève, Suisse. Septembre 2003.
- A. Etien, C. Salinesi. *Towards a Systematic Definition of Requirements for Software Evolution: A Case-study Driven Investigation*. Actes de Evaluation of Modeling Methods in Systems Analysis and Design (EMMSAD'03), en association avec CAISE'03. Klagenfurt Velden, Autriche. Juin2003.

- M. Frendi, C. Salinesi. *Requirements Engineering for Data Warehousing*. Actes de 9th International workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'03), en association avec CAISE'03. Klagenfurt Velden, Autriche. Juin 2003.
- C. Souveyet, C. Salinesi. *Generating Lyee Programs from User Requirements with a Meta-Model based Methodology*. Actes de International Workshop on Lyee Methodology (SoMeT), pp. 212 – 224. Paris, France. 2002.
- C. Ben Achour, C. Ncube. *Engineering the PORE Method for COTS Selection and Implementation with the MAP Process Meta-Model*. Actes de 6th International workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'00), en association avec CAiSE'00. Stockholm, Suède. Juin 2000.
- M. Tawbi, J. F. Velez, C. Ben Achour, C. Souveyet. *Evaluating the CREWS-L'Ecritoire Requirements Elicitation Process*. Actes de 6th International workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'00), en association avec CAiSE'00. Stockholm, Sweden. Juin 2000.
- M. Tawbi, C. Ben Achour, J. F. Velez. *Guiding the process of requirements elicitation through scenario analysis: Results of an empirical study*. Actes de l'atelier Requirements Engineering Process (REP), pp. 345 - 352, en association avec DEXA'00. Florence, Italie. Septembre 1999.
- C. Ben Achour. *Writing and Correcting Textual Scenarios for System Design*. Actes de l'atelier Natural Language and Information Systems (NLIS), Vienna, Autriche. Août 1998.
- C. Ben Achour, C. Rolland, C. Souveyet. *A Proposal for Improving the Quality of the Organisation of Scenario Collections*. Actes de 5th International workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'98), en association avec CAiSE'98. Pise, Italie. Juin 1998.
- J. Ralyte, C. Ben Achour. *Scenario Integration into Requirements Engineering Methods*. Actes de l'atelier Many Facets of Process Engineering (MFPE). Grammarth, Tunisie. Septembre 1997.
- C. Ben Achour. *Linguistic Instruments for the Integration of Scenarios in Requirement Engineering*. Actes de 4th International workshop on Requirements Engineering: Foundation for Software Quality (REFSQ'97), en association avec CAiSE'97. Barcelona, Catalonia, Espagne. Juin 1997.

8. Articles présentés dans le cadre d'ateliers nationaux avec ou sans actes

- Alberto Lora-Michiels. Camille Salinesi. Raúl Mazo. *The Baxter Return of Experience on the Use of Association Rules to Construct its Product Line Model*. Journée SPL, Lignes de produits logiciels et usines logicielles. Nantes. Octobre 2009.
- C. Salinesi, D. Diaz, R. Mazo, O. Djebbi. *Spécification d'Exigences dans le Contexte de Lignes de Produits*. Journée Action IDM - INFORSID « Exigence, Traçabilité et Co-conception dans les processus de développement ». Paris. Octobre 2009.

- L. H. Thevenet, C. Salinesi. *Proposition de Mesure de l'Alignement Stratégique du Système d'Information dans la Méthode INSTAL*. Actes de l'Atelier Ingénierie d'Entreprise et de Systèmes d'Information (IESI), en association avec la conférence INFORSID'09. Toulouse, Mai 2009.
- E. Kornyshova, R. Deneckère, C. Salinesi. *Improving Software Development Processes with Multicriteria Methods*. Méthodes Avancées de Développement des Systèmes d'Information (MADSI), Fontainebleau, Septembre 2008.
- L. H. Thevenet, C. Salinesi. *Documenter l'alignement des objectifs stratégiques de l'entreprise et du SI avec la méthode INSTAL*. Ingénierie et gestion des processus d'entreprise (ECI), Paris, FRANCE, 2007.
- M. Lassoued, C. Salinesi. *Analyse des besoins de sécurité pour les Systèmes d'Information Pair-à-Pair*. Gestion de données dans les Systèmes d'Information pervasifs (GEDSIP), En association avec INFORSID, Perros-Guirec, France, 2007.
- E. Papadacci Stephanopoli, C. Salinesi. *NENO: Une approche d'aide à l'arbitrage par l'évaluation qualitative et quantitative de la valeur métier de SI durant la phase amont du processus d'urbanisation*. Ingénierie et gestion des processus d'entreprise (ECI), à l'initiative du GT « Entreprise Communicante et Interopérabilité » CNRS GDR I3 et GDR MACS, Paris, France, 2005.
- C. Salinesi. *Cartographie du Système d'Information par les Intentions et Stratégies Métiers*. Colloque Urbanisme des Systèmes d'Information (Urba SI), I. Contini, Paris, Sorbonne, Paris, 2002.
- M. Tawbi, C. Ben Achour, J. F. Velez. *Modélisation des buts et extraction des besoins*. Journée Jeunes Chercheurs Ingénierie des Systèmes d'Information et Ingénierie des Connaissances (ISI&IC), Paris, France, 1999.

9. Rapports internes et livrables de projets

- C. Salinesi. *A Method to Analyse Changes in the Realisation of Business Intentions and Strategies for Information System Adaptation*. Centre de Recherche en Informatique (CRI), Rapport de Fin de Projet, Collaboration FT R&D - CRI, N° 01 1B A30, 2002.
- C. Salinesi, R. Kla. *L'Ecritoire Linguistic Approach: Concept Definition and Implementation*. Centre de Recherche en Informatique (CRI), Projet E-Lyee, Rapport Interne, 2002.
- C. Salinesi. *Changement des Intentions, Stratégies et Processus Métiers: Cas d'Etude du Financement Unifié*. Centre de Recherche en Informatique (CRI), Livrable Projet FT R&D N° 01 1B A30, 2002.
- C. Salinesi. *Cadre Méthodologique pour l'Analyse du Changement d'Entreprise et de l'Adaptation de Systèmes d'Information*. Centre de Recherche en Informatique (CRI), Livrable Projet FT R&D N° 01 1B A30, 2002.
- C. Salinesi, M. Ben Ayed, S. Nurcan. *Development Using Lyee, a Case Study with LYEEALL*. Centre de Recherche en Informatique (CRI), 2001.
- C. Salinesi, J. Wäyrynen. *Business Maps Presentation of Concepts and Illustration with the Stockholm Central Service (SCS) Case Study*. Centre de Recherche en Informatique (CRI), Livrable Projet FT R&D N° 01 1B A30, 2001.

Curriculum

Thèses soutenues

- Laure Hélène Thévenet. « Vers une Modélisation Conceptuelle de l'Alignement Stratégique ». (Décembre 2009)
- Elena Ivankina. « RITA, une méthode pour guider la Découverte d'Exigences par l'Analyse des Menaces ». (Novembre 2009)
- Ines Gam. « Ingénierie des Exigences pour les Systèmes d'Information Décisionnels : Concepts, Modèles et Processus. La Méthode CADWE » (2008)
- Emmanuel Papadacci. « Evaluation Qualitative et Quantitative de Scénarios d'Evolution de l'Organisation et de son Système d'Information » (2008).
- Anne Etien, « Ingénierie de l'alignement : Concept, Modèle et Processus. La méthode ACEM pour l'alignement d'un système d'information aux processus d'entreprise » (2006).
- Iyad Zoukar, « MIBE : Méthode d'Ingénierie des Besoins pour l'Implantation d'ERP ». (2005).

Direction de thèses en cours

- Olfa Djebbi. « REDPL, une Méthode pour la Configuration Optimale de Produits à Partir de Modèles de Domaines »
- Ramzi Bouzid. « Réutilisation lors de la Personnalisation d'ERP »
- Elena Kornyshova. « L'Emploi des Techniques d'Aide à la Décision Multicritère en Ingénierie des Systèmes d'Information »
- Raul Mazo. « Vérification de Modèles de Ligne de Produit »

Appartenance à des jurys de thèse

- Roberto Santana Tapia "Assessing business-IT alignment in networked Organizations". Université de Twente, Pays Bas. Décembre 2009.
- Stephen Boyd "Design and Evaluation of a Method to Reduce Lexical Ambiguity of Requirement Specifications". University of Technology, Sydney, Australie. Mars 2009.
- Ammar Qusaibaty. « La Conception d'un Processus Guidé par l'Intention ». Université Paris 1 Panthéon – Sorbonne. Novembre 2008.
- Howard Newton. « Le Paradigme Centré sur l'Intention ». Avril 2007.
- Régis Kla. « OsMoSE : Une Approche de Modélisation Orientée Objet Basée sur le Langage Naturel et un Outil d'Aide ». Université Paris 1 Panthéon – Sorbonne. Juin 2004.
- Mustapha Tawbi. « CREWS-L'Ecritoire : un Guidage Outillé du Processus d'Ingénierie des Besoins ». Université Paris 1 Panthéon - Sorbonne. Octobre 2001.

Conférences Invitées

- « Les exigences Confrontées à l'Architecture » dans le cadre d'un colloque professionnel d' « Introduction aux Systèmes d'Information » à l'Institut National Géographique, Dec 2007.
- « Change and Alignment » au Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia (Espagne), Avril 2007.
- « Les Exigences de Changement » au Laboratoire Etudes des Systèmes Informatiques et Automatiques LESIA) de l'INSA à Toulouse, Mai 2007.

Valorisation

Projets Institutionnels

- DELLISS : Montage et mise participation au projet de collaboration multilatérale DELIIS financé par la communauté Européenne dans le cadre de ERASMUS avec les institutions faisant partie de l'EIIS et plusieurs entreprises partenaires dont, pour la France IBM Europe, l'ANPE, le Groupement pour la modernisation du Système d'Information Hospitalier, BNP Paribas, Volle Systèmes d'Information, et la société Fast booking.
- CREWS : participation au projet Long Term Research financé par la communauté Européenne (FPVI) sur le thème de l'Ingénierie des Exigences par les Scénarios. Développement et outillage de la méthode L'Ecritoire.
- NATURE : participation au projet Long Term Research financé par la communauté Européenne (FPIV) sur le thème de l'Ingénierie des Exigences. Participation au développement de l'outil Mentor d'exécution de processus de décision.

Coopération scientifique

- Création de l'Institut Européen EIIS pour l'innovation dans la science des services. Avec le CRP Henri Tudor (Luxembourg), l'Université de Genève (Suisse), l'Université de Skovde (Norvege), l'Université Polytechnique de Catalogne (Espagne), Université Vrije (Pays Bas), et l'Université de Porto (Portugal)
- Fondation du Groupement d'Interêt Scientifique INTEROP Grande Région, sur le thème de l'interopérabilité avec les universités de Nancy 1, Nancy 2, et la Faculté des Universités Notre Dame de la Paix de Namur dans le cadre du Groupement d'Intérêt Scientifique « GIS INTEROP Grande Région ».
- Membre actif du groupe de travail Ingénierie des Exigences (GTIE) de l'Association Française d'Ingénierie des Systèmes (AFIS)
- Membre du Requirements Engineering Working Group (REWG) et ambassadeur de l'International Council of Systems Engineering (INCOSE)

La participation à ces clubs de professionnels me permet de rencontrer ceux-ci de manière régulière, et de développer des coopérations sur des thèmes de recherche appliquée. Par exemple:

- L'outil REGAL (<http://p-ring.net/>) actuellement utilisé par l'INCOSE au niveau mondial pour le partage de bonnes pratiques d'Ingénierie des Exigences a été développé avec Laure Hélène Thévenet par réutilisation de l'outil DEBATER qu'elle avait développé sous ma direction lors de son stage de Master.
- Plusieurs communications faisant état de travaux de recherche appliquée ont été co-rédigés avec les membres de ces clubs ([INV3] [ACT10] [ACT42] [ACT43] [COM7] [AFF4]).
- Vulgarisation par le biais de présentations et animation de groupes de réflexion sur l'exploitation de nos résultats de recherche, par exemple sur le thème de l'utilisation des scénarios en Ingénierie des Exigences à l'AFIS.

Collaboration avec des entreprises et transfert

J'ai mené plusieurs projets de coopération avec des entreprises. Ces coopérations ont permis la définition de sujets de recherche appliquée et l'expérimentation empirique des méthodes, techniques et outils issus de mes travaux.

- EDF : projet ayant pour objet une recherche bibliographique sur le thème de l'urbanisation des systèmes d'information. Les résultats issus de cette collaboration ont fait l'objet de plusieurs publications. 2004.
- BNP-Paribas : projet faisant l'objet d'un contrat CIFRE sur le thème de l'alignement stratégique,
- Renault : projet faisant l'objet d'un contrat CIFRE, sur le thème de la priorisation des exigences dans le cadre de portefeuilles de projets

- STAGO Instruments : projet faisant l'objet d'un contrat CIFRE sur le thème de la rétro-conception d'une ligne de produit.
- SNCF : projet faisant l'objet d'un contrat CIFRE sur le thème de la personnalisation de Progiciels de Gestion Intégrés.
- France Télécom : étude sur le thème de l'alignement stratégique

Développement de logiciels

- REGAL : plateforme logicielle pour le partage de connaissances méthodologiques en partenariat avec l'International Council of Systems Engineering. Cette plateforme est actuellement utilisée par l'INCOSE (International Council of Systems Engineering) pour le partage et l'évaluation de bonnes pratiques d'ingénierie des exigences.
- CREWS-L'Ecritoire : outil pour la découverte d'exigence guidant par la rédaction, la conceptualisation et l'analyse de scénarios textuels. Développé dans le cadre du projet Européen LTR FPV CREWS.
- Mentor : outil de guidage méthodologique par l'exécution de modèles de processus décisionnels. Développé dans le cadre du projet Européen LTR FPV NATURE.

Animation de la recherche

- Présidence du Comité d'Organisation de la Conférence RE'05 (13th IEEE International Conference on Requirements Engineering) qui a eu lieu à la Sorbonne en Septembre 2005. La conférence a rassemblé plus de 450 inscrits qui ont participé à 12 ateliers de travail, 6 tutoriels et 3 journées de séances plénières.
- Co-animation de l'atelier « l'Ingénierie des Systèmes par la Recherche » au Forum Académique AFIS en Décembre 2007.

Travaux d'expertise

- Membre du comité de programme des conférences internationales:
 - CAiSE,
 - RE,
 - REFSQ,
 - ICIS,
 - INFORSID,
 - IBIMA,
 - BDA
- Membre du comité de programme des ateliers:
 - AWRE,
 - POEM,
 - RIGIM,
 - SOCCER,
 - VAMOS,
 - REBNITA
- Relecteur pour les revues:
 - Requirements Engineering Journal (Springer),
 - IEEE Transactions on Software Engineering,
 - Automated Software Engineering (Springer),
 - Information and Software Technology (Elsevier),
 - Software Process Improvement and Practice (Wiley),
 - IEEE Software, Journal of Research and Practice in Information Technology (Australian Computer Society),
 - International Journal of Software Engineering and Knowledge Engineering (World Scientific)

Enseignement

Responsabilités pédagogiques

- Responsable de l'équipe pédagogique informatique de:
 - la Licence Gestion et Finance (UFR06), depuis 2000
 - la Licence Economie et Gestion des Entreprises (UFR06), de 2005 à 2007
 - la Licence Mathématiques et Informatique Appliquée (UFR27), depuis 2007
- Direction de nombreux mémoires de Master Professionnel (MIAGE, CAAE) et Recherche (SID).
- Professeur référent à la Licence Mathématique, Informatique et Applications

Enseignement

Matières

- Développement de Systèmes Transactionnels (L3)
- Algorithmique (L3)
- Bases de données (L3)
- Personnalisation de Systèmes ERP (L3, M2)
- Urbanisation et Enterprise Architecture (M2)
- Ingénierie des Exigences (M2)
- Management des SI (M2)

Publics

Université Paris 1 Panthéon - Sorbonne

- Licence 3 MIAGE (UFR27) : Développement de Systèmes transactionnels, Algorithmique
- Licence 3 Gestion Finance (UFR06) : Informatique (bases de données)
- Master 2 MIAGE/SIC (UFR27) : Personnalisation de Systèmes ERP, Urbanisation et Enterprise Architecture
- Master Systèmes d'Information et de Décision (UFR27) : Ingénierie des Exigences
- Master CAAE (IAE Paris) : Informatique (modélisation de données)
- Master Comptabilité Contrôle et Audit (UFR06) : Management des SI

Autres institutions

- CNAM (cursus NFE101 « Ingénierie de reconstruction ») : Progiciels de Gestion Intégrés
- CNAM (Licence Professionnelle LP008 ACSID « Analyse et Conception des Systèmes de Décision ») : Progiciels de Gestion Intégrés
- EFREI (Master 2 option Systèmes d'Information) : Progiciels de Gestion Intégrés
- Institut National Géographique & Ecole Nationale de Ponts et Chaussées (Master) : Ingénierie de la Solution
- Université de Genève : Analyse des exigences d'évolution
- Alger Business School (MBA) : Analyse et Spécification des Exigences

Responsabilités administratives

- Membre de la commission de spécialistes « Mathématiques et Informatique » de l'Université Paris 1
- Membre du Conseil de l'UFR 27

