



De l'ingénierie des besoins à l'ingénierie des exigences : vers une démarche méthodologique d'ingénierie de systèmes complexes, de vérification et de validation appliquée à l'informatisation des PME

Nawel Amokrane

► To cite this version:

Nawel Amokrane. De l'ingénierie des besoins à l'ingénierie des exigences : vers une démarche méthodologique d'ingénierie de systèmes complexes, de vérification et de validation appliquée à l'informatisation des PME. Autre [cs.OH]. Université Montpellier, 2016. Français. NNT : 2016MONTT259 . tel-01816973

HAL Id: tel-01816973

<https://theses.hal.science/tel-01816973>

Submitted on 15 Jun 2018

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.

THÈSE

Pour obtenir le grade de
Docteur

Délivré par l'UNIVERSITÉ MONTPELLIER

**Préparée au sein de l'école doctorale
Information, Structure et Systèmes I2S
et de l'unité de recherche LGI2P
de l'école des mines d'Alès**

**Spécialité : Systèmes Automatiques et Microé-
lectroniques**

Présentée par Nawel AMOKRANE

**De l'Ingénierie des Besoins à l'Ingénierie
des Exigences : Vers une Démarche Mé-
thodologique d'Ingénierie de Systèmes
Complexes, de Vérification et de Validation
Appliquée à l'Informatisation des PME**

Soutenue le 27/04/2016 devant le jury composé de

Mr. Jean-Pierre BOUREY, Professeur, école Centrale de Lille	Rapporteur
Mr. Yves DUCQ, Professeur, U.Bordeaux	Rapporteur
Mme Aline CAUVIN, Maître de Conférences HdR, Polytech'Marseille	Examineur
Mr. Didier CRESTANI, Professeur, U.Montpellier	Examineur, président du jury
Mr. Mohssine RAHHOU, Directeur, société RESULIS	Examineur
Mme Anne-Lise COURBIS, Maître-Assistant, école des mines d'Alès	Encadrant
Mr. Thomas LAMBOLAIS, Maître-Assistant, école des mines d'Alès	Encadrant
Mr. Vincent CHAPURLAT, Professeur, école des mines d'Alès	Dir. de Thèse

TABLE DES MATIERES

Introduction générale	7
Chapitre I : L'informatisation des PME Contexte et problématique	10
1.1 Introduction.....	11
1.2 Contexte : ingénierie des besoins et des exigences pour l'informatisation des PME ..	11
1.2.1 Caractéristiques des Petites et Moyennes Entreprises (PME)	11
1.2.2 Ingénierie des besoins et des exigences : concepts et définitions.....	14
1.3 Conclusion : problématique RESULIS et objectifs des travaux	19
Chapitre II : État de l'art	22
2.1 Introduction.....	23
2.2 Comprendre la PME : apports de la Modélisation d'Entreprise	23
2.2.1 Modèle d'entreprise.....	25
2.2.2 Principes de la Modélisation d'Entreprise.....	25
2.2.3 Cadres et méthodes en Modélisation d'Entreprise.....	26
2.2.4 Efforts d'unification et de normalisation	29
2.2.5 Synthèse : verrous constatés.....	32
2.3 L'IBE pour comprendre et formaliser le projet d'évolution de la PME.....	34
2.3.1 Analyse comparative de langages de modélisation en IBE	35
2.3.2 Vers des notations plus accessibles en IBE.....	41
2.3.3 Synthèse : verrous constatés.....	43
2.4 Progresser en confiance durant le processus d'IBE : vérification et validation	44
2.4.1 Définitions : vérification et validation des exigences	44
2.4.2 Qualités attendues d'une exigence et d'un référentiel d'exigences.....	45
2.4.3 Techniques de vérification et de validation	47
2.4.4 Synthèse	49
2.5 Conclusions et contribution des travaux.....	50
2.5.1 Positionnement	50
2.5.2 Hypothèse de travail : l'organisation avant l'informatisation.....	52
2.5.3 Contribution attendue.....	54
2.5.4 Exemple illustratif.....	55
Chapitre III : Concepts supports à la Méthode d'IBE	56
3.1 Introduction.....	57
3.2 Cadre normatif pour la construction du modèle de l'entreprise.....	57
3.3 Syntaxe abstraite pour l'expression des besoins et le passage des besoins aux exigences	59
3.3.1 Vue du contexte de l'étude	60
3.3.2 Vue organisationnelle.....	61
3.3.3 Vue des ressources.....	63
3.3.4 Vue fonctionnelle	64
3.3.5 Vue informationnelle.....	67
3.3.6 Vue des exigences	68
3.4 Règles de modélisation	70
3.4.1 Règles favorisant la complétude	71
3.4.2 Règles favorisant la non-reformulation.....	73
3.4.3 Règles favorisant la cohérence.....	73
3.5 Conclusion	75

Chapitre IV : Langages supports à la Méthode d'IBE.....	76
4.1 Introduction.....	77
4.2 Les DSL supports de la méthode d'IBE	77
4.2.1 Le DSL textuel	78
4.2.2 Les DSL graphiques.....	86
4.3 Conclusion	94
Chapitre V : Démarche illustrée et outillages	95
5.1 Introduction.....	96
5.2 Description de la démarche	96
5.2.1 Identification des parties prenantes	98
5.2.2 Elicitation des besoins	100
5.2.3 Vérification des besoins	107
5.2.4 Validation des restitutions	115
5.2.5 Validation des exigences	117
5.3 Conclusion	120
Conclusion générale, limites et perspectives.....	122
Références.....	126

LISTE DES FIGURES

Figure 1. Catégories d'entreprise selon la loi de modernisation de l'économie (OSEO 2012).	12
Figure 2. Parties prenantes intervenants habituellement dans un processus d'IBE.	18
Figure 3. Concepts des vues de la modélisation selon CIM-OSA (Berio & Vernadat 1999).	28
Figure 4. Concepts de base de UEML (Vernadat 2001).	30
Figure 5. Cadre de modélisation ISO19439 (ISO/DIS 2002).	32
Figure 6. Concepts ISO19440 et les quatre vues de la modélisation (ISO/DIS 2004).	33
Figure 7. Processus de RDM – CIM-OSA (Zelm et al. 1995).	36
Figure 8. Concepts proposés par la méthode KAOS (Lamsweerde 2009).	37
Figure 9. Démarche agile d'informatisation.	50
Figure 10. Cycle de développement basé sur les modèles. Apport de l'interopérabilité dirigé par les modèles (Bourey et al. 2007)	51
Figure 11. Enrichissement du cadre normatif ISO 19439 [Amokrane et al. 2014].	58
Figure 12. Concepts de la vue du contexte.	61
Figure 13. Concepts de la vue organisationnelle.	62
Figure 14. Concepts de la vue des ressources.	64
Figure 15. Concepts de la vue fonctionnelle.	66
Figure 16. Concepts de la vue informationnelle.	68
Figure 17. Concepts de la vue des exigences.	70
Figure 18. Représentation d'un modèle d'entreprise selon les vues de la modélisation et les vues des collaborateurs.	71
Figure 19. Questions intervenant dans la définition d'une activité.	85
Figure 20. Exemples de correspondance entre le DSL textuel et le métamodèle SI-PME.	90
Figure 21. Métamodèle du diagramme de contexte.	91
Figure 22. Construction de la syntaxe concrète du diagramme de contexte.	92
Figure 23. Processus proposé d'ingénierie des besoins et des exigences.	97
Figure 24. IHM - Identification des collaborateurs intervenant lors de l'expression des besoins.	100
Figure 25. IHM – définition des domaines d'activité, politique et objectifs stratégiques.	102
Figure 26. IHM – définition des partenaires, échanges, produits et services.	103
Figure 27. IHM – définition de l'organisation interne.	104
Figure 28. IHM – définition de ressources.	104
Figure 29. IHM – définition des rôles et des activités.	105
Figure 30. IHM – définition du détail d'une activité.	106
Figure 31. IHM – détection d'une erreur syntaxique.	109
Figure 32. Diagramme de contexte – partenaires de SDT.	115
Figure 33. Diagramme de contexte – services de SDT.	116
Figure 34. Organigramme partiel de SDT.	116
Figure 35. Diagramme de processus « Prise en compte de la commande client ».	117

LISTE DES TABLES

Tableau 1. Couvertures des vues de la ME par les langages d'IBE.....	40
Tableau 2. Concepts de la vue du contexte.	60
Tableau 3. Concepts de la vue organisationnelle.....	62
Tableau 4. Concepts de la vue des ressources.	63
Tableau 5. Concepts de la vue fonctionnelle.	64
Tableau 6. Concepts de la vue informationnelle.....	67
Tableau 7. Concepts de la vue des exigences.	69
Tableau 8. Formulations associées aux concepts de la syntaxe abstraite.	81
Tableau 9. Eléments de syntaxe concrète du diagramme de contexte.	87
Tableau 10. Eléments de syntaxe concrète de l'organigramme	88
Tableau 11. Contributions des choix conceptuels de la méthode aux qualités attendues des exigences.....	114
Tableau 12. Contributions des choix conceptuels de la méthode aux qualités attendues du référentiel des exigences.	115
Tableau 13. Template d'une exigence fonctionnelle.	118
Tableau 14. Template d'un objet d'entreprise.	120

Introduction générale

Face à une clientèle volatile et toujours plus exigeante, les entreprises, et particulièrement les petites et moyennes entreprises (PME), doivent être réactives et flexibles, préserver leurs marges et leurs capacités à innover et à suivre des marchés concurrentiels et mouvants. Ce sont des objectifs incontournables qui garantissent, une fois atteints, une prospérité et une reconnaissance dont les PME ont grand besoin.

Le système d'information (SI) est une composante essentielle de la mise en œuvre de la stratégie de toute entreprise pour atteindre ces objectifs. Il assure le rôle incontournable de courroie de transmission entre le système de décision (SD) et le système opérationnel (SO). Le SD peut se définir comme la partie de l'entreprise où s'évaluent puis se prennent les décisions stratégiques, tactiques et opérationnelles de l'entreprise. Le SO est, à l'opposé, la partie de l'entreprise créatrice de la réelle valeur ajoutée de cette dernière et ce par la production ou encore de la mise à disposition de produits ou de services répondant aux besoins des clients et représentant le cœur de métier de l'entreprise. Le SI est un ensemble organisé de ressources de diverses natures (matérielles, logicielles, humaines, données, procédures) permettant d'acquérir, de traiter, de stocker et de communiquer des informations (sous diverses formes). Il est de fait le siège d'activités pouvant être assistées par de l'outillage informatique et des applicatifs d'entreprise dont l'objectif est l'automatisation ou la fluidification des échanges entre SD, SO et l'environnement de l'entreprise. C'est pourquoi les PME sont tentées de se lancer, seule ou accompagnée, dans des projets dits d'informatisation i.e. des projets visant le développement ou l'amélioration de ses outils et applications informatiques et ce dans l'objectif de garantir l'efficacité du fonctionnement et de la prise de décision au sein de leur organisation.

Les travaux présentés dans ce manuscrit sont menés en collaboration avec la société RESULIS, dans le cadre d'un contrat CIFRE en partenariat avec le Laboratoire LGI2P de l'école des mines d'Alès et le soutien de l'ANRT. RESULIS est une SSII qui travaille essentiellement pour des PME afin de prendre en charge leurs projets d'informatisation. Nous nous intéressons plus particulièrement aux projets visant à développer des applicatifs de gestion et de pilotage pour ces PME.

La PME, prenant le rôle de la maîtrise d'ouvrage (MOA), tout comme RESULIS qui l'accompagne et remplit le rôle de maîtrise d'œuvre (MOE), doivent partager une vision commune des besoins d'informatisation. Ces deux parties prenantes sont de fait appelées à mener en commun des activités d'ingénierie des besoins et des exigences (IBE) avant même de passer au développement proprement dit des applicatifs attendus, activité dévolue alors à la MOE, avec de nécessaires allers-retours entre les parties prenantes. Au cours de cette IBE, la MOA doit formuler et structurer une description initiale de ses **besoins** pour permettre ensuite à la MOE de spécifier de manière plus formelle ces besoins sous forme d'**exigences** qui définissent alors les travaux de développement souhaités. La réussite du projet d'informatisation dépend fortement de la qualité des exigences définies en amont lors de la phase d'IBE. Lorsque la PME n'a pas la compétence ou les ressources pour conduire la maîtrise d'ouvrage et rédiger seule ses besoins, elle est amenée à faire appel à une

troisième partie prenante ayant le rôle d'assistance à maîtrise d'ouvrage. Cependant, les PME ne peuvent souvent pas se faire assister par manque de moyens financiers. Par conséquent, ces activités d'IBE sont la plupart du temps confiées à des équipes faisant partie de la MOE qui établissent les besoins de la PME et ce souvent de manière informelle ne lui permettant pas de s'assurer ou de rassurer la PME de leur qualité. Or, l'implication des collaborateurs de la PME est primordiale à ce niveau de formalisation des besoins, surtout si l'applicatif ou les applicatifs attendus impactent durablement le fonctionnement de la PME.

Dans ce contexte, ces travaux de recherche tentent de répondre à la problématique de RESULIS qui souhaite mettre en œuvre des activités d'ingénierie des besoins et des exigences lors des projets d'informatisation. L'objectif est de favoriser l'implication de la maîtrise d'ouvrage i.e. les PME clientes de RESULIS, avec la MOE i.e. le département en charge des développements chez RESULIS. RESULIS estime en effet que l'implication de la PME passe par l'autonomie de ses collaborateurs pour décrire leurs besoins. Cet objectif se décline en sous-objectifs comme suit. Premièrement, il faut proposer à ces collaborateurs de la PME des moyens d'expression de leurs besoins qui soient simples, si possibles intuitifs et aisément manipulables en privilégiant l'emploi de termes et de concepts en phase avec leur langage métier. Deuxièmement, RESULIS souhaite pouvoir disposer d'exigences de conception de l'applicatif qui traduisent sans perte ou défaut ces besoins, qui soient claires, si possible validées et approuvées par ces collaborateurs. Troisièmement, ces exigences doivent être exploitables pour entamer une conception et un développement itératif des solutions proposées à la PME cliente, progressant ainsi efficacement vers la solution attendue. De fait, des actions de vérification et de validation sont requises pour éviter que les défauts inhérents aux besoins, tels que l'incohérence, l'ambiguïté ou la non-complétude ne se reportent sur ces exigences.

Le présent manuscrit présente comment ces trois objectifs ont été interprétés puis solutionnés. Le premier chapitre expose précisément le contexte et la problématique de ces travaux. Le plan du reste du manuscrit sera également présenté en conclusion du premier chapitre.

Chapitre I : L'informatisation des PME Contexte et problématique

1.1 Introduction

Cette thèse aborde la problématique de la conduite d'un projet d'informatisation pour la réalisation d'applications de gestion supports au Système d'Information (SI) d'entreprises de type PME (Petite et Moyenne Entreprise). Plus particulièrement, nous considérons la première phase de ce projet, à savoir la phase d'Ingénierie des Besoins et des Exigences (IBE). Nous souhaitons assurer une collaboration réussie entre la PME qui exprime des besoins relatifs au projet d'informatisation et la maîtrise d'œuvre (MOE) qui s'engage à concevoir et à réaliser un ou des applicatifs répondant aux besoins de la PME. Ces travaux ont été développés en collaboration avec la société RESULIS, qui joue ici le rôle de la MOE, une société de service et de conseil en informatique installée à Alès, Gard. RESULIS travaille essentiellement pour des PME en leur fournissant des applications logicielles originales, conçues en adéquation avec leur processus de gestion.

Ce chapitre a pour objectif de positionner le contexte de ces travaux. Les caractéristiques des PME sont tout d'abord présentées, mettant en avant leurs possibles difficultés à se lancer dans un projet d'informatisation et à communiquer leurs besoins à la maîtrise d'œuvre. Cette MOE doit de fait savoir collecter et structurer les besoins de l'entreprise pour mieux y répondre, rendant nécessaire l'application d'activités d'ingénierie des besoins et des exigences. Les concepts et pratiques de l'ingénierie des besoins et des exigences sont introduits plus précisément avant de présenter les objectifs attendus de ces travaux.

1.2 Contexte : ingénierie des besoins et des exigences pour l'informatisation des PME

1.2.1 Caractéristiques des Petites et Moyennes Entreprises (PME)

1.2.1.1 Définition des PME

L'entreprise est un lieu de création de valeurs, liées au développement et à la commercialisation de produits et/ou de services destinés à un marché donné et dont une clientèle solvable reconnaît l'utilité, dans un ou plusieurs domaines d'activité et en interagissant avec des partenaires externes. L'entreprise évolue suivant une politique et des objectifs stratégiques qui favorisent l'obtention d'un avantage compétitif.

En comparaison avec les grandes entreprises ou encore les grands groupes, les petites et moyennes entreprises (PME) ont généralement une structure plus simple avec un ensemble plus limité de ressources. Ce sont, selon les textes (Figure 1), des entreprises qui emploient moins de 250 salariés et dont le chiffre d'affaires annuel n'excède pas 50 millions d'euros et/ou le total annuel de bilan n'excède pas 43 millions d'euros. Parmi elles, les microentreprises (MIC) occupent moins de 10 personnes et ont un chiffre d'affaires annuel ou un total de bilan n'excédant pas 2 millions d'euros (OSEO 2012).

		Effectifs			
Chiffre d'affaires	Total Bilan	Moins de 10 salariés	De 10 à 249 salariés	De 250 à 4 999 salariés	5 000 salariés et plus
2 M€ au plus	2 M€ au plus	Microentreprises	Petites et moyennes entreprises (PME)	Entreprises de taille intermédiaire (ETI)	Grandes entreprises (GE)
Plus de 2 M€ à 50 M€ inclus	2 M€ au plus				
	Plus de 2 M€				
Plus de 50 M€ à 1,5 Md€ inclus	2 M€ au plus	Microentreprises			
	Plus de 2 M€ à 43 M€ inclus				
	Plus de 43 M€				
Plus de 1,5 Md€	2 M€ au plus	Microentreprises	PME		
	Plus de 2 M€ à 43 M€ inclus				
	Plus de 43 M€ à 2 Md€ inclus				
	Plus de 2 Md€				

Figure 1. Catégories d'entreprise selon la loi de modernisation de l'économie (OSEO 2012).

Les PME jouent un rôle central dans l'économie de la plupart des pays et leur poids économique est décisif. Pour la Commission Européenne, elles sont le moteur de l'économie puisqu'elles sont une source majeure de compétences entrepreneuriales, d'innovation et d'emplois. En France, durant l'année 2013, les PME représentaient plus de 98% de la totalité des entreprises (dont 94% sont des microentreprises) et emploient 42% des salariés (Planson et al. 2013). Nous remarquons donc que la quasi-totalité des PME sont des microentreprises. En outre, sur 129 000 PME étudiées, le profil type des PME françaises est de 20 salariés avec un chiffre d'affaires de 4,6 millions d'euros assorti d'une valeur ajoutée de 1,3 million d'euros (OSEO 2012). De fait, leurs ressources humaines sont limitées et centrées principalement sur le cœur de métier de l'entreprise.

Les différentes expériences de RESULIS lui ont permis de constater que les PME, et surtout les microentreprises, fonctionnent en suivant des procédures simples et évolutives au gré des situations. Leurs processus métiers impliquent des ressources humaines qui collaborent, généralement polyvalentes, sans liens nécessairement clairs entre leurs rôles ou entre les responsabilités exercées. Elles ont aussi une structure organisationnelle qualifiable de simple en raison de leur petite taille, avec peu de profondeur dans la hiérarchie des postes.

1.2.1.2 Stratégies adoptées et compétences des PME pour se doter d'applications de gestion

La PME étant créée, elle se développe, accroît sa charge de travail pour faire face aux demandes de ses clients ou encore pour suivre l'évolution de son marché. Cette PME est donc logiquement conduite à passer d'un traitement plutôt manuel de ses processus de gestion à un traitement plus assisté par des moyens informatiques puis totalement informatisé. Enfin, au fil du temps, elle doit également penser à l'amélioration ou à l'adaptation de ces moyens.

Les PME informatisent leurs processus graduellement et leur stratégie d'acquisition des solutions informatiques évolue au fil du temps.

Il est important de noter que les PME manquent de vision stratégique quant à l'utilisation des Technologies de l'Information et de la Communication (TIC). Selon (Besnard et al. 2008), l'implantation des TIC est plus forte dans les grandes entreprises comparée aux PME. Cette différence est d'autant plus importante que les applications sont complexes et correspondent à un état avancé de l'utilisation des TIC comme l'utilisation des progiciels de gestion intégrée (PGI), outils de travail collaboratif (groupware) ou des outils de modélisation et d'automatisation des processus (BPM) (Faure 2010).

Les solutions qui assistent les processus qui font partie du cœur du métier de la PME (systèmes de gestion de la production, technologies spécifiques...) sont plus prioritaires et leur acquisition accompagne souvent la création de l'entreprise. Les solutions supports aux autres processus de gestion tels que la gestion des ventes, la comptabilité, la planification, etc. sont acquises généralement plus tard. De ce fait, les PME se dotent de solutions disparates, développées en interne ou achetées sur étagère lorsqu'elles sont financièrement et techniquement abordables. De plus, cela est souvent lié à l'initiative personnelle des collaborateurs. Toutefois, soumise à divers changements au fil des années, cette situation engendre un empilement de développements logiciels souvent non interopérables, jusqu'à ne plus répondre aux objectifs de croissance ou d'optimisation de la PME. Afin de réduire cette complexité et mettre les nouvelles possibilités technologiques au service de l'entreprise, les responsables des PME se tournent vers des solutions plus globales qui prennent en compte les processus de bout en bout. Ils ont alors le choix entre :

- (i) l'acquisition de solutions basées sur des progiciels intégrés tels que les ERP, CRM, SCM, etc. Ce type de solution est souvent contraignant du point de vue du paramétrage et de l'adaptation à la PME et va à l'encontre de la souplesse souhaitée des procédures de la PME (Cragg et al. 2011). Ces solutions proposent en effet des procédures et des fonctionnalités souvent trop élaborées, voire inappropriées à l'organisation relativement simple des PME. Ces progiciels impliquent également une série de changements que la PME doit entreprendre dans un but d'alignement de son fonctionnement propre avec celui plus ou moins imposé par ce progiciel. L'adhésion des collaborateurs est dans ce cas primordiale pour exploiter au mieux cette nouvelle composante du SI de la PME.
- (ii) le développement spécifique de solutions conçues pour répondre précisément à leurs besoins, où les exceptions de gestion liées à la nature du métier et à la façon de faire de la PME sont prises en compte. Ceci évite une rupture brutale des habitudes des collaborateurs de la PME, garantit l'adhésion de ces collaborateurs et permet de converger vers une solution d'informatisation correspondant aux attentes et aux pratiques de la PME.

Lorsqu'une PME décide d'informatiser ses processus par un développement spécifique, elle peut être tentée de lancer un projet de développement des applications supports en interne mais elle manque de compétences quant à la gestion d'un projet d'une telle envergure. En effet, Le niveau de compétences présentes au sein de la PME en termes de développement

et de gestion de tout ou partie des SI est souvent limité. En France, selon une enquête de l'INSEE (INSEE 2013), 70% des entreprises parmi celles d'au moins 250 personnes emploient des spécialistes en TIC (ingénieurs informaticiens, techniciens...) contre seulement 15% des sociétés d'au moins 10 personnes. Néanmoins, ces spécialistes représentent 7 % des effectifs (Besnard et al. 2008). Ces collaborateurs sont principalement employés pour la maintenance du parc informatique de la PME et n'ont pas des profils d'analyste ou de concepteur.

Les responsables de la PME font donc appel à un tiers qui possède les compétences requises et des méthodes de travail plus adéquates et certainement plus productives. Le prestataire externe, appelé maîtrise d'œuvre (MOE), est communément sélectionné par un mécanisme d'appel d'offre, généralement après que l'entreprise a défini un cahier des charges décrivant plus ou moins bien ses besoins. De cette façon, la sous-traitance crée des tâches nouvelles liées à la formalisation du contenu de la prestation dans un cahier des charges. Or les ressources et compétences de la PME ne lui permettent pas de mener ces tâches efficacement.

Nous nous intéressons dans la section suivante, à la première étape d'un tel projet d'informatisation. Il s'agit de l'étape d'ingénierie des besoins et des exigences qui est une suite d'activités cruciales pourtant sous-estimées dans le milieu des PME.

1.2.2 Ingénierie des besoins et des exigences : concepts et définitions

L'ingénierie des besoins et des exigences (IBE) est une discipline commune à toutes les ingénierie¹ métiers (génie logiciel, mécanique, électronique, etc.) comme à l'Ingénierie Système (BKCASE Editorial Board 2015) qui la reconnaît comme la phase décisive la plus en amont dans tout projet d'ingénierie d'un système i.e. de produits et/ou de services satisfaisant les besoins de différentes parties prenantes².

Le processus regroupant toutes les activités d'IBE est défini par (Loucopoulos & Champion 1989) comme étant *“the systematic process of developing requirements through an iterative process of analyzing a problem, documenting the resulting observations, and checking the accuracy of the understanding gained”*. Il suppose donc l'application systématique et répétitive de techniques d'analyse, de documentation et de vérification³ et de validation⁴ des exigences.

Dans le domaine du Génie Logiciel, selon (Rolland 2003) « *l'ingénierie des besoins est concernée par l'identification de buts assignés au système envisagé et l'opérationnalisation de ces buts en contraintes et exigences imposées au système et aux agents qui en assureront le fonctionnement. L'ingénierie des besoins peut être vue comme le processus qui permet de*

¹ Une ingénierie peut être définie comme une suite d'activités de conception, de vérification et de construction d'un artefact jugé suffisant pour répondre aux besoins de parties prenantes identifiées.

² Toute personne, groupe de personnes ou organisation pouvant influencer ou être affecté par le système ou son élaboration, de façon directe ou indirecte (Sharp et al. 1999).

³ La vérification permet de répondre à la question : « *Construisons-nous correctement le modèle ?* »

⁴ La validation est réalisée sur des modèles vérifiés et permet de répondre à la question : « *Construisons-nous le bon modèle ?* »

transformer une idée floue en spécification précise des besoins servant de support à la spécification du système et de ses interfaces avec l'environnement ». Notons enfin que l'Ingénierie Système (BKCASE Editorial Board 2015) propose deux processus distincts d'ingénierie des besoins et d'ingénierie des exigences qui se positionnent en amont des processus de conception architecturale de la solution.

L'IBE guide et accompagne le passage d'une description initiale, souvent peu claire voire contradictoire que nous appelons **besoin**, à une spécification plus formelle et plus contractuelle de ce besoin, appelée **exigence**. Ceci se fait au moyen d'activités, entre autres, d'expression, de documentation et d'évaluation continue des besoins et de construction d'un référentiel d'exigences.

Dans la littérature anglo-saxonne, les auteurs ne différencient pas réellement les concepts de besoin et d'exigence. En effet, selon (Chemuturi 2012) « *Une exigence est un besoin, une attente, une contrainte ou une interface de toutes les parties prenantes, qui doit être satisfaite par le système logiciel proposé* ». Le besoin est alors plutôt vu comme une attente basique voire vitale, le minimum absolu sans lequel le système ne sera pas utilisable. Nous préférons la stratification proposée par (Jackson 1997) « *Une exigence est le produit intermédiaire entre le besoin et le programme* ».

Nous considérons dans la suite deux phases de maturité qui distinguent les besoins des exigences :

- **Besoin** : « *stakeholder requirement* » ou « *user requirement* » en anglais, représente l'expression initiale de la MOA, considérée alors comme la représentante des parties prenantes commanditaires. Il exprime et décrit une situation, un vœu, un but ou une contrainte vis-à-vis du système à réaliser. Les besoins sont le plus souvent définis de façon informelle, dans le langage de la maîtrise d'ouvrage, afin de simplifier la communication avec ces parties prenantes. Ils reflètent le monde du problème.
- **Exigence** : « *system requirement* », une exigence est définie comme « *un énoncé qui prescrit une fonction, une aptitude ou une caractéristique à la quelle doit satisfaire un produit ou un système dans un contexte donné* » (Scukanec & Gaasbeek 2007). Les exigences traduisent donc les besoins, elles sont établies de fait dans le langage de la maîtrise d'œuvre. Elles représentent les propriétés attendues du système qui vont ainsi contraindre la conception de la solution. Une collaboration MOE / MOA est ainsi nécessaire pour analyser les besoins et déduire les exigences. Ces exigences sont organisées dans un référentiel, i.e. l'ensemble des exigences qui reflète les consensus et décisions prises par les parties prenantes et est à la base d'un document contractuel définissant les rôles et responsabilités des commanditaires et prestataires.

En ce qui concerne les projets de développement d'applications logicielles, la distinction entre besoin et exigence est d'autant plus importante qu'elle permet de mettre l'accent sur le problème à solutionner en considérant le point de vue de chaque partie prenante pour

éviter que les seules descriptions du système à réaliser soient exprimées en termes de solution (Hull et al. 2010).

Il existe deux catégories d'exigences (ISO15288 2008) à savoir : les exigences fonctionnelles et les exigences non-fonctionnelles. **Les exigences fonctionnelles** font référence aux fonctionnalités et aux services que le système doit fournir. Lorsqu'il s'agit par exemple d'une application de gestion, elles traduisent les opérations à automatiser et sont fortement liées au fonctionnement de l'entreprise. **Les exigences non-fonctionnelles** contraignent et précisent comment le système doit satisfaire les exigences fonctionnelles en termes de performance, sécurité, fiabilité, ergonomie, niveau de maintenabilité, portabilité, réutilisabilité, etc. Elles peuvent concerner également certains aspects liés à la gestion du projet de développement comme le coût et les délais de réalisation. Leur identification est importante puisqu'elles servent de critères de sélection permettant à l'équipe de développement de faire des choix en termes de conception et d'implémentation.

La première référence à la nécessité d'actions d'ingénierie des besoins et des exigences était faite au milieu des années 70 : « *the requirements for a system do not arise naturally; instead, they need to be engineered and have continuing review and revision* » (Bell & Thayer 1976) et l'IBE a été alors reconnue comme discipline au milieu des années 80 (Lamsweerde 2000). Depuis, un grand nombre de méthodes et de langages ont été proposés, suivant différentes orientations (buts, scénarios, etc.) qui sont repris au chapitre II. Cependant, quelle que soit la méthode proposée, l'objectif des travaux en IBE est d'améliorer la qualité des exigences en atténuant les défauts, omissions ou erreurs souvent rencontrés dans la phase d'expression des besoins et inhérents à la nature informelle de ces derniers, tels que :

- **L'ambiguïté** : informations données au niveau des besoins ayant différentes interprétations possibles;
- **L'imprécision** : besoins flous ou insuffisamment définis, structurés ou détaillés suite à des reformulations par exemple ambiguës;
- **L'omission** : besoins omis par oubli objectif ou subjectif, menant ici aussi à des incohérences ou des imprécisions;
- **La redite** : besoins exprimés plus d'une fois ou reformulés en d'autres termes induisant encore des incohérences ou de l'ambiguïté;
- **L'incohérence** : besoin en contradiction avec un ou plusieurs besoins.

Nous pouvons aussi citer le bruit, la non-mesurabilité, la non-intelligibilité ou la non-traçabilité (Lamsweerde 2009). Certains de ces défauts pourraient être levés ou limités par l'emploi de méthodes éprouvées, de bonnes pratiques, le respect de normes et de standards tels que ceux proposés par (INCOSE 2012) et (ITU-T 2008) dont une analyse est faite au chapitre II, analyse qui nous permettra d'argumenter les contributions attendues de ces travaux.

La présence de ces défauts peut avoir des conséquences désastreuses dans les phases suivantes du projet d'informatisation ou pendant l'exploitation du système. En effet, dès le milieu des années 70, les auteurs de (Bell & Thayer 1976) ont remarqué que des exigences ina-

déquates, incohérentes ou ambiguës ont un impact critique sur la qualité des applications qui en résultent. C'est aujourd'hui un des problèmes reconnu dans le domaine du Génie Logiciel, comme le montrent des études telles que celles proposées par (Ibanez & Rempp 1996; The Standish Group 2009). Par exemple, (The Standish Group 2009) souligne que 44% des projets de développement logiciel ont dépassé leur délai et budget sans nécessairement arriver, d'une part, à décrire et, d'autre part, à respecter toutes les exigences attendues. Ceci est lié principalement à une gestion inadéquate des exigences. Il est donc proposé par les praticiens de l'IBE de systématiser un processus d'expression puis de passage des besoins aux exigences. Ce processus est composé généralement des activités suivantes :

1 - L'identification des parties prenantes. Il est important d'identifier toute personne, groupe de personnes ou organisation pouvant influencer ou être affecté par le système ou son élaboration, de façon directe ou indirecte (Sharp et al. 1999). Le processus d'ingénierie des besoins et des exigences concerne ainsi un grand nombre de parties prenantes en relation avec les parties prenantes principales du projet d'informatisation, à savoir la maîtrise d'ouvrage et la maîtrise d'œuvre : les clients, les fournisseurs, le législateur, les sous-traitants, etc. La liste des parties prenantes peut également évoluer durant le processus d'IBE. Aussi, dans la plupart des projets d'informatisation, les entreprises font appel à une assistance à maîtrise d'ouvrage pour la représenter, jouer le rôle de coordinateur avec la MOE et l'aider à élaborer le cahier des charges.

La MOE exprime des contraintes qui sont vues ici comme un type d'exigences non-fonctionnelles, par exemple en termes de gestion de projet, de disponibilité des acteurs ou de techniques utilisées lors du développement. Elle apporte son expertise et doit proposer à la MOA des solutions fonctionnelles.

2 - Compréhension du domaine et expression des besoins. C'est ici que le problème à résoudre est décrit. Après l'identification des parties prenantes, il est nécessaire de les impliquer pour qu'elles puissent décrire, partager et faire comprendre :

- les objectifs stratégiques et la politique de l'entreprise, ses partenaires essentiels concernés *a priori* par le futur système,
- le contexte dans lequel le système devra être exploité, le fonctionnement, l'organisation et les interactions de l'entreprise avec son environnement,
- les besoins fonctionnels, non fonctionnels et les contraintes liées à l'organisation et aux ressources internes à l'entreprise auxquels le système doit répondre et les informations qu'il doit gérer.

3 - Analyse, vérification et négociation. Le processus d'IBE implique plusieurs des parties prenantes identifiées. C'est pourquoi les besoins qu'elles expriment peuvent être contradictoires voire conflictuels de par les rôles détenus par ces parties prenantes dans le processus d'IBE ou dans l'entreprise, mais aussi de par les points de vue de chacun sur le système souhaité. Ces conflits doivent être détectés et résolus. Une phase d'explication, de vérification, de justification et de négociation est requise pour parvenir à un consensus.

4 - Passage des besoins aux exigences, vérification et documentation des exigences. Le passage des besoins aux exigences doit être systématique et rigoureux en s'appuyant sur différentes méthodes (raffinement, choix, revue, etc.). Durant cette phase, sur la base des besoins établis et vérifiés, la MOA (ou son assistance) en collaboration avec la MOE décident de ce qui peut être conservé ou mis à jour dans le SI actuel de la PME (système « *as-is* ») si celui-ci existe partiellement, de ce qui doit être prévu pour former le système attendu (système « *to-be* ») et de ce qui sera effectivement développé en tenant compte des contraintes projet (système « *to-implement* »). Ces choix conduisent à l'élaboration du référentiel des exigences qui sont écrites, typées, documentées et structurées. En parallèle, la vérification continue de la qualité attendue de ces exigences est nécessaire pour garantir qu'elles sont bien formées, bien écrites, et éliminer les possibles défauts cités précédemment (ambiguïtés, omissions ou autres défauts inhérents aux besoins).

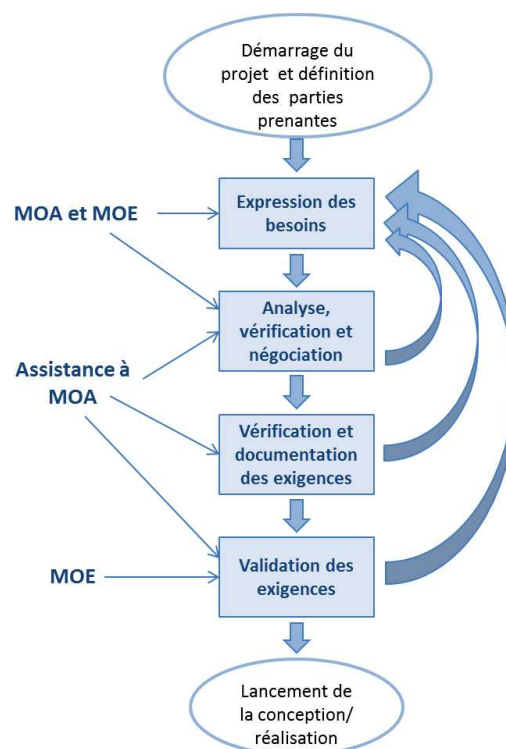


Figure 2. Parties prenantes intervenant habituellement dans un processus d'IBE.

5 - Validation des exigences. Il s'agit alors de valider les exigences par rapport aux besoins des parties prenantes. Il est nécessaire ici de se référer au métier de l'entreprise et aux domaines évoqués dans les besoins pour déterminer la pertinence des exigences vis-à-vis du monde réel tel que perçu par chacune des parties prenantes, dont la MOA et la MOE.

Ce processus n'est pas séquentiel. Des retours en arrière et des itérations apparaissent, par exemple, lorsque la résolution d'un conflit implique des modifications ou que la vérification des exigences implique des corrections. La Figure 2 met en évidence ces itérations et montre l'intervention des différentes parties prenantes (cas où la MOA est assistée) dans le processus d'IBE. Il est également important de garder une trace de « qui a dit quoi » et des modifications qui ont été apportées afin de pouvoir justifier et argumenter de la qualité résiduelle

du référentiel des exigences. Ce référentiel est ensuite utilisé comme entrée pour les phases suivantes du projet.

1.3 Conclusion : problématique RESULIS et objectifs des travaux

L'ingénierie des besoins et des exigences est une phase décisive de tout projet de développement logiciel. Son résultat est en effet le cahier des charges du ou des applicatifs attendus, document utilisé tout au long du cycle de vie du projet d'informatisation. En plus de leur intérêt économique, car la correction *a posteriori* des erreurs liées au non-respect des exigences est coûteuse une fois le système livré, les exigences ont un intérêt légal, technique, de communication entre la MOA et la MOE et de gestion de projet.

La réussite du projet d'informatisation dépend donc fortement de la qualité des exigences. Or, Le processus d'IBE requiert des ressources en termes de temps, compétences et moyens financiers que les PME ne possèdent pas nécessairement ou ne peuvent pas mobiliser. RESULIS a constaté que même s'il existe plusieurs techniques et outils pour gérer les phases d'expression des besoins, ils restent très peu utilisés dans les PME. Comme dit plus haut et précisé dans le chapitre II, cela est dû essentiellement à l'intérêt limité, à l'image fastidieuse du travail à faire et au niveau limité de compétences, liées à la définition des besoins d'informatisation, au sein des PME et c'est d'autant plus critique pour les microentreprises. La PME peut éventuellement se faire assister par des experts indépendants du prestataire MOE (Figure 2) or elle manque souvent de moyens financiers pour cela. Par conséquent, ces activités sont la plupart du temps confiées à des équipes faisant partie de la MOE. Elles sont réalisées informellement par des développeurs qui risquent de proposer leurs propres interprétations (Cao & Ramesh 2008a; Nikula et al. 2000), ce qui peut accentuer le risque d'incompréhension entre la PME qui est très peu habituée à formaliser ses besoins et les équipes de la MOE qui s'expriment souvent en termes de solution.

Notre objectif est de développer une méthode outillée permettant à RESULIS d'accompagner durant la phase d'IBE tout projet d'informatisation d'une PME. Cette méthode doit répondre aux objectifs suivants :

- Représenter la nature des systèmes à réaliser, à savoir des applications de gestion supports au système d'information de la PME. En effet, l'ingénierie des besoins et des exigences doit inclure une analyse et une représentation du « monde » dans lequel les exigences ont un sens (Lamsweerde 2009), ici l'entreprise. Or l'entreprise est un système complexe et multi vues. La méthode d'IBE doit permettre de le décrire intuitivement en présentant la situation de l'entreprise à informatiser : son fonctionnement, ses échanges avec l'environnement, son organisation interne et ses ressources.
- Considérer la difficulté des PME à mener à bien les activités d'IBE. les PME manquent de compétences, de moyens financiers, de temps et ne peuvent souvent pas se passer de leurs collaborateurs pour des formations qu'elles ne peuvent pas capitaliser.

Or, les collaborateurs de la PME sont les plus aptes à décrire et à argumenter les besoins liés à leur système d'information. Ils doivent utiliser des concepts et langages simples, intuitifs, maîtrisables, en phase avec leur langage métier et ne nécessitant pas de formation ni d'assistance permanente.

- Pallier le manque de formalisation et de moyens pour s'assurer de certaines qualités attendues d'une exigence ou d'un référentiel d'exigences. RESULIS collecte les besoins des PME par des interviews où le savoir-faire et l'expérience peuvent sembler suffisants, mais n'évitent pas les défauts inhérents à la collecte de besoins qui se reporteront ensuite sur les exigences. La détection d'erreurs et de défauts est primordiale pendant le processus d'IBE pour entreprendre des actions correctives et garantir la qualité des exigences et des livrables du projet d'informatisation par la suite.

Finalement, nous devons garder en vue le passage crucial mais difficile entre les activités d'ingénierie des besoins et des exigences et celles de conception et de réalisation. La méthode proposée doit permettre de faire passer aux équipes de la MOE des exigences exploitables et prévoir l'interopérabilité des langages et outils de la MOA et de la MOE pour accélérer la conception et la réalisation des applicatifs.

Le présent manuscrit de thèse expose, après la présentation de l'état de l'art, les composantes principales de la méthode proposée. Il est organisé comme suit :

Le deuxième chapitre présente un état de l'art de travaux portant sur la Modélisation d'Entreprise (ME) d'une part, où un ensemble de cadres et de méthode permettant de représenter la nature multi vues de l'entreprise est présenté. Ils nous permettent d'articuler la définition des besoins de la PME selon plusieurs vues pour tenter de maîtriser sa complexité. Nous présentons, d'autre part, des travaux se spécialisant plus précisément dans l'expression des besoins et des exigences et nous analysons leur utilisabilité en tenant compte des objectifs et des hypothèses présentés dans le premier chapitre. Enfin nous recensons un ensemble de techniques de vérifications et de validation pouvant être utilisées pour garantir le passage de modèles d'exigences vérifiées et validées à la maîtrise d'œuvre,

Le troisième chapitre présente notre proposition relative à la première composante constituant la méthode proposée à savoir les **concepts** et les **relations** qui les lient. Ils seront manipulés par les collaborateurs de la PME pour définir leurs besoins d'informatisation par le biais de langages de modélisation. Ces *construits* représentent de fait un modèle conceptuel permettant d'analyser le modèle des besoins et de définir des règles de modélisation pour vérifier certaines des qualités nécessaires des exigences telles que la cohérence, la non-reformulation et la complétude.

Le quatrième chapitre est consacré au choix et à la définition de **langages de modélisation** qui formalisent la manière de manipuler, instancier et renseigner les construits présenté au troisième chapitre. Ces langages supportent l'activité d'expression des besoins mais aussi de

vérification et de validation. Nous nous sommes attachés dans ce chapitre à détailler la notation de ces langages de modélisation et à formaliser leur construction.

Le cinquième chapitre présente enfin **la démarche** d'ingénierie des besoins et des exigences liée à l'informatisation de la PME. Où les concepts et langages sont mis en œuvre. La démarche proposée s'articule en un ensemble d'étapes allant de l'identification des parties prenantes à la validation des exigences. Cette démarche est supportée par un ensemble d'outils développés et intégrés dans un prototype démonstrateur.

■ Chapitre II : État de l'art

2.1 Introduction

Nous nous intéressons dans ce chapitre à un ensemble de cadres, méthodes, pratiques et langages qui nous permettraient de (i) éliciter les besoins des collaborateurs des PME quant à la réalisation de nouvelles applications de gestion supports au système d'information et ce en représentant la nature multi vues de l'entreprise, (ii) favoriser le passage de modèles d'exigences vérifiées et validées à la maîtrise d'œuvre, pour que cette partie prenante puisse entamer avec confiance les étapes de conception, de réalisation puis d'intégration des applications attendues dans l'entreprise.

Ce chapitre présente un état de l'art de travaux essentiels pour les fondements de notre travail de recherche. Les domaines couverts sont, d'une part, la modélisation d'entreprise ainsi que les méthodes et langages utilisés en ingénierie des besoins et des exigences, et d'autre part, les techniques de vérification et de validation des exigences, analysés et discutés en tenant compte des objectifs et des hypothèses présentés dans le chapitre précédent.

2.2 Comprendre la PME : apports de la Modélisation d'Entreprise

Tout le monde s'accorde sur la vision systémique de l'entreprise. (LeMoigne 1977) y identifie trois sous-systèmes : le système de transformation opérationnel qui délivre les produits et/ou les services de l'entreprise donc sa valeur ajoutée en termes de production de biens et de services, le système de pilotage qui est divisé en centres de décision délivrant les objectifs, les plans et les ordres d'exécution et le système d'information constitué de l'ensemble des informations formelles et informelles qui transitent dans l'entreprise et des moyens et ressources qui permettent cette transmission. Le système d'information, comme vu dans la chapitre I, ne se limite donc pas aux applications informatiques utilisées par l'entreprise. Il englobe de plus un ensemble de ressources humaines, processus métiers, bases de données et ressources matérielles mis en place pour accompagner l'activité de l'entreprise.

Les besoins concernant le système d'information de l'entreprise doivent être collectés et cela nécessite de prendre en compte et de formaliser diverses informations concernant les domaines d'activités de l'entreprise, les relations et échanges avec ses partenaires, son organisation interne, son fonctionnement à des niveaux de détail de plus en plus précis lorsque cela s'avère nécessaire, la distribution des responsabilités et les informations et les ressources de l'entreprise. Ceci permet de comprendre de façon plus globale la structure et le fonctionnement de l'entreprise. Ainsi, plusieurs considérations doivent être prises en compte, faisant de l'entreprise un système complexe et multi vues (Kuras & White 2005). La complexité est notamment liée à la caractéristique sociotechnique de l'entreprise puisqu'elle représente un ensemble d'interactions entre des sous-systèmes de nature hétérogène dans lesquels le facteur humain joue un rôle prépondérant. Plus qu'un ensemble de ressources, les collaborateurs de l'entreprise représentent un ensemble de compétences qui induisent des modes de gestion, des décisions, des conflits, etc. qui conditionnent de fait le fonctionnement et les usages de l'entreprise et font d'elle un système sociotechnique complexe.

Les méthodes de Modélisation d'Entreprise (ME) (Berio & Vernadat 1999; Bernus & Nemes 1997, Doumeingts et al. 1992; Williams 1994) abordent la complexité inhérente du système d'entreprise. Elles proposent des langages et des cadres de travail pour bâtir des modèles et faciliter la compréhension puis l'analyse de la structure et du comportement de l'entreprise et ce dans le but d'améliorer sa performance et sa productivité.

La modélisation d'entreprise peut être définie comme « *l'art d'externaliser les connaissances qui peuvent apporter une valeur ajoutée à l'entreprise, ou qui doivent être partagées* » (Vernadat 1999). Pour représenter et manipuler ces connaissances, les modèles jouent un rôle crucial, en offrant un filtre de la réalité, certes imparfait mais devant être suffisant et rendu nécessaire pour raisonner, décider et agir en entreprise, c'est-à-dire élaborer divers types de projets visant à faire évoluer l'entreprise. Ces projets peuvent se focaliser sur des problématiques d'analyse de flux, de conception ou de re-conception d'un SI, de gestion de l'activité et des ressources, de simulation, d'évaluation de la performance, ou d'optimisation (Berio & Vernadat 1999). Dès la fin des années 70, la modélisation d'entreprise a donc essentiellement répondu à un problème d'intégration de l'information et des flux matériels entre les différentes parties de l'entreprise, suite à l'introduction puis l'utilisation massive de systèmes automatisés de plus en plus complexes (Bernus et al. 2014). Elle a évolué pour viser l'intégration de la structure, du comportement et de l'information en entreprise afin d'améliorer son agilité (Fox & Gruninger 1998). Actuellement, c'est une discipline largement utilisée dans des projets d'évolution visant à s'aligner avec les besoins changeants du marché, et améliorer continuellement la qualité des produits et services, et ce en répondant aux besoins suivants :

- Comprendre, analyser et maîtriser la structure et le fonctionnement de l'entreprise;
- Prévoir le comportement et les performances des processus avant leur mise en œuvre ou pendant un projet de réingénierie;
- Choisir et justifier les meilleures alternatives de réalisation et d'implantation des processus;
- Identifier les risques d'implantation à gérer;
- Bâtir une vision commune du fonctionnement de l'entreprise et la communiquer facilement aux collaborateurs et aux partenaires.

Nous n'avons pas l'ambition de présenter dans la suite une liste exhaustive des problématiques et travaux proposés dans la littérature concernant la ME mais plutôt de présenter ses principes, ses méthodes et ses principaux cadres d'architecture de référence, qu'ils soient normatifs ou non, mais considérés comme pertinents pour cette étude.

Nous nous plaçons ici dans le cadre d'un projet de mise en œuvre d'applications supports au système d'information et nous utilisons alors la modélisation d'entreprise comme un moyen de compréhension de ce qu'est l'entreprise et de ce qu'elle veut être. Plus particulièrement, nous nous intéressons à la possibilité de construire des modèles de manière à faciliter le recueil des besoins.

2.2.1 Modèle d'entreprise

Un modèle est le résultat des efforts de collecte, de conception et de modélisation pour représenter un sujet. D'après (Vernadat 1999), il représente une abstraction de la réalité exprimée dans un formalisme défini par des concepts de modélisation adaptés au besoin de l'utilisateur. Ces concepts de modélisation forment les éléments du langage de modélisation.

En modélisation d'entreprise, le modèle est le moyen d'« extérioriser, construire et partager » des connaissances sur l'entreprise (Fischer 2005). Il est donc très utile pour permettre la communication entre les parties prenantes d'un projet. Le modèle est aussi et surtout un moyen de raisonnement et de prise de décisions puisqu'il a pour objectif de « *formaliser tout ou une partie de l'entreprise dans le but de comprendre ou d'expliquer une situation existante ou pour réaliser puis valider un projet conçu* » (Braesch et al. 1995).

2.2.2 Principes de la Modélisation d'Entreprise

Les méthodes, cadres d'architecture et outils de modélisation d'entreprise ont été développés comme support au cycle d'ingénierie et de modélisation des entités de l'entreprise. La maîtrise de la complexité apparente ou inhérente de l'entreprise a été traitée, dans la plupart des méthodes, par une décomposition selon des points de vue structurels (organisation, ressource, information) et fonctionnels. Plusieurs approches basées sur la mise en avant des processus de l'entreprise sont effectivement aujourd'hui reconnues pour permettre de comprendre, d'améliorer et de piloter les activités entre collaborateurs et partenaires pour aboutir à un résultat correspondant aux objectifs de l'entreprise. De nombreux auteurs arguent de fait que la maîtrise du système d'information d'une entreprise passe par la maîtrise ses processus. Ces derniers sont perçus traditionnellement comme la source principale des besoins fonctionnels, lorsqu'il s'agit de développer des applications supports au SI de l'entreprise (Brocke & Rosemann 2010). Cependant, cette vision fonctionnelle (vision processus) n'est pas suffisante pour saisir la totalité du système d'information de l'entreprise. En effet, pour pouvoir représenter les autres points de vue de l'entreprise au moyen de modèles, d'autres langages de modélisation sont nécessaires. Selon (Vernadat 2001), les concepts que ces langages permettent de manipuler doivent être définis en prenant en compte les principes suivants :

- Les concepts du langage sont en nombre fini, et leur syntaxe et sémantique sont fixées, figeant ainsi tout problème d'interprétation.
- Séparation entre les processus et les ressources : l'entreprise est perçue comme un entrelacs de processus interdépendants mais séparés et non couplés avec un ensemble de ressources.
- Séparation entre le comportement et le fonctionnement de l'entreprise : il faut pouvoir séparer ce qui est fait (le fonctionnement) de l'ordre selon lequel c'est fait (le comportement).
- Séparation entre les ressources et les unités organisationnelles : les unités organisationnelles (services, départements...) sont définies dans le but de représenter la

structure de l'organisation. Les ressources (humaines et matérielles) doivent être définies séparément puisqu'elles peuvent être allouées à différentes unités organisationnelles.

Ces principes ont été définis dans un souci d'interopérabilité entre les langages proposés en modélisation d'entreprise dans le cadre du projet UEML (Vernadat 2001). Ils garantissent l'évolution et la maintenabilité des modèles produits lors des projets concernés par la modélisation d'entreprise, puisque des éléments du modèle peuvent être mis à jour sans pour autant modifier tout le modèle.

2.2.3 Cadres et méthodes en Modélisation d'Entreprise

L'émergence de méthodes de modélisation d'entreprise a accompagné l'introduction de l'automatisation complète des procédés de fabrication à la fin des années 70 afin d'intégrer le flux d'informations nécessaire ou généré par ces nouveaux systèmes de production au reste des systèmes de l'entreprise (Bernus et al. 2014). Nous pouvons remonter jusqu'aux milieux des années 70 pour identifier les premières méthodes de modélisation d'entreprise. Les développements aux États-Unis qui ont donné lieu à des langages de modélisation fonctionnelle (SADT⁵ (Marca & McGowan 1987), SSAD⁶ (Gane & Sarson 1979)) ou de modélisation de l'information (modèle Entité Relation (Chen 1976), *Data Flow Diagrams* (Bruza & Weide 1989)) (Vernadat 2001). Dans les années 80, des méthodes spécifiques ont été proposées pour modéliser de larges systèmes de production en vue d'assister leur conception et simulation telles que la suite IDEF⁷ aux États-Unis (Menzel & Mayer 1998) ou la méthode GRAI en Europe (Chen et al. 1997). La ME est passée d'une vision orientée activités (ou fonction) à une vision orientée processus comme préconisé par le Consortium ESPRIT « AMICE »⁸ qui a conduit au développement de la méthode CIM-OSA (Zelm et al. 1995) qui supporte la modélisation orientée processus.

La recherche en France a également joué un rôle non négligeable dans le développement de méthodes en ME. Nous pouvons citer par exemple : GRAI (Doumeingts 1984), MECI (Pourcel & Gourc 2002) et GIM (Doumeingts et al. 1992) qui est élaborée autour de GRAI dans le cadre de projets ESPRIT. Les années 90 ont marqué l'avènement de méthodes associées au déploiement d'ERP (ARIS (Scheer 1994)) et de systèmes de gestion du workflow (WPD (Junginger 2000)).

Nous allons, dans la suite, considérer plus particulièrement les méthodes CIMOSA, GIM et PERA qui ont été retenues dans des projets d'unification tels qu'UEML (Vernadat 2001) et

⁵ *Structured Analysis and Design Technique.*

⁶ *Structured System Analysis Design.*

⁷ *ICAM (Integrated Computer Aided Manufacturing) Definition.*

⁸ Programme cadre de Recherche et Développement de la Commission Européenne (programme Esprit : Project 418, Open CAM System) (AMICE. 1993) qui a largement permis de développer et de diffuser les outils et méthode de ME dans le monde industriel.

GERAM (Bernus & Nemes 1997) et qui ont fortement influencé le développement des standards CEN/ISO 19439 (ISO/DIS 2002) et CEN/ISO 19440 (ISO/DIS 2004) présentés plus loin.

2.2.3.1 CIM - Open System Architecture

CIM-OSA (*Computer Integrated Manufacturing, Open System Architecture*), littéralement : architecture système ouverte pour l'intégration des systèmes de production « informatisées », est une architecture et une méthode dont l'objet principal est de permettre la modélisation orientée processus d'entreprises industrielles (Zelm et al. 1995; Bernus & Nemes 1997) et qui peut être appliquée à tout type d'entreprise (Berio & Vernadat 1999). Son objectif est de proposer un environnement support pour l'intégration des opérations des systèmes de production. Ceci est réalisé en se basant sur l'usage des modèles (Fox & Gruninger 1998). CIM-OSA propose à la fois :

- **Un langage de modélisation** proposant des concepts pour couvrir les différents aspects de l'entreprise et articulés autour de quatre vues de modélisation (Figure 3) : vue fonctionnelle, vue informationnelle, vue des ressources et vue organisationnelle. Comme CIM-OSA propose une modélisation orientée processus, la vue fonctionnelle est au cœur des concepts. C'est donc une approche modulaire et hiérarchisée, où l'entreprise est décomposée en domaines décrits à travers des processus de domaine, détaillés enfin en processus métiers. Ces derniers sont un enchaînement d'activités détaillées en opérations fonctionnelles. L'activité est le lieu d'utilisation des ressources et de l'information.
- **Un processus de modélisation** : CIM-OSA définit une démarche qui couvre tout le cycle du projet l'intégration des systèmes de production en entreprise (*Computer Integrated Manufacturing*). Celui-ci couvre les phases suivantes : définition des besoins, conception, mise en œuvre et maintenance (Zelm et al. 1995). La première phase (RDM : *Requirements Definition Modelling*) est celle de spécification des besoins définie grâce aux concepts de modélisation proposés ci-dessus. Elle reprend la décomposition fonctionnelle hiérarchisée et l'applique à la partie de l'entreprise (domaines) à modéliser en représentant ce qui doit être fait sans introduire de contrainte de conception et d'implémentation.
- **Un cadre d'architecture** où les phases du processus de modélisation représentent un des trois axes de l'architecture de référence nommée usuellement le « cube CIMOSA » (plus tard enrichi par GERAM). Un deuxième axe reprend les quatre vues de la modélisation (fonctionnelle, informationnelle, des ressources et organisationnelle). Enfin, le troisième axe de spécialisation permet de transformer le modèle générique en un modèle spécifique pour correspondre au modèle particulier d'une entreprise.

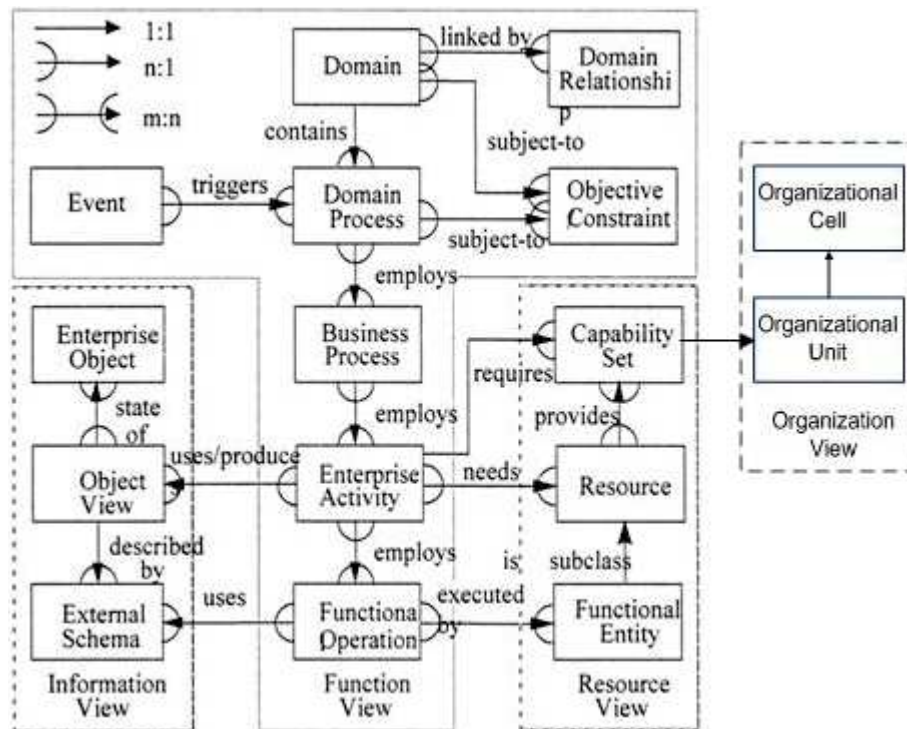


Figure 3. Concepts des vues de la modélisation selon CIM-OSA (Berio & Vernadat 1999).

2.2.3.2 GRAI Integrated Methodology

La méthode GIM (*GRAI Integrated Methodology*) (Doumeingts et al. 1992) est la proposition élaborée par le laboratoire LAP GRAI (Laboratoire d'Automatique et de Productique) de l'Université Bordeaux, dans le cadre des projets européens ESPRIT Open CAM, IMPCAS et FLEXQUAR. La méthode a été conçue pour fournir des outils d'analyse et de conception de systèmes complexes liés à un contexte industriel. Elle s'appuie principalement sur les éléments suivants :

- **Un modèle conceptuel**, qui représente la structure de base des entreprises principalement industrielles, structurée en trois sous-systèmes : le système physique de production, le système de décision qui gère et contrôle le système physique et le système d'information, lieu de circulation et courroie de transmission de l'information entre les systèmes de décision et physique.
- **Différents langages de modélisation** : La méthode GIM utilise le formalisme entités relations ou des diagrammes de classes UML pour la vue informationnelle et le formalisme IDEF0 pour la vue physique et la vue fonctionnelle. La méthode suit une démarche originale reposant sur un modèle du système décisionnel de l'entreprise, appelé « grille GRAI » qui définit des *centres de décision* relatifs à des fonctions de l'entreprise. Un centre de décision est détaillé en un réseau d'activités différenciant les activités de décision et d'exécution. La grille GRAI conduit à différencier les échanges de décision qui représentent une dépendance fonctionnelle entre centres de décision des simples échanges informationnels (Millet 2008).
- **L'approche structurée de GIM** : elle permet de faire évoluer un système par étapes, d'un état existant vers un état cible, et utilise un ensemble d'indicateurs permettant

d'évaluer différentes qualités (cohérence, fiabilité, etc.). Elle propose deux méthodes de conception orientée utilisateur et orientée technologie à partir d'une première phase d'analyse des besoins. La première transforme les besoins utilisateurs en spécifications exprimées en termes de fonctions, informations, décisions et ressources. La deuxième transforme les spécifications utilisateurs en spécifications techniques exprimées en termes d'information et de composants technologiques nécessaires à la mise en œuvre du système.

Dans la littérature, les modèles de GRAI ont été utilisés pour spécifier des exigences fonctionnelles d'une PME, quant à l'acquisition d'un nouvel ERP (Lacombe et al. 2011), pour décrire les processus de l'entreprise, les décisions prise lors de l'exercice de son métier et pour montrer les liaisons existantes entre les vues décisionnelle, informationnelle et physique.

2.2.3.3 PERA

L'architecture PERA (*Purdue Enterprise reference Architecture*) (Williams 1994) est une autre architecture favorisant l'intégration de l'entreprise en structurant les processus de développement et d'implémentation en entreprise. Cette architecture, comme la plupart des méthodes et architectures en modélisation d'entreprise, considère l'entreprise selon différents points de vue. La description des fonctions de l'entreprise est axée selon deux flux principaux : un flux informationnel, qui représente le flux de décision, de contrôle et d'information et un flux matériel, initié par le flux physique de production ou par les demandes des clients. Ces flux sont implémentés par des activités humaines ou informatisées.

2.2.4 Efforts d'unification et de normalisation

Devant ces nombreuses propositions en modélisation d'entreprise, des projets et groupes de travail ont été initiés par la communauté afin d'homogénéiser les différents modèles et cadres existants. Nous pouvons citer notamment le projet UEML (Vernadat 2001) et GERAM. Il faut également souligner que la modélisation d'entreprise fait l'objet d'un groupe de standardisation ISO TC184 SC5 WG1 dénommé « *Industrial-automation systems and integration, Architecture, communications, and integration frameworks, Modelling and architecture* » dont l'un des objectifs est d'unifier les pratiques en modélisation d'entreprise.

2.2.4.1 UEML

Parallèlement, au développement de méthodes de ME, plusieurs outils ont été développés depuis les années 90 pour permettre aux utilisateurs de créer et de manipuler des modèles d'entreprise tels que : ARIS Toolset, FirstSTEP, NCR Metis, Bonapart, Enterprise Modeller, PROPLAN, PrimeObject, MOOGO, CimTool, IMAGI, etc. (Vernadat 2001). Ces outils supportent des langages qui permettent de modéliser les mêmes éléments tout en étant basés sur des concepts différents qui sont de fait non interopérables. Il a donc été proposé de définir UEML (*Unified Enterprise Modelling Language*) comme un langage commun. UEML n'est pas un nouveau langage mais un langage pivot développé dans un souci d'interopérabilité, qui assure la transformation et la mise en relation entre les langages existants. Son objectif est

de trouver un consensus dans la communauté de modélisation d'entreprise pour ce qui est des concepts à manipuler et de leurs relations. La Figure 4 représente les concepts de base proposés par ce projet, que tout langage en ME doit pouvoir représenter ou pour lesquels des correspondances existent.

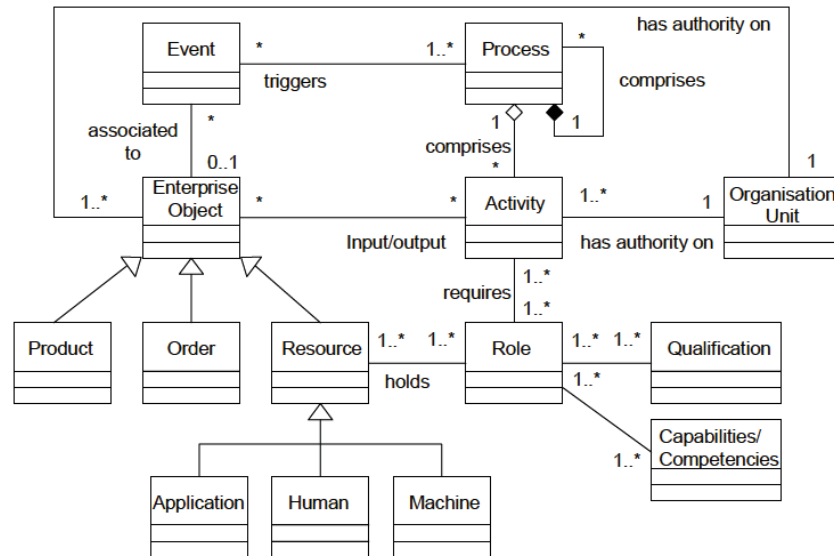


Figure 4. Concepts de base de UEML (Vernadat 2001).

2.2.4.2 GERAM

GERAM (*Generalized Enterprise Reference Architecture and Methodology*) (Bernus & Nemes 1997) est une architecture de référence établie à partir du constat que les méthodes existantes devaient être étendues pour obtenir une vision architecturale complète et une méthode qui organise les composants nécessaires en modélisation et intégration d'entreprise. En effet, une collaboration entre les fédérations IFIP (*Federation for Information Processing*) et IFAC (*International Federation for Automation and Control*) a donné lieu à un groupe de travail dont le mandat était d'examiner les approches d'architecture existantes en modélisation et intégration d'entreprise et de les généraliser (Bernus et al. 2014). Trois approches ont été retenues pour élaborer GERAM : CIM-OSA, GRAI-GIM et PERA, en raison de leur complémentarité et aussi parce qu'elles couvrent tout le cycle de vie des entités modélisées – ceci étant une des recommandations initiales du groupe de travail. Le processus de normalisation du groupe de travail IFAC/IFIP s'est poursuivi, en collaboration avec le groupe de standardisation ISO TC184 SC5 WG1, pour aboutir à l'élaboration d'un standard ISO1570 « *Requirements for Generalised Enterprise Reference Architectures and Methodologies* » (ISO15704 2000).

GERAM est conçue comme une « boîte à outils » de concepts. Elle peut être appliquée à tout type d'entreprise et n'impose pas d'outils ni même de langages de modélisation ou de méthodes particulières de modélisation. GERAM s'articule en un ensemble de recommandations concernant les méthodes d'ingénierie d'entreprise (EEM), les langages de modélisation d'entreprise (EML), les outils associés (EET), etc. le tout employé dans le cadre de référence proposé (GERA – *Generalized Enterprise Reference Architecture*) (ISO15704 2000).

GERA est donc l'architecture de référence fournie par GERAM schématisée dans la Figure 5. Fortement influencée par le cadre d'architecture de CIM-OSA, elle cadre les méthodes d'ingénierie d'entreprise à travers les trois axes suivants :

- **Le cycle de modélisation** : il fournit le processus de modélisation des entités d'entreprise relativement à leur cycle de vie depuis la création à la fin de vie. Ce cycle n'est pas forcément séquentiel, GERA couvre toutes les séquences possibles même itératives.
- **La généricité** : cet axe représente le processus d'instanciation des modèles génériques et partiels en modèles spécifiques. Les concepts et règles de modélisation, réutilisables dans tout contexte, résident au niveau générique. Ces derniers peuvent être utilisés au niveau partiel pour construire des modèles prédéfinis, adaptés à des typologies d'entreprise, à personnaliser au niveau spécifique avec une entreprise.
- **Les vues** : la décomposition en vues permet la visualisation et la manipulation des entités d'entreprise selon un intérêt spécifique de modélisation. Cet axe regroupe quatre types de vues :
 - **Les vues du contenu des modèles** : les vues de la modélisation d'entreprise proposées dans CIM-OSA (fonctionnelle, informationnelle, des ressources et organisationnelle) qui permettent de filtrer les modèles pour en simplifier la lecture.
 - **Les vues de l'entité modélisée** : de production / service client ou de contrôle / gestion. Ces vues reprennent les flux fonctionnels de PERA.
 - **Les vues de l'implémentation de l'entité modélisée** : manuelle ou automatisée.
 - **Les vues de la manifestation physique des entités** : software (code / texte / connaissance), hardware (PC, personne, machine).

2.2.4.3 Le cadre de normalisation ISO 19439 et ISO 19440

La norme ISO 15704 est la version standardisée de l'architecture GERAM. Elle définit les exigences attendues d'une méthode de modélisation d'entreprise pour être complète. Pour répondre à ces recommandations, le groupe de travail ISO TC184 SC5 WG1 a élaboré, entre autres, les normes :

- CEN/ISO 19439 *Framework for Enterprise Modelling* (ISO/DIS 2002) : une norme qui standardise une partie de l'architecture de référence GERA (composante de GERAM) en proposant un cadre unifié pour structurer les entités considérées et modélisées par une méthode d'ingénierie d'entreprise. Elle est structurée en trois axes (Figure 5) : les phases du cycle de la modélisation, les niveaux de généricité des modèles et les vues de la modélisation.

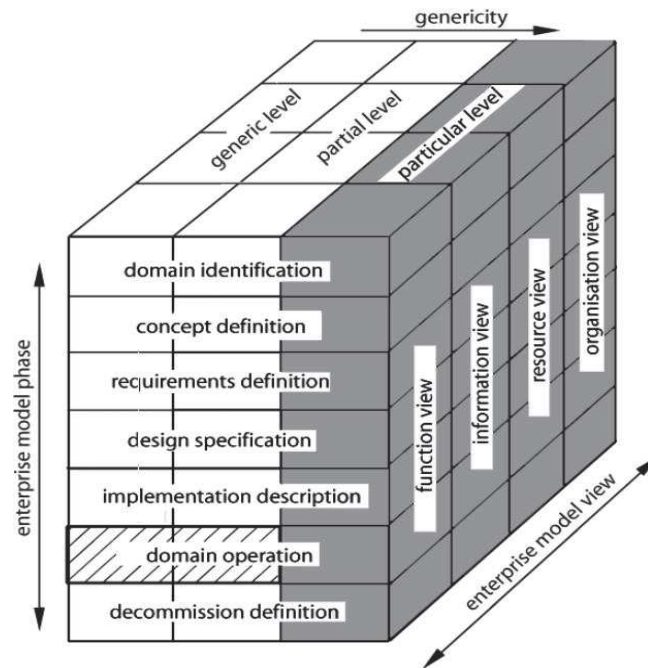


Figure 5. Cadre de modélisation ISO19439 (ISO/DIS 2002).

- CEN/ISO 19440 Concepts for Modelling (ISO/DIS 2004) : cette norme détaille l'axe des vues de la modélisation du cadre d'architecture de l'ISO 19439 et définit 14 constructs de modélisation (Figure 6), proches des concepts proposés par le langage de modélisation de CIM-OSA. Elle rejoint aussi les concepts de base d'UEML. Ces constructs reprennent la vision hiérarchisée et orientée processus de CIM-OSA, dans laquelle l'entreprise est considérée à la fois comme un ensemble de processus métiers concurrents pour réaliser les objectifs de l'entreprise et comme une fédération d'entités fonctionnelles (humains, machines et applicatifs) traitant les objets de l'entreprise pour exécuter les étapes des processus métiers.

Le cadre représenté par les normes ISO 19439 et ISO 19440 peut être considéré comme une base structurante pour tout travail d'ingénierie couvrant totalement ou partiellement le cycle de modélisation d'entités liées à l'entreprise : le cadre ISO 19439 représente les considérations nécessaires au pilotage d'une méthode d'ingénierie, et le métamodèle ISO 19440 représente les concepts à modéliser.

2.2.5 Synthèse : verrous constatés

La modélisation d'entreprise nous offre un cadre scientifique et normatif pour représenter le fonctionnement et la structure du système d'entreprise. «*Enterprise Modeling is used to represent the structure, behavior, components and operations of a business entity to understand, (re)engineer, evaluate, optimize and even control business operations and performance*» (Vernadat, 2004). Dans le cadre de l'ingénierie des besoins et des exigences, nous proposons de suivre les principes de la modélisation d'entreprise pour représenter les entités de l'entreprise afin de les comprendre et de faire l'ingénierie des exigences qui leur sont associées.

Fonctionnelle

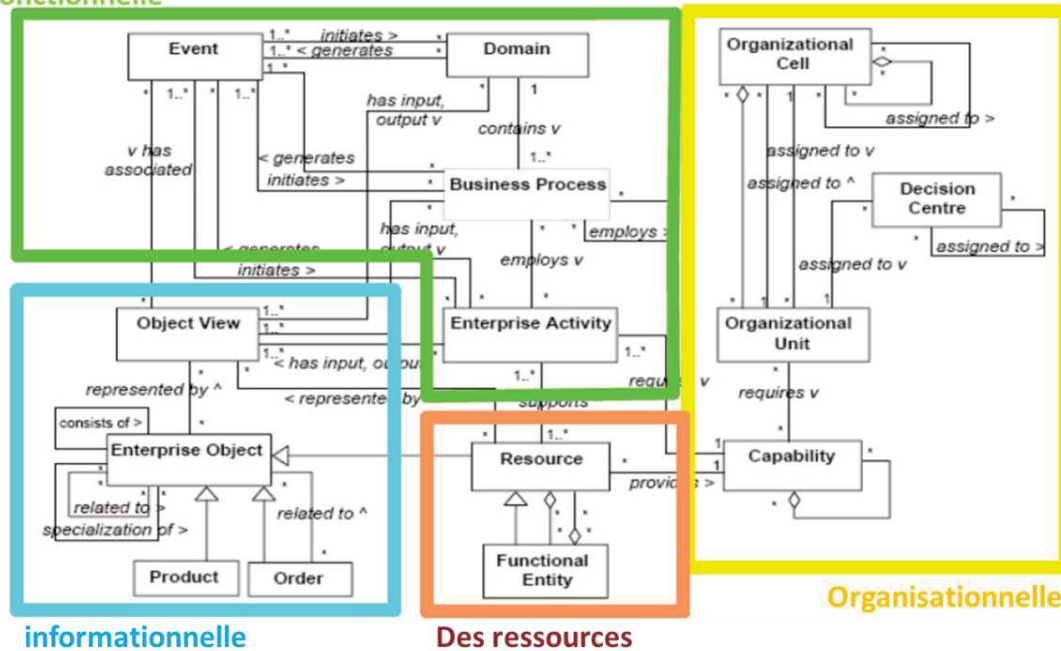


Figure 6. Concepts ISO19440 et les quatre vues de la modélisation (ISO/DIS 2004).

Les normes ISO 19439 et ISO 19440 recommandent que les méthodes d'ingénierie d'entreprise, et donc par implication les méthodes d'ingénierie des besoins et des exigences (IBE) qui leur sont associées, proposent :

- une démarche de modélisation pour prendre en charge une ou plusieurs phases du cycle de vie des entités modélisées en tenant compte des besoins des phases avalées,
- des concepts génériques ou partiels pouvant couvrir les quatre vues de la modélisation,
- un ou des langages pouvant manipuler ces concepts et les instancier au niveau spécifique de chaque entreprise.

Nous remarquons que les activités de l'ingénierie des besoins et des exigences couvrent les trois premières phases l'axe « cycle de la modélisation » (*Enterprise model phase*) du cadre pour la modélisation ISO 19439 (Figure 5) :

- l'identification du domaine : où sont définies les entités de l'entreprise à étudier,
- la définition des concepts : où est définie une vision macro de l'entreprise : sa mission, sa stratégie, etc.,
- la définition des exigences de l'entreprise, correspondant aux exigences à prendre en compte dans les phases avalées de conceptions et d'implémentation.

De plus, des méthodes de modélisation d'entreprise tel que GRAI et CIM-OSA sont utilisées pour la spécification des exigences liées au fonctionnement ou à la structure d'une entreprise (Lacombe et al. 2011; Grangel et al. 2007; Zelm et al. 1995). Toutefois, l'IBE reste une discipline indépendante, où l'on ne fait pas souvent référence à la modélisation d'entreprise. Ceci est sans doute dû au fait qu'elle est commune à plusieurs disciplines d'ingénierie.

Durant ces phases, ce sont les concepts des vues de la modélisation (Figure 6) qui sont manipulés. Les besoins et les exigences sont donc exprimés en termes de processus, activités, objets d'entreprise, ressources, unités organisationnelles, etc. Ainsi les langages associés à une méthode d'IBE pour la réalisation d'applications supports au SI doivent couvrir les vues fonctionnelle, informationnelle, organisationnelle et des ressources. Nous estimons que l'utilisation des principes de modélisation d'entreprise dès le début d'un projet d'informatisation, permet de maîtriser la complexité du modèle de l'entreprise et d'assurer la cohérence entre les points de vue étudiés. Dans la section suivante, nous nous intéressons plus particulièrement aux méthodes et aux langages utilisés pour l'ingénierie des besoins et des exigences (IBE) et analysons la correspondance des concepts proposés par celles-ci avec les concepts des vues de la modélisation. Ces derniers restent accessibles aux collaborateurs de la PME. Donner la possibilité aux collaborateurs de se référer à ces concepts permettrait, entre autres, de lever l'ambiguïté qui peut être associée aux besoins et de partager avec la MOE une vision du modèle de la PME telle qu'elle est perçue par ses collaborateurs. Néanmoins, le métamodèle proposé par l'ISO 19440 détaillant les concepts de la modélisation ne comporte pas de concepts pour manipuler, partager et garder une trace des concepts spécifiques à la phase de recueil des besoins et de définition d'exigence tels que : les parties prenantes qui émettent les besoins, leur rôles et positions dans le projet d'informatisation ou dans la structure de l'entreprise et les processus métiers. Ceci implique un manque de traçabilité aussi bien au niveau des versions des différentes déclarations des parties prenantes, qu'entre les besoins initiaux et les exigences validées. Ceci est pourtant crucial pour argumenter et juger de certains choix, et alimenter ainsi le processus de décision.

2.3 L'IBE pour comprendre et formaliser le projet d'évolution de la PME

L'ingénierie des besoins et des exigences est une discipline qui prend ses racines dans la modélisation fonctionnelle avec le langage SADT (Ross & Schoman 1977), basée sur le principe de points de vue. Elle propose plusieurs modèles traitant chacun un point de vue distinct (datagramme, acti-gramme) (Lamsweerde 2000). D'autres méthodes orientées point de vue ont vu le jour tels que CORE (Mullery 1979), Viewpoint Resolution (Leite 1989), VOSE (Finkelstein & Kramer 1992), VORD (Kotonya & Sommerville 1996). L'IBE a été reconnue comme une discipline indépendante au milieu des années 80 (Lamsweerde 2009). Par la suite, les propositions ont évolué vers des méthodes plus complètes couvrant plus largement le processus d'IBE, tout en restant multi-vues. Deux catégories de méthodes se sont distinguées :

- Les méthodes orientées buts, ou méthodes prescriptives, ayant comme concept de base l'objectif justifiant le besoin où l'ensemble des besoins est défini par raffinement de buts : i^* (Chung et al. 2012), GBRAM (Antón 1996), KAOS (Lamsweerde et al. 1991), GRL (ITU-T 2008), etc.
- Les méthodes orientées scénarios où l'on s'intéresse à la séquence d'interactions entre le système envisagé et son environnement pour réaliser une activité : *Scenario-based Requirements Analysis* (Sutcliffe 1998), UCM (ITU-T 2008).

Il existe aussi des méthodes mixtes reposant sur le couple (but, scénario) (Ecritoire (Rolland 2003), URML (Schneider et al. 2012)). Nous allons dans la suite exposer des méthodes et langages utilisés en IBE. Nous ne souhaitons pas faire ici un inventaire des langages existants, mais plutôt d'identifier les concepts représentés par ces langages et évaluer leur accessibilité et leur adéquation par rapport aux besoins et usages de collaborateurs dans une PME.

Les approches de génie logiciel de conception et de développement d'application telles que UP (Jacobson et al. 1999), 2TUP (Roques & Vallée 2011), RUP (Kruchten 2004), etc. préconisent également une première étape de capture des besoins, où il s'agit le plus souvent de décrire l'utilisation et les fonctionnalités de la solution informatique. Ces méthodes sont principalement adaptées aux phases de conception de solution en utilisant principalement le langage UML et en aboutissant à des modèles orientés objet. Elles supposent souvent, comme dans 2TUP (Roques & Vallée 2011), qu'un recueil initial des besoins est fait au préalable. Nous ne considérons donc pas ces méthodes et analysons des langages et méthodes utilisés précisément pour le recueil des besoins et la définition des cahiers des charges.

2.3.1 Analyse comparative de langages de modélisation en IBE

Nous analysons ici des langages proposés dans la littérature pour spécifier les besoins et les exigences par le biais de la modélisation. En effet, pour apporter plus de précision aux méthodes informelles habituellement utilisées, dites observationnelles (analyse sociale, observation, analyse de protocole...) ou conversationnelles (interview, atelier, groupes de discussion, brainstorming...) (Al Mrayat et al. 2013), des langages plus formalisés ont été proposés pour définir les besoins en utilisant des modèles souvent avec une syntaxe concrète graphique. Nous avons considéré trois critères pour classer et comparer ces langages :

- **L'orientation est le concept de base** : l'orientation définit les aspects selon lesquels les besoins et les exigences sont considérés et la façon avec laquelle ils sont modélisés et décrits. Ceci influe grandement sur l'accessibilité du langage car cela implique pour ses utilisateurs une façon de raisonner qui peut être plus ou moins intuitive.
- **Les types d'informations représentées** : suivant les recommandations du cadre de normalisation ISO 19439 et ISO 19440, nous considérons ici les informations définies dans les quatre vues de la modélisation (fonctionnelle, informationnelle, des ressources et organisationnelle). Nous distinguons pour chaque lan-

gage les concepts qui sont susceptibles de représenter la même information que les concepts de la norme ISO 19440 pour identifier les points de vue de modélisation couverts par le langage, ou toute autre information adressée par le langage.

- **Les utilisateurs cibles** : les langages proposés spécifient les utilisateurs auxquels ils sont destinés. Ce critère est directement influencé par l'accessibilité du langage. Plus le langage est détaillé et nécessite des capacités d'abstraction spécifiques, moins il est accessible à des utilisateurs non-initiés ou formés à ce langage.

2.3.1.1 RDM

RDM (*Requirement Definition Model*) est la première étape du processus de modélisation proposé par l'architecture CIM-OSA (Zelm et al. 1995), concernée par la spécification des besoins au travers des connaissances préliminaires sur les fonctionnalités et l'organisation de l'entreprise. Ce modèle reprend la décomposition fonctionnelle hiérarchisée proposée au niveau du langage de modélisation de CIM-OSA. Il s'agit de décomposer la partie de l'entreprise à modéliser (*domaines*) en processus et durant la phase d'analyse comportementale (BP 1.2 du processus RDM – Figure 7). A partir de cette décomposition, les activités sont détaillées et les informations, ressources et structures organisationnelles impliquées dans ces processus sont définies.

Appartenant à la méthode CIM-OSA qui a fortement influencé les normes ISO 19439 et ISO 19440, ce modèle exprime tous les besoins métiers liés aux quatre vues de la modélisation. Il est orienté processus et ses utilisateurs cibles qui sont des analystes et des ingénieurs utilisent des notations textuelles (patron et détails des attributs) et visuelles (concepts, relations, etc.) élaborées.

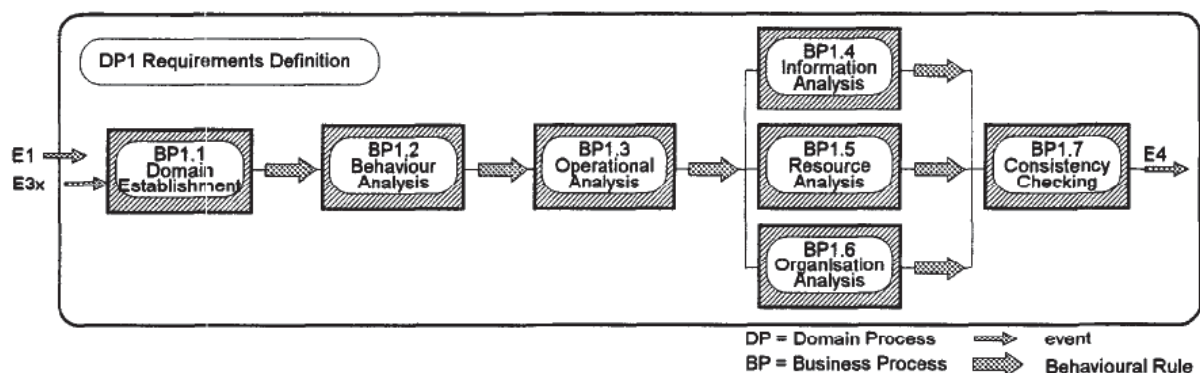


Figure 7. Processus de RDM – CIM-OSA (Zelm et al. 1995).

2.3.1.2 KAOS

KAOS (Lamsweerde et al. 1991; Lamsweerde 2001; Lamsweerde 2009) est une méthode de spécification d'exigences dotée d'un langage orienté but. Un but est défini comme une asserption prescriptive représentant un objectif que le système doit atteindre par la coopéra-

tion d'agents. KAOS est un langage multi vues. Il permet de représenter les *buts*, les *obstacles* qui vont à l'encontre de leur réalisation, les *propriétés* et les *hypothèses* faites sur l'environnement et les *exigences*. Ces concepts sont regroupés dans une vue des buts (Figure 8). Leur opérationnalisation est décrite par des *opérations* à réaliser dans le système. Les *agents* responsables de la réalisation de ces opérations sont définis dans la vue des responsabilités. Les *entités* informationnelles utilisées sont définies par un modèle à objet. De ce fait, le modèle de KAOS couvre une partie des vues fonctionnelle, informationnelle et des ressources de la norme ISO 19440 et couvre également une vues des buts. Ceci montre les relations existant entre les concepts habituellement utilisés en modélisation d'entreprise et les concepts plus communément utilisés pour faire référence aux besoins et aux exigences - ici des buts.

La démarche associée à KAOS prend en charge l'expression des buts : ils sont obtenus à partir des objectifs stratégiques, des problèmes et défauts mis en avant. Ceci est fait par abstractions si des buts sont implicites, en posant la question « pourquoi ? », ou par affinement pour avoir les sous buts en posant la question « comment ? ».

Elle propose une modélisation à l'aide d'arbres logiques (et/ou) de raffinement pour la représentation des entités et de leurs états. KAOS est un langage destiné à des experts métiers formés et secondés par des assistants à maîtrise d'ouvrage.

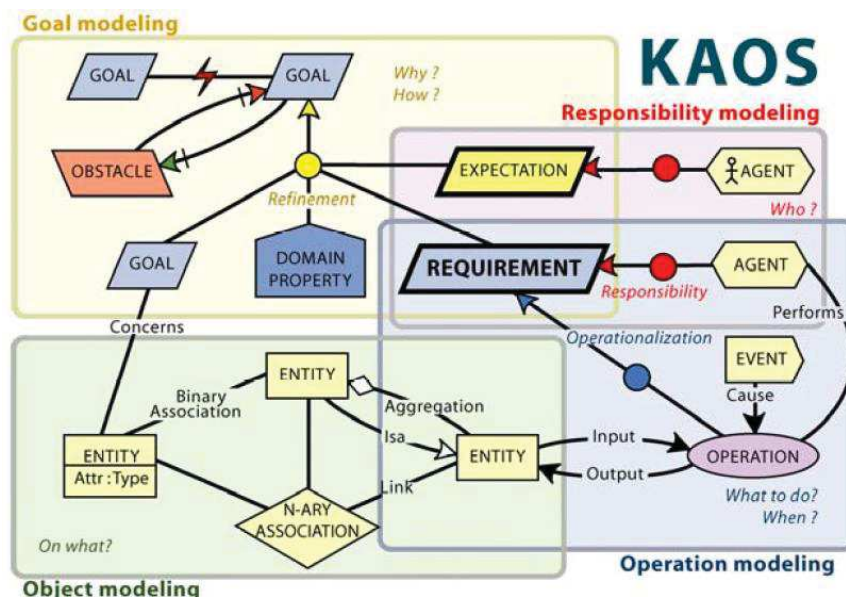


Figure 8. Concepts proposés par la méthode KAOS (Lamsweerde 2009).

2.3.1.3 UCM

UCM (*Use Case Maps*) fait partie de la norme URN (*User Requirements Notation*) pour la représentation visuelle, l'analyse et la validation des exigences développée par l'Union Internationale des Télécommunications (ITU-T 2008). UCM est un langage orienté scénario pour la définition des exigences fonctionnelles. Il permet de visualiser le fonctionnement et la structure du système en une seule représentation. Les exigences fonctionnelles correspondent aux opérations définies au niveau du scénario. Ce dernier est représenté par un

chemin formé de structures de contrôle de type : séquence, parallélisme et alternative. Ce dernier relie les *responsabilités* (opérations) localisées dans des *composants* de diverses natures : objets, processus, collaborateurs... etc. De ce fait, UCM couvre pleinement la vue fonctionnelle avec les concepts de scénario, de responsabilités et d'événements, tout en représentant les objets du système faisant partie de la vue informationnelle. Quelques concepts des vues organisationnelle et des ressources sont implicites à la description d'un scénario fonctionnel, comme le composant ressource humaine responsable de la réalisation d'une opération.

Il est utilisé par les analystes, les ingénieurs et les assistants à maîtrise d'ouvrage pour collecter et valider les cas d'utilisation et prendre des décisions de conception et d'architecture avant d'entamer la conception détaillée.

2.3.1.4 GRL

GRL (*Goal-Oriented Requirement Language*) est le langage de la norme URN (ITU-T 2008) dédié principalement aux exigences non-fonctionnelles. Il permet la modélisation et le raffinement des buts stratégiques pour déduire les exigences et les solutions alternatives. GRL prend en compte les *buts* de plusieurs *acteurs* (parties prenantes) et la façon avec laquelle ces buts peuvent être opérationnalisés. Les éléments intentionnels : *but*, *opinion*, *tâche* et *ressource* qui sont exprimés pour chaque acteur sont mis en relation grâce à des liens de dépendance, de contribution, d'opérationnalisation et de décomposition. GRL propose également un mécanisme d'annotation des liens de dépendance et de contribution (négatif/positif, total/partiel, etc.) pour évaluer la satisfaction des buts et explorer les opportunités et vulnérabilités du système. GRL ne couvre donc que partiellement les vues fonctionnelle et des ressources, et propose des concepts qu'on peut regrouper (comme dans KAOS) dans une vue des besoins et des exigences.

GRL propose une analyse stratégique des buts et de leur opérationnalisation à réaliser par des experts métiers avec l'assistance d'ingénieurs analystes.

2.3.1.5 URML

URML (*Unified Requirements Modeling language*) (Schneider et al. 2012) est un langage de modélisation préliminaire d'exigences fonctionnelles et non-fonctionnelles principalement destiné à l'ingénierie système. Il prend en compte l'expression des exigences, la modélisation de gammes de produits, l'analyse des risques et la modélisation des buts. Pour répondre à des problèmes de communication entre la MOE et MOA, URML regroupe dans un seul modèle différents points de vue du système. Un modèle URML peut être composé d'un ensemble d'éléments connectés : les principaux *processus* (sans les détailler) qui décrivent l'usage du système, les *caractéristiques* offertes par le système, les *fournisseurs de services* (humains, machines ou applicatifs), les *buts* justifiant les caractéristiques, des *menaces* et *hasards* pouvant altérer le bon fonctionnement du système, les *acteurs* intervenant dans les processus et les *exigences* permettant la réalisation des processus et l'évitement des risques. URML propose des concepts relatifs aux vues fonctionnelle, informationnelle et des res-

sources et permet de représenter d'autres informations liées à la définition des exigences fonctionnelles et non-fonctionnelles.

URML est destiné aux analystes et ingénieurs système faisant partie de la MOE, utilisé pour avoir un seul point de contact avec les décideurs de la MOA dans un projet d'ingénierie système.

2.3.1.6 Synthèse : verrous constatés

Les langages et méthodes étudiés traitent la problématique de modélisation et systématisation de la définition des besoins et des exigences. Cette activité se fait de façon différente d'un langage à un autre et est plus ou moins intuitive.

Les langages orientés but considère le but comme concept de base. Ce concept représente ce que le système doit atteindre pour solutionner un problème ou garantir une fonctionnalité. L'utilisation de ces langages par les collaborateurs de la PME nécessite inévitablement formation et assistance. En effet, il n'est pas intuitif pour les collaborateurs des PME de raisonner par but pour l'expression des besoins fonctionnels. L'expérience de RESULIS confirme qu'ils sont plus autonomes en décrivant leurs activités quotidiennes qu'en réfléchissant aux motifs justifiant ces activités. De plus, la difficulté à manipuler des concepts abstraits tels que des buts a été reconnue par des études de sciences cognitives (Benner & Feather 1993). Finalement, les buts sont plus adéquats pour la définition des besoins non-fonctionnels (Liu & Yu, 2001; Rolland et al. 1999). Bien que prometteuse, l'adoption des buts comme orientation n'est pas adéquate aux utilisateurs non-initiés, nous faisons donc le choix d'offrir aux collaborateurs de la PME le moyen de décrire leurs activités et les scénarios de leur réalisation.

La description « top-down » des principaux processus, comme définie dans RDM, n'est pas évidente pour des collaborateurs qui n'ont pas une vision globale des processus dans lesquels ils interviennent, surtout lorsque ces processus sont transversaux à différentes structures organisationnelles. De la même manière, les collaborateurs qui peuvent avoir cette vision globale, c'est-à-dire les responsables et décideurs, ne connaissent pas souvent le détail et le déroulement des activités au sein de ces processus. Nous faisons donc le choix de focaliser la description des activités sur une vision locale, où les collaborateurs ne décrivent que ce qui est réalisé au niveau de leur poste.

Selon leurs orientations et le système pour lequel les langages étudiés sont utilisés, les informations représentées sont considérées différemment :

- Une *activité* dans RDM détaille une étape d'un processus et est le lieu d'utilisation et de création d'information et de ressource. Dans KAOS une *opération* exprime l'opérationnalisation d'une exigence, à savoir sa réalisation au niveau du système. Le concept d'activité comme étape d'un processus n'est pas couvert par URML et est englobé dans UCM par la notion de *responsabilité*.

- Les *ressources* définies dans RDM comme entités réutilisables et partageables nécessaires à l'accomplissement des activités sont pour URML des *fournisseurs de service* (humains, machines ou applicatifs). Les ressources sont englobées dans KAOS dans la notion d'agent (machine, collaborateur ou applicatif) responsable de l'exécution d'une exigence du système (agent applicatif ou machine) ou d'une attente de l'environnement (agent collaborateur ou machine).
- Même si les agents et acteurs interagissant avec le système sont définis au niveau de KAOS, UCM et URML, leurs rôles et positions dans l'organisation ne sont pas représentés.

Nous remarquons que les langages étudiés font référence aux concepts pris en considération en modélisation d'entreprise, tels que les activités, les ressources ou les entités. Ils couvrent donc, plus au moins complètement, les concepts des vues de la modélisation de la norme ISO 19440 et présentent des relations existantes entre ces concepts et des concepts relatifs à la phase d'ingénierie des besoins et des exigences que nous proposons de regrouper dans une vue des exigences. Dans le Tableau 1 nous résumons les vues couvertes par chaque langage.

Tableau 1. Couvertures des vues de la ME par les langages d'IBE

	RDM	KAOS	UCM	GRL	URML
Vue fonctionnelle	Domaine, processus de domaine, processus métier, activité d'entreprise, opération fonctionnelle, règle, Séquencement des étapes (when [...] do [...])	Opérations, événements, Transition d'états des objets par l'application d'opérations	Responsabilité, chemin, événement, sous scénario	Tâche, service	Groupe de processus, processus de cas d'utilisation, processus d'environnement, procédure, caractéristique du système
Vue des ressources	Entité fonctionnelle, ressource, ensemble de capacités	Agent (machine, collaborateur ou applicatif)	Composant (agent, groupe...)	Ressource (entité physique ou informationnelle)	Fournisseur de services (humain, machine ou applicatif)
Vue informationnelle	Objet d'entreprise et vue d'objet	Entité et relations	Le composant objet	Ressource informationnelle	Objet d'entité
Vue organisationnelle	Unité et cellule organisationnelle	/	/	/	/
Vue des exigences	Objectif, contrainte	But, hypothèse, obstacle, attente, exigence, propriété du domaine	/	Acteur, but, opinion, conflit, contribution	Partie prenante, but, hasard, menace, exigence fonctionnelle ou de qualité, idée, demande

La vue des exigences regroupe des concepts relatifs à la phase d'IBE, couverts par les langages d'IBE étudiés mais non proposés par la norme ISO 19440. Ils permettent de gérer les

besoins et argumenter la prise de décision concernant les exigences à réaliser. Ces concepts sont relatifs aux parties prenantes et aux exigences :

Les parties prenantes : pour analyser et gérer les besoins, il est important de garder une trace des versions des différentes déclarations des parties. Dans les langages étudiés, il est proposé de représenter les parties prenantes exprimant des besoins ou des buts, comme dans GRL et URML.

Les exigences : au niveau des langages orientés buts comme KAOS et GRL, les besoins et les exigences sont considérés comme deux concepts indépendants. KAOS et GRL proposent une représentation initiale du besoin sous forme de buts qu'ils proposent de raffiner pour en déduire les exigences et leur opérationnalisation, comme dans KAOS, ou en déduire directement les opérations/solutions qui répondent aux buts, comme dans GRL. URML ne fait pas de lien direct entre les buts et les exigences. Cette stratification entre besoins et exigences est intéressante. Elle permet de distinguer dans tout ce qui est collecté comme besoins ce qui est considéré comme exigences à réaliser.

En considérant leurs notations, leurs niveaux de détail et la capacité d'abstraction qu'ils nécessitent, tous les langages étudiés sont destinés à des utilisateurs experts sinon tout au moins formés et assistés. Ils sont utilisés par des experts faisant partie de la MOE, comme dans RDM, URML OU UCM. Dans KAOS et GRL, les experts métiers, faisant partie de MOA, participent à la définition des besoins mais sont secondés par des assistants à maîtrise d'ouvrage. Tous les langages étudiés proposent des artefacts qui nécessitent des compétences en modélisation. Ils ne sont donc pas accessibles aux collaborateurs des PME sans formation au préalable ou assistance.

Les langages étudiés apportent certes plus de rigueur à la définition des besoins et des exigences et sont une alternative aux méthodes traditionnelles informelles. Mais ils restent non accessibles à des utilisateurs n'ayant pas de compétences en modélisation, comme c'est souvent le cas des collaborateurs de la MOA et plus particulièrement les collaborateurs des PME. Par conséquent, les besoins sont définis par des collaborateurs de la MOE et ceci introduit le plus souvent un biais à la vision et aux besoins de la MOA. Nous considérons dans la partie suivante des langages plus accessibles qui profitent de la facilité d'utilisation du langage naturel et qui proposent des mécanismes pour pallier son aspect informel.

2.3.2 Vers des notations plus accessibles en IBE

L'étude proposée dans (Carrillo de Gea 2012) affirme que le langage naturel comme notation est utilisé (et couplé avec d'autres langages) dans 71% des 38 outils d'IBE étudiés. La représentation textuelle en langage naturel des exigences reste fréquente compte tenu de son utilisation facile par toutes les parties prenantes. Toutefois, ces exigences qui sont le plus souvent écrites en langage naturel libre, sont imprécises, ambiguës et ne se prêtent pas au traitement automatique. Des langages naturels « contrôlés » (*Controlled Natural Language* – CNL) ont donc été introduits pour remédier à l'aspect informel du langage naturel libre, en proposant un sous ensemble formalisé du langage naturel (Williams et al.2014). Les

CNLs, tels que Attempto Controlled English (Fuchs et al. 2006), SBVR (*Semantics of Business Vocabulary and Business Rules*) Structured English, RuleSpeak (OMG 2008), RuleCNL (Njonko et al. 2014) ou EARS (*Easy Approach to Requirements Syntax*) (Mavin & Box 2009) sont des langages semi-formels qui restreignent l'expression des exigences avec un ensemble de formulations permises. Les CNL sont considérés comme des notations semi-formelles puisqu'ils essaient de combiner la facilité d'utilisation du langage naturel informel avec la capacité d'outiller le processus d'ingénierie des exigences.

Nous présentons dans ce qui suit un principe de formulation d'exigences textuelles appelé « *boilerplate* » et le métamodèle SBVR qui est un standard proposant une structuration étudiée du vocabulaire et des règles métiers d'une organisation.

2.3.2.1 Boilerplates d'exigences

Les *boilerplates*, ou patrons textuels, sont un exemple d'approche utilisant un CNL, proposé à l'origine par (Hull et al. 2010) pour aider les ingénieurs et analystes dans l'écriture de documents de qualité et faciliter leur lecture. L'idée est simple : il s'agit de modèles à trous. Plus exactement un *boilerplate* est un modèle de phrase simple, dont la grammaire et une partie du vocabulaire sont données. Un *boilerplate* est constitué d'éléments fixes et d'un ensemble d'attributs (entre < >) paramétrables, dont la nature est connue (ex. The <user> shall be able to <capability>). Ces attributs peuvent être classifiés comme étant soit des actions, des entités (système, fonction, objet, événement, etc.) ou des quantités (Johannessen 2012)

Les éléments fixes de syntaxe des *boilerplates* peuvent dans ce cas être couplés à un vocabulaire prédéfini (représentant l'ensemble des attributs paramétrant les *boilerplates*) standardisé par l'organisation qui utilise les *boilerplates* ou par le domaine d'étude. Ceci apporte plus d'homogénéité et réduit l'ambiguïté en uniformisant les termes manipulés.

Les *boilerplates* présentent aussi une variabilité dans les expressions, où chaque *boilerplate* principal peut être dérivé par l'ajout de préfixes ou de suffixes. Aussi, des *boilerplates* peuvent être ajoutés au fur et à mesure de leur utilisation, ce qui suppose la maintenance d'un répertoire de *boilerplates*.

Le concept de *boilerplate* a été adopté par un certain nombre d'approches et de projets. Nous pouvons citer les travaux de (Stålhane et al. 2011) qui décrivent une approche d'analyse de la sûreté des systèmes où les exigences de sûreté sont définies en respectant un ensemble de *boilerplates*. Le prototype développé par le Projet Européen (CESAR Consortium 2011) utilise des *boilerplates* pour définir le RSL (Requirements Specification Language). (Mavin & Box 2009) dans leur approche EARS (Easy Approach to Requirements Syntax) proposent des patrons génériques sous forme de *boilerplates* pour exprimer des exigences orientées état et événement, des comportements non-désirés et des caractéristiques d'un système.

2.3.2.2 SBVR

SBVR (*Semantics of Business Vocabulary and Business Rules*) (OMG 2008) est un métamodèle dédié à la définition des connaissances métiers en CNL d'une manière non ambiguë et compréhensible par l'homme et la machine. Ce n'est pas un langage mais plutôt une syntaxe abstraite qui standardise la définition de modèles sémantiques du vocabulaire et des règles métiers d'une organisation (Hall, 2006). SBVR supporte la construction d'un vocabulaire constitué de l'ensemble des termes, définitions et relations qu'une organisation utilise lors de l'exercice de son métier. Ce dernier est construit à partir de :

- Concepts, qui représentent : (i) des concepts individuels correspondant à un seul objet, (ii) des types d'objet, qui sont des concepts génériques classifiant des objets selon leurs propriétés communes, (iii) des rôles, joués par les concepts pour définir leur fonction ou leur utilisation dans des situations particulières.
- "Fact types" (*faits*) : les *faits* représentent une relation entre des *concepts* ou une caractéristique d'un *concept*. Les *faits* sont construits à l'aide de verbes et peuvent être associatifs, de catégorisation, partitifs, etc.
- Règles métiers : elles sont construites à partir de *faits*, eux-mêmes construits à partir de *concepts*. Les *règles métiers* et les *conseils* (bonnes pratiques) pris en considération au niveau d'une organisation représentent la politique de gestion de cette dernière.
 - Les règles métiers peuvent être opératives (régissent le fonctionnement de l'organisation – elles expriment l'obligation, l'interdiction et la permission restreinte) ou structurelles (vraies par définition – elles expriment la nécessité, l'impossibilité et la possibilité restreinte)
 - Les conseils expriment la permission ou la possibilité.

Des langages ont été définis sur la base de SBVR et permettent son utilisation comme *SBVR Structured English* et *RuleSpeak* (OMG 2008). Formulés en langage naturel, ces langages sont utilisés par les experts et décideurs métiers formés sur les règles d'écritures et les formulations à respecter.

2.3.3 Synthèse : verrous constatés

Une méthode adéquate d'ingénierie des besoins et des exigences fournit aux émetteurs des besoins, à savoir les parties prenantes du projet de développement, une meilleure compréhension de leurs besoins et des implications de leurs décisions, et ce, pour une évaluation efficace des solutions proposées. Elle fournit à la MOE une définition claire du problème à partir de laquelle elle peut envisager des solutions. De ce fait, une compréhension commune est établie. Elle est souvent conditionnée par l'implication de la partie prenante MOA. Les méthodes étudiées réussissent plus au moins à impliquer la MOA, en fonction de leur orientation et de l'accessibilité des langages qu'elles proposent.

L'étude comparative des langages de modélisation d'exigence nous a permis de constater que les notations visuelles détaillées ne sont pas accessibles aux collaborateurs des PME sans formation ni assistance. Comme nous souhaitons par ce travail de recherche favoriser leur autonomie, notre choix se porte sur des langages CNL formalisant le langage naturel. Parmi ces CNL, ceux utilisant le principe de *boilerplate* assurent une formalisation correcte et guident les utilisateurs dans l'activité de modélisation.

Les CNL recensés dans cet état de l'art (EARS, SBVR *Structured English* et *RuleSpeak*) sont souvent utilisés de façon directe par des ingénieurs et des analystes pour améliorer la qualité des cahiers des charges. Il n'existe pas de méthodes associées à ces langages pour systématiser la production d'exigences comme le préconise le processus d'IBE. Aussi, selon (Pagès 2009) les langages basés sur SBVR tels que SBVR *Structured English* et *RuleSpeak* ne disposent pas de grammaires formellement définies pouvant être mises en correspondance avec le métamodèle de SBVR. Or, l'automatisation de l'utilisation d'un CNL requièrent une définition formelle de sa syntaxe, aussi appelée grammaire, pour guider les utilisateurs dans la modélisation syntaxiquement correcte.

En outre, les langages tels que EARS et RSL sont assez globaux : ils s'inscrivent dans une démarche d'ingénierie système et décrivent souvent des caractéristiques et fonctionnalités du système solution. Ils ne sont donc pas appropriés pour spécifier les besoins liés aux applications supports au système d'information par des collaborateurs non experts. Les langages relatifs à l'entreprise et à ses règles métiers, pouvant être utilisés par les collaborateurs de la MOA, comme ceux basés sur SBVR ne couvrent que la vue informationnelle du modèle de l'entreprise. Nous estimons que les CNL peuvent être davantage spécialisés et proposons, dans le chapitre IV, un CNL pour collecter des informations relatives au fonctionnement, à l'organisation, aux informations et aux ressources utilisées au sein de l'entreprise.

2.4 Progresser en confiance durant le processus d'IBE : vérification et validation

La mise en place d'un processus d'IBE a pour objectif l'amélioration continue de la qualité des exigences et ce par l'application de techniques de vérification et de validation des modèles et des spécifications des exigences. Ceci permet d'entamer avec confiance les étapes de conception et de réalisation du système. Dans cette partie, nous exposons la distinction entre les activités de vérification et de validation des exigences, puis nous présentons les qualités attendues d'une exigence ou d'un ensemble d'exigences. Enfin, nous exposons les types de techniques de vérification et de validation et leur utilisation dans les domaines de modélisation d'entreprise et d'ingénierie des besoins et des exigences.

2.4.1 Définitions : vérification et validation des exigences

En ingénierie système (BKCASE Editorial Board 2015), les activités de vérification et de validation ont été définies par « *do the thing right* » pour la vérification et « *do the right thing* » pour la validation (Boehm 1981).

En effet, **vérifier** un modèle de système revient à s'assurer que ce modèle, utilisé pour représenter et raisonner sur ce système, est conforme aux exigences de modélisation syntaxiques et sémantiques imposées par le langage de modélisation. Le but est bien de répondre à la question « *Construisons-nous correctement le modèle ?* » (Chapurlat, 2007). Nous devons donc vérifier que le modèle est bien construit, cohérent et complet dans la mesure du possible. Par analogie, le modèle des exigences est « correctement construit » lorsqu'il satisfait les exigences de qualité relatives à une spécification d'exigence ou à un ensemble d'exigences telles que décrites plus loin et quand il est conforme aux exigences de modélisation imposées ou induites par le langage de modélisation d'exigences utilisé.

Pour **valider** un modèle de système, il faut s'assurer que le modèle est pertinent et fidèle ou encore que le système, qu'il permet de représenter, répond aux besoins réels des parties prenantes qui ont initié sa construction. La question est donc de savoir si « *l'on construit le bon modèle* ». Ceci revient à supposer que le comportement décrit par le modèle est équivalent au comportement du système, en prenant en considération les hypothèses parfois restrictives de la modélisation (Chapurlat, Kamsu-foguem, & Prunet, 2006). Comme la validation est une confrontation entre une spécification et les besoins réels, elle implique la participation des parties prenantes qui ont exprimé ces besoins. Selon (Brottier 2009), dans le domaine du génie logiciel, les spécifications qui sont traditionnellement validées sont d'une part la solution logicielle, car elle reste concrète pour l'utilisateur, et d'autre part les spécifications d'exigences puisqu'elles ne comportent pas de détails techniques.

Il est donc nécessaire, pour valider les exigences, de se référer aux métiers de l'entreprise et aux domaines évoqués au niveau des besoins, d'une part, pour déterminer la pertinence des exigences vis-à-vis du monde réel et d'autre part, pour se contraindre quant à la faisabilité technique d'une solution.

Dans ce travail de recherche, nous nous intéressons principalement à la vérification des besoins et des exigences pour s'assurer que le modèle des exigences est correctement construit. Nous permettons aussi sa validation dans la mesure du possible. Nous présentons donc dans ce qui suit les qualités qu'une spécification d'exigence et un ensemble d'exigences doivent avoir.

2.4.2 Qualités attendues d'une exigence et d'un référentiel d'exigences

La vérification doit permettre d'affirmer l'existence ou l'absence de défaut au niveau des modèles d'exigences et ce pour assurer des critères de qualité. Il existe des normes (INCOSE 2012; IEEE 1998) qui guident l'élaboration des exigences en définissant la notion de qualité d'une spécification d'exigences. Ces normes préconisent que chaque exigence doit être :

- Nécessaire, utile et justifiable : sa suppression affecte la complétude de l'ensemble des exigences.
- Indépendante de l'implémentation.

- Complète : contient toutes les informations nécessaires pour démontrer sa satisfaction.
- Non ambiguë : ayant une seule interprétation possible.
- Singulière : portant sur une seule idée.
- Identifiable et traçable : son origine doit être clairement localisée et les éléments de conception concourant à sa satisfaction sont clairement identifiés.
- Faisable : réaliste, pour laquelle une solution existe ou peut être construite.
- Vérifiable : en terme de vérification du système ou du modèle du système. Il doit y avoir un moyen de s'assurer que le modèle la satisfait. Ce moyen doit être raisonnable en temps et coût d'application.
- Correcte : reflétant un besoin réel.
- Conforme au standard ou au langage appliqué.
- Concise et simple : ne comportant pas de spécifications complexes ou composées de sous spécifications.
- Communicable : formulée d'une manière comprise par les parties prenantes.

L'ensemble d'exigences, encore appelé référentiel des exigences, doit ensuite être :

- Complet : reflétant l'ensemble des besoins des parties prenantes et prenant en compte toutes les hypothèses et contraintes recensées sur l'environnement du système.
- Cohérent : ne comportant pas d'exigences contradictoires ou conflictuelles.
- Faisable : une solution satisfaisant l'ensemble des exigences peut être réalisée en respectant le délai, le budget, les choix techniques du projet et la législation.
- Limité : les exigences n'appartenant pas au contexte du problème à solutionner doivent être exclues.
- Structuré et homogène : respectant un cadre défini en adéquation avec la nature du système à réaliser de façon à ce que des sous-ensembles d'exigences en relation puissent être identifiés
- Non-redondant : ne comportant pas la même exigence plus d'une fois, ou reformulée d'une autre manière.
- Priorisé : composé d'exigences classées suivant leur importance.

Certaines de ces qualités restent des vœux pieux et sont par nature difficiles à évaluer. La non-ambiguïté des exigences peut par exemple être altérée par des conflits d'interprétation lorsque le vocabulaire des parties prenantes n'est pas unifié. Aussi, la complétude du référentiel des exigences dépend fortement de la capacité de détecter tout oubli objectif ou subjectif des parties prenantes, ce qui reste très difficile à réaliser.

D'autres qualités relèvent de la validation des exigences puisqu'elles supposent une confrontation avec le monde réel, telles que la faisabilité et la pertinence par rapport aux besoins. Les autres peuvent être garanties grâce à l'adoption d'un processus systématisé d'IBE doté de mécanismes de vérification. Nous cherchons avec ce travail à s'assurer, ou à défaut à se rassurer avec un niveau de confiance suffisant, sur les qualités attendues des exigences et du référentiel des exigences. Pour cela, nous présentons dans la suite quelques techniques de vérification et de validation susceptibles d'être utilisées.

2.4.3 Techniques de vérification et de validation

Les techniques de vérification et de validation sont utilisées pour s'assurer de la bonne construction d'un modèle, de sa satisfaction aux exigences et de sa pertinence. Certaines sont considérées comme informelles comme l'expertise, ou semi-formelles comme la simulation, le test ou la démonstration puisqu'elles ne revêtent pas un caractère exhaustif et indiscutable. D'autres sont considérées comme des techniques formelles comme le *model checking* ou le *theorem proving*. Ces techniques sont utilisées autant dans le domaine de la modélisation d'entreprise qu'en ingénierie des besoins et des exigences.

D'après (Chapurlat, 2007) les principales stratégies de vérification et de validation utilisées en modélisation d'entreprise sont :

- L'utilisation de standards et de cadres de référence pour guider la modélisation en respectant un ensemble de règles. Ceci réduit les possibilités d'erreurs au niveau de la construction de modèles et permet la vérification de la bonne utilisation des moyens de modélisation. C'est aussi un support pour l'analyse de la cohérence des modèles.
- L'expertise humaine pour effectuer des revues des modèles selon des critères prédéfinis pour en tirer des conclusions ou apporter un jugement. Et ce, en se référant aux connaissances et aux savoir faire des experts d'un domaine ou d'un métier, souvent à des fins de validation.
- L'exécution. Cette technique est possible lorsqu'on dispose d'une sémantique opérationnelle permettant l'exécution effective du modèle. Elle nécessite souvent le développement de modèles de simulation, devant être alimentés au cours de leur exécution par des scénarios.
- Les méthodes formelles utilisées avec des langages et concepts relevant du domaine des mathématiques ou pouvant être transformés vers un langage formel. Ces méthodes offrent des preuves rigoureuses et tangibles sur la satisfaction d'exigences, traduites en propriétés prouvables.

Ces techniques sont également utilisées par des méthodes en ingénierie des besoins et des exigences. RDM et KAOS préconisent par exemple des revues par des experts du métier. KAOS utilise également la preuve et le raisonnement formel sur la base des buts, traduits en modèles de Logique Temporelle Linéaire (LTL). RDM et UCM utilisent des techniques d'ani-

mation pour exécuter les processus et les scénarios pour vérifier leur cohérence et détecter des interactions indésirables.

Un récent état de l'art (Kamalrudin & Sidek 2015) recense des techniques semi formelles de vérification et de validation. Ces dernières fournissent un processus pour la détection des défauts d'exigences, sans être doté d'un modèle mathématique sous-jacent pour effectuer des raisonnements formels ou des procédures de preuve mathématique. Elles fournissent par exemple : des environnements de modélisation (Sinha et al. 2008), des moyens d'analyse inter modèle (Sabetzadeh & Nejati 2007), des règles de vérification qui peuvent être vérifiées avec des contraintes définies sur les modèles (Acharya et al. 2005; Blanc & Mounier 2008) ou grâce à des algorithmes de recherche (Litvak 2003) sur des modèles.

Une étude des pratiques actuelles en IBE pour la vérification et la validation des exigences (Fanmuy et al. 2012), affirme que les techniques semi formelles restent les moyens les plus utilisés, en particulier la revue et l'inspection des modèles par expertise, et la relecture des spécifications avec les parties prenantes. L'utilisation de modèles pour la définition des besoins et exigences est une valeur ajoutée aux projets d'ingénierie, car ils constituent un support pour l'analyse de la cohérence et de la complétude ainsi que pour l'évaluation du comportement du système dans un contexte ou une architecture donnée.

Par l'utilisation des CNL, les méthodes d'IBE répondent à certains problèmes liés à la formulation des exigences tels que l'ambiguïté et la redite et assurent d'avoir des exigences conformes au modèle du langage appliqué, concises, simples et communicables. Ceci est une manière de faire de la vérification par modélisation guidée comme proposé dans (Chapurlat, 2007).

Des techniques de traitement de langage naturel (*Natural Language Processing* – NLP) peuvent être utilisées pour vérifier l'exactitude syntaxique des exigences textuelles et leur conformité aux patrons textuels dans le cas d'utilisation de *boilerplates*. Pour améliorer la pertinence et la précision sémantique des exigences spécifiées en CNL, des travaux récents (Annervaz et al. 2013; Farfeleder et al. 2011; Tommila et al. 2013; CEZAR 2011) leur associent des ontologies qui représentent les connaissances des domaines concernés par le projet d'ingénierie.

Dans certaines méthodes, les modèles d'exigences en langage naturel sont transformés en modèles plus formels pour garantir certaines qualités ou appliquer des techniques formelles de vérification et de validation. Dans (Aceituna et al. 2014), des diagrammes d'état-transition sont déduits des exigences en langage naturel libre pour détecter des incohérences et des non-complétudes. Les transformations des exigences en langage naturel vers des langages formels tels que LTL (*Linear Temporal Logic*) sont réalisés dans (Tommila et al. 2013) et (Yan et al. 2015) dans le but de vérifier la satisfaction d'exigences de systèmes critiques par des modèles.

2.4.4 Synthèse

Nous souhaitons, par ce travail de recherche, améliorer le niveau de confiance des collaborateurs de la MOE en leur fournissant des modèles d'exigences vérifiés et, idéalement, validés au moins en partie.

Un modèle d'exigences vérifié doit respecter les propriétés de qualité relatives à une spécification d'exigence et au référentiel d'exigences. Il doit aussi être conforme au langage de modélisation utilisé.

Nous nous basons sur l'application d'un processus formalisé d'ingénierie dotés de techniques semi-formelles de vérification et de validation pour assurer les qualités attendue des exigences, nous utilisons:

- Des standards et des cadres bien définis (en modélisation d'entreprise) comme base pour définir le modèle conceptuel régissant les règles de construction syntaxiques et sémantiques des exigences et fournissant un support pour l'analyse de la cohérence intra et inter modèles et pour la définition de règles de vérification.
- La modélisation guidée et contrainte par utilisation de CNL de type *boilerplate* couplée avec des techniques de traitement de langage naturel pour assurer une représentation rigoureuse des besoins et exigences.
- L'expertise : pour valider les exigences, il est nécessaire de disposer de connaissances relatives au domaine de la PME. Nous nous appuyons donc sur l'expertise des parties prenantes.

D'autres techniques sont écartées parce que les conditions nécessaires pour leur application ne sont pas disponibles. Les techniques ayant comme objectif d'améliorer la pertinence et la précision sémantique des modèles textuels disposent de modèles du domaine comme des ontologies. Or, nos travaux sont dans un premier temps génériques à tout type de PME. Nous ne disposons pas *a priori* de connaissances sur les métiers ou secteurs d'activité des PME. Nous nous basons sur les connaissances et savoir-faire des parties prenantes pour décider de la pertinence des exigences.

Les techniques formelles telles que le *model checking* et le *theorem proving* sont un moyen prometteur pouvant garantir de manière indiscutable la satisfaction de certaines qualités. Cependant leur utilisation suppose des transformations du langage avec lequel les exigences sont spécifiées vers un langage formel. Ceci est difficile dans notre contexte puisque les besoins et les exigences relèvent des aspects d'un haut niveau d'abstraction liés à la définition du système d'information d'une entreprise.

2.5 Conclusions et contribution des travaux

Le but de nos travaux est de développer une méthode qui facilite la construction et la vérification, avant toute utilisation ultérieure, de modèles utilisés pour comprendre la structure, le fonctionnement et les besoins d'informatisation des processus de gestion d'une PME.

Pour cela, nous avons précisé dans le chapitre I que cette méthode doit répondre à plusieurs objectifs. En particulier, elle doit assurer l'autonomie des collaborateurs de la PME pour décrire intuitivement et aussi simplement que possible leurs besoins d'informatisation mais aussi pour vérifier et valider autant que possible ces informations. Enfin, elle doit favoriser la coopération entre la MOA et la MOE lors du recueil de besoins et du passage vers la définition des exigences. Nous avons ensuite proposé dans ce qui précède une analyse critique de travaux portant sur les moyens et les méthodes de Modélisation d'Entreprise (ME) et d'Ingénierie des Besoins et des Exigences (IBE).

Les travaux présentés dans la suite de ce mémoire convergent donc vers la proposition d'une méthode d'ingénierie des besoins et des exigences :

- La méthode supporte une approche descriptive, et non prescriptive, pour représenter les besoins en décrivant la structure et le fonctionnement de l'entreprise.
- La méthode se base, pour la description des activités, sur une vision locale où les collaborateurs définissent ce qui est réalisé au niveau de chaque poste. Une déduction des processus transversaux est faite *a posteriori*.

Le positionnement de la méthode, dans le contexte présenté au chapitre I et au regard des méthodes d'ingénierie existantes dans la littérature, est présenté dans la section suivante.

2.5.1 Positionnement

La méthode que nous proposons s'intègre au projet d'informatisation des processus d'une PME, dans lequel des solutions applicatives sont développées. Pour cela, RESULIS souhaite disposer d'une démarche de conception et de développement agile (Figure 9).

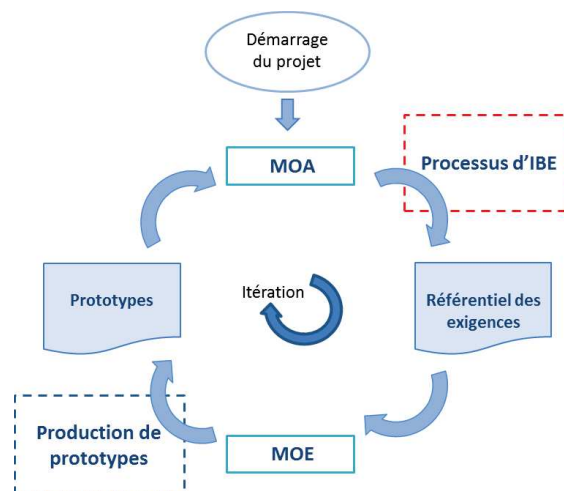


Figure 9. Démarche agile d'informatisation.

En effet, les méthodes agiles (Stapleton 1997; Beck 2000; Cao & Ramesh 2008b; Schwaber & Beedle 2002) placent la plus haute priorité à la satisfaction du client et lui livrent rapidement dans chaque cycle du développement itératif un système opérationnel (Wikipédia 2015). En particulier, nous nous concentrons ici sur la phase d'IBE pendant laquelle RESULIS souhaite assurer une collaboration réussie entre elle et la PME pour l'élaboration du référentiel des exigences. Dans les approches agiles, cette collaboration MOE / MOA est fondamentale.

La méthode d'IBE que nous proposons s'intègre également au cycle de développement basé sur les modèles (Figure 10) qui favorise aussi le développement agile.

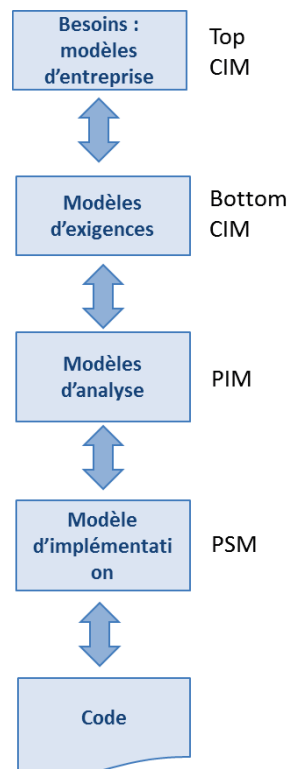


Figure 10. Cycle de développement basé sur les modèles. Apport de l'interopérabilité dirigé par les modèles (Bourey et al. 2007)

Ceci permet d'accélérer le cycle de développement et de production de prototypes et de rendre interopérables les langages et outils de la MOA et de la MOE. L'ingénierie dirigée par les modèles (IDM) (Kent 2002) préconise l'utilisation intensivement de modèles et de reporter le recours au code aux phases aval du cycle de développement quitte à le générer automatiquement. Elle propose une chaîne de transformation de modèles (Lúcio et al. 2014) : CIM (*Computation Independent model*) vers PIM (*Platform Independent Model*) vers PSM (*Platform Specific Model*). Les modèles de type CIM sont des modèles d'exigences indépendants des détails liés à l'implémentation du système. Les PIM sont des modèles d'analyse qui concrétisent le métier de l'entreprise et les PSM intègrent les spécificités de la plateforme d'exécution du système. Ces différents niveaux d'abstraction sont nécessaires pour que les transformations soient possibles en réduisant l'écart existant entre les modèles liés à l'entreprise et le code. De la même manière, le modèle de référence MDI (*Model Driven Interoperability*) (Bourey et al. 2007), qui profite de l'approche dirigée par les modèles pour

réduire les problèmes d'interopérabilité, propose une stratification supplémentaire du modèle CIM, le partitionnant en deux niveaux : CIM-Top et CIM-Bottom, pour réduire l'écart souvent existant entre les niveaux CIM et PIM (Grangel et al. 2007; Lemrabet et al. 2012).

- Le niveau CIM-Top : où le problème est abordé du point de vue de la modélisation d'entreprise pour représenter le point de vue métier des utilisateurs.
- Le niveau CIM-Bottom : représente les exigences relatives système à réaliser. Il s'agit de la partie du CIM-Top qui doit être implémentée, toujours indépendamment des détails liés à l'implémentation.

Cette stratification conforte notre idée de distinction entre les besoins et les exigences, où l'on définit d'abord l'ensemble des besoins avant de préciser ce qui est considéré comme exigences à réaliser. Tout comme le modèle de référence MDI, nous considérons les exigences du point de vue de la modélisation d'entreprise dans le niveau CIM-Top. Nous contribuons donc à la déduction des exigences représentés par le niveau CIM-Bottom par l'application systématique du processus d'IBE que nous proposons.

Nous proposons avec la méthode d'IBE un point d'entrée de la chaîne de transformation de l'IDM. L'intégration de l'ingénierie des besoins et des exigences au processus d'ingénierie dirigée par les modèles est de fait encore une question ouverte (Bryant et al. 2003; de Almeida Ferreira & da Silva 2009; Brottier 2009; Yue et al. 2011) et le développement de transformations même semi-formelles des modèles d'entreprise vers des modèles d'exigences ou d'analyse comblerait un manque important dans le cycle de l'IDM (Yue et al. 2011).

La méthode d'IBE, à proprement dite, dont la démarche est présentée au chapitre V, repose sur un socle de concepts et de relations réunissant les concepts et relations de la modélisation d'entreprise d'une part et de l'ingénierie des besoins et des exigences d'autre part. Ce socle conceptuel permet ensuite d'assister voire d'automatiser certaines étapes de modélisation, de vérification et de transformation de modèles restitués pour la validation de façon à accroître le niveau de confiance des parties prenantes du projet d'informatisation. Les composantes essentielles de la méthode sont précisées au niveau des contributions attendues.

2.5.2 Hypothèse de travail : l'organisation avant l'informatisation

Rappelons que le système d'information d'une entreprise est structuré en un ensemble connecté et interagissant de composants de diverses natures : ressources humaines, processus, applications logicielles supports à ces processus, bases de données, réseaux et protocoles. Trois types de processus tapissent l'organisation pour servir la stratégie de l'entreprise (Morley et al. 2005) :

- des **processus opérationnels** : représentant des processus opérationnels qui couvrent le cycle de vie des produits ou des services de l'entreprise, de la définition à la livraison, voire la maintenance. Ils représentent ce que fait l'entreprise pour produire

sa valeur ajoutée vis-à-vis de ses clients et de ses partenaires externes. Ils produisent un résultat qui correspond à la raison d'être de l'entreprise.

- des **processus supports** : ce sont des processus qui couvrent l'approvisionnement pour l'entreprise en ressources nécessaires à la réalisation des processus métiers. Ils ne font pas intervenir les clients finals de l'entreprise. Ils sont source de coût sans création directe de valeur,
- des **processus de pilotage** : ce sont les processus qui permettent à l'entreprise de conduire et d'orienter ses processus métiers et supports dans le respect de ses objectifs stratégiques. Sous l'angle système d'information, ils se traduisent souvent par la production d'un tableau de bord (suivi des ventes, suivi des réclamations, etc.)

Les processus que nous appelons « **de gestion** » peuvent être des sous-processus des processus métiers, supports ou de pilotage. Il s'agit de tout processus dont l'informatisation peut procurer un avantage stratégique à l'entreprise. Toutefois, RESULIS se spécialise dans l'informatisation de processus relatifs à la collecte, le traitement et la restitution d'informations administratives, commerciales liées à la vente, à la planification, etc. RESULIS ne traite pas les projets relatifs aux systèmes de production de produits physiques ou liés à des technologies offrant des fonctionnalités spécifiques.

Par hypothèse, nous considérons connus et stables les processus de gestion et l'organisation de la PME, avant même de lancer un projet d'informatisation de ces processus. De même, nous ne cherchons pas ici à évaluer ou à optimiser l'activité ou la performance de l'entreprise.

Plus qu'une hypothèse, ceci est un constat réel. Forte de son expérience, RESULIS affirme que les projets d'informatisation qui réussissent sont ceux qui se basent sur un fonctionnement et une organisation réfléchies. De plus, il est largement reconnu que la technologie apportée par l'informatisation seule n'est pas nécessairement une solution adaptée aux problèmes de gestion. L'effort d'amélioration doit donc d'abord se faire au niveau organisationnel. Ceci garantit l'adhésion des collaborateurs et permet de converger vers une solution d'informatisation correspondante aux attentes et aux pratiques de la PME.

En outre, la maîtrise et la connaissance claire des activités par les collaborateurs de la PME est une condition nécessaire pour leur assurer une autonomie lors de l'expression des besoins, puisqu'ils sont les acteurs principaux de ce processus. Néanmoins, si les situations fonctionnelles et organisationnelles voulues de l'entreprise ne sont pas totalement réfléchies ou partagées, nous offrons aux collaborateurs de la PME, avec ce travail, le moyen de considérer et d'homogénéiser leur organisation.

Le décideur de la PME peut enfin, éventuellement *a posteriori*, demander d'apporter de nouvelles pratiques selon de nouvelles considérations ou contraintes de l'environnement. En effet, ces décisions doivent être centralisées, et c'est souvent le choix du responsable visionnaire ayant une vue d'ensemble de la stratégie de l'entreprise. Une gestion du changement sera, dans ce cas, nécessaire pour préserver l'adhésion des collaborateurs de l'entreprise.

2.5.3 Contribution attendue

Pour répondre aux objectifs établis pour ces travaux de recherche, la méthode d'IBE que nous proposons s'articule autour des quatre composantes classiques d'une méthode :

Les **concepts** et les **relations** qui les lient ainsi que les propriétés ou attributs qui les caractérisent sont présentés et illustrés dans le chapitre III. Il s'agit ici de désigner et formaliser les *choses* que manipulent les utilisateurs et auxquelles ils font référence lorsqu'ils modélisent leur environnement de travail et leurs besoins d'informatisation. Ces *choses* sont décrites de manière statique comme dynamique en respectant un ensemble de règles de construction et de règle d'évolution qu'il faut expliciter, puis ensuite vérifier pour s'assurer, ou à défaut se rassurer sur l'absence d'omissions, d'incohérences, d'imprécisions ou encore de redites et reformulations parasites.

En s'inspirant des cadres de référence d'architecture d'entreprise présentés au chapitre II, les concepts que nous proposons reflètent la situation de l'entreprise à informatiser et sont relatifs au fonctionnement de l'entreprise, aux informations et ressources qu'elle utilise et à son organisation interne. D'autre part, notre stratégie d'expression des besoins est orientée scénario. Les activités qui décrivent le fonctionnement sont définies en termes d'actions et d'interactions et les conditions de leur environnement d'exécution sont définies en termes de ressources disponibles, de documents nécessaires, d'habilitations à utiliser les informations, etc. Ces concepts restent familiers et accessibles pour les collaborateurs de la PME et en phase avec leur langage métier.

Les **langages de modélisation** permettant de manipuler les concepts et relations proposés sont présentés et illustrés dans le chapitre IV. Les concepts et relations vus plus haut sont regroupés par vues et l'ensemble est représenté au moyen d'un modèle conceptuel de données ou métamodèle que nous décrirons plus loin. Ce métamodèle décrit ainsi la syntaxe abstraite associée aux langages de modélisation utilisés aussi appelés DSL (*Domain Specific Language*). Une notation (syntaxe concrète), qui représente le moyen d'expression utilisé (textuel ou graphique) est ensuite associée à chaque DSL. Nous proposons, pour la phase d'expression des besoins, un langage textuel basé sur un langage naturel contrôlé (*Controlled Natural Language* (Fuchs et al. 2006; Njonko et al. 2014; OMG 2008)). D'autres DSL, à la syntaxe concrète graphique et aux formalismes simples sont utilisés ensuite lors des restitutions.

Une **démarche** est décrite sous forme d'un ou plusieurs processus opératoires permettant d'appliquer les langages pour modéliser et analyser les besoins de la PME au regard de sa structure et de son fonctionnement. Cette démarche est itérative (Figure 11) et enchaîne des étapes :

- 1) D'expression des besoins dans un premier temps où les utilisateurs de la PME sont guidés et assistés pour décrire leur besoins d'informatisation.
- 2) De vérification pour empêcher que les défauts inhérents aux besoins se reportent sur les exigences. Les modèles des besoins sont analysés et vérifiés pour détecter

des incohérences et tenter de juger de leur pertinence.

- 3) De validation et de collaboration avec la MOE pour mettre au point le référentiel des exigences, par exemple au travers de phases de négociations et de prise de décisions MOA/MOE qui s'appuient sur un référentiel initial des exigences pour arriver à un consensus.

Les **outils** permettent de supporter les activités de la démarche, de manipuler mais aussi d'analyser les modèles établis avec les langages proposés. Il s'agit ici d'outils informatiques de modélisation, de vérification, de validation, de transformation et de documentation. Les outils et langages que nous utilisons sont ouverts aux transformations de modèles. Les modèles établis sont donc transmis aux équipes de développement de la MOE et peuvent être exploités grâce à des techniques de génération de code.

La démarche opératoire et les outils supports proposés sont enfin présentés et illustrés dans le chapitre V.

2.5.4 Exemple illustratif

Pour faciliter la lecture et l'appropriation des éléments de la méthode, nous les illustrons dans la suite du manuscrit sur un cas d'étude expérimental réalisé avec une PME cliente de RESULIS, la société SDT. SDT est une PME régionale de 30 collaborateurs spécialisée dans la micronisation, l'analyse et le traitement à façon des poudres fines et ultrafines.

SDT collabore avec des clients de divers secteurs industriels en leur offrant une large palette de prestations : broyage, micronisation, analyses physico-chimiques, tamisage, granulation, mélange, formulation, etc. La gamme des services proposés concerne des matériaux dosés de quelques grammes à plusieurs centaines de tonnes de produit. SDT est également un organisme de formation continue du personnel technique (ingénieurs, techniciens, ...) dans le domaine des sciences et technologies des poudres.

Chapitre III : Concepts supports à la Méthode d'IBE

3.1 Introduction

Nous présentons dans ce chapitre notre proposition relative à la première brique constituant la méthode d'IBE, à savoir les **concepts** et les **relations** qui les lient. Nous commençons par présenter le cadre normatif dans lequel ces derniers s'inscrivent et sont choisis. L'ensemble de ces concepts et relations est ensuite présenté. Ils constituent la syntaxe abstraite, un constituant du métamodèle SI-PME. Il s'agit de la syntaxe abstraite sur laquelle nous basons la définition des langages utilisés par la méthode proposée, présentés au chapitre IV. Des règles de modélisation sont ensuite introduites pour vérifier certaines des qualités nécessaires des exigences telles que la cohérence, la non-reformulation et la complétude.

3.2 Cadre normatif pour la construction du modèle de l'entreprise

Notre approche d'IBE inclut une analyse et une représentation du modèle de la PME. En effet, les besoins et les exigences décrivent la situation de l'entreprise à informatiser et sont relatifs au fonctionnement de l'entreprise (activités, processus, échanges avec l'environnement...) et à sa structure (organisation, postes, ressources...).

Comme vu précédemment, le modèle de l'entreprise, tel que défini en modélisation d'entreprise, est bâti pour évoquer les concepts et relations, propriétés, attributs et contraintes nécessaires pour construire et formaliser les besoins relatifs à la réalisation d'applications supports au SI de la PME. C'est pourquoi nous proposons de suivre les principes de la modélisation d'entreprise et d'exploiter le cadre normatif défini par les normes ISO 19439 et ISO 19440.

L'ISO 19439 (*Framework for Enterprise Modelling*) (ISO/DIS 2002) propose un cadre unifié pour structurer les entités devant être manipulées par une méthode d'ingénierie d'entreprise. Les entités sont articulées en 3 axes (définis dans le chapitre II) : les phases du cycle du projet, les niveaux de généricité des modèles et les vues de modélisation nécessaires. Ce dernier axe est détaillé par la norme ISO 19440 (*Constructs for Enterprise Modelling*) (ISO/DIS 2004) en 14 *constructs* (patrons partiels de modèles) de modélisation (Figure 6) qui couvrent les vues fonctionnelle, informationnelle, ressources et organisationnelle. Cependant, pour les objectifs de ces travaux, ces normes n'incluent pas de concepts pour manipuler et partager des concepts relatifs aux activités du processus d'IBE. En effet, les modèles que l'on peut construire selon les vues de modélisation donnent une vision des exigences sans pour autant les séparer et les formaliser de façon à construire un référentiel clair facilitant la prise de décision des parties prenantes. De plus, les concepts des vues de modélisation de la norme ISO 19440 ne permettent pas de représenter les parties prenantes, leurs rôles et des différentes versions de leurs déclarations.

Nous proposons donc, dans le contexte de nos travaux, un enrichissement (Amokrane et al. 2014) du cadre ISO 19439 pour prendre en compte le cycle nécessaire d'ingénierie des besoins et des exigences pour des projets d'ingénierie en entreprise, ici des projets d'informatisation de PME. Cet enrichissement est schématisé dans la Figure 11.

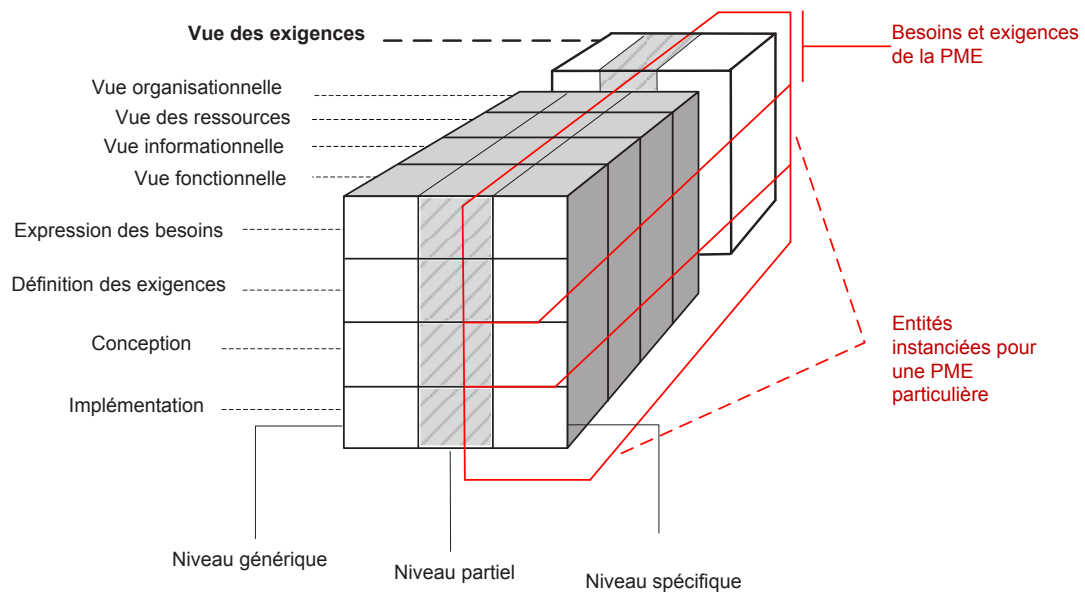


Figure 11. Enrichissement du cadre normatif ISO 19439 [Amokrane et al. 2014].

Le cycle de vie des entités modélisées : nous résumons les phases de définition des concepts et d'identification des domaines de l'ISO 19439, dans la phase d'expression des besoins, suivie de la phase de définition des exigences.

Pendant les phases d'expression des besoins et définition des exigences, l'instanciation du modèle générique permet de construire des modèles spécifiques à la PME et un référentiel d'exigences. Les applications supports au système d'information de l'entreprise de la PME sont conçues pendant la phase de conception et réalisées pendant la phase d'implémentation. Pour adopter un cycle de développement agile il est nécessaire de garder une trace des modifications apportées aux exigences pendant tout le cycle du projet. Le référentiel d'exigences est également utile pour la planification du projet de conception et d'implémentation, pour l'estimation des coûts de réalisation ou pour justifier de proche en proche les choix de conception mais aussi de développement qui suivront.

La généricité : cet axe représente le processus d'instanciation des modèles génériques et partiels en modèles spécifiques à l'entreprise. Le niveau générique est défini par la syntaxe abstraite que nous proposons. Le niveau partiel de généricité peut regrouper une typologie d'entreprises ayant un secteur d'activité commun (finances, santé, agroalimentaire...) ou des fonctions communes à plusieurs entreprises (facturation, planification, vente en ligne...). Nous n'abordons pas ces spécificités dans notre travail et considérons les PME indépendamment de leur secteur d'activité avec tous les processus de gestion qu'elles souhaitent informatiser. Nous proposons donc un modèle générique, à instancier au niveau spécifique avec chaque PME.

Les vues de la modélisation : nous introduisons une vue de modélisation des exigences complémentaire et cohérente avec les autres vues de modélisation (Figure 12) pour définir des concepts relatifs à la phase d'IBE. En effet, ces concepts ne sont pas proposés distinctement par la norme ISO 19440. Ils permettent de gérer et de restituer les exigences et argu-

menter les prises de décision concernant les exigences à réaliser. Nous enrichissons de même et détaillons les *constructs* relatifs aux quatre vues de la modélisation de l'ISO 19440 pour construire la syntaxe abstraite des langages d'IBE définis plus loin.

L'ajout de la vue des exigences nécessite évidemment de définir de nouvelles relations entre les concepts communément admis en modélisation d'entreprise (processus, activités, rôles, etc.) et ceux relatifs aux besoins et exigences. Plus particulièrement, ces relations doivent permettre de :

- Préciser les rôles des parties prenantes dans le projet d'informatisation.
- Connaître la position, dans la structure de l'entreprise, des parties prenantes faisant partie des collaborateurs de l'entreprise.
- Garder une traçabilité de « qui a dit quoi » et des versions des différentes déclarations de toutes les parties prenantes pour pouvoir les confronter et, si nécessaire, arbitrer.
- Faciliter le passage des besoins aux exigences, en permettant de faire des choix sur la base des besoins collectés pour définir les exigences à réaliser.

Finalement, l'adoption d'un métamodèle commun, présenté dans la suite, pour représenter ces concepts issus de la modélisation d'entreprise et de l'ingénierie des besoins et des exigences permet de construire une base conceptuelle pour l'analyse et la vérification des besoins et des exigences. De plus, ceci permet de justifier et de garder une trace des choix et décisions conceptuelles, puisque les besoins sont définis en termes de fonctionnement, organisation et ressources et permettent de déduire des exigences ayant un sens pour les phases aval du projet d'informatisation.

3.3 Syntaxe abstraite pour l'expression des besoins et le passage des besoins aux exigences

Nous explicitons ici le modèle conceptuel de la méthode proposée, ici baptisé métamodèle SI-PME, défini sur la base des *constructs* de la norme 19440. Pour simplifier la lecture de ce métamodèle, nous présentons les vues de modélisation séparément, bien qu'il s'agisse d'un métamodèle unique où les concepts et relations de toutes les vues sont interconnectés. Nous définissons dans la suite les concepts principaux constituant chaque vue et les principales relations qui les lient. De fait, certains concepts sont reportés dans plus d'une vue pour montrer les relations entre vues.

Pour une meilleure lisibilité du métamodèle SI-PME présenté dans les figures 12 à 17, certaines propriétés communes à plusieurs concepts sont regroupées dans les classes abstraites (colorées en gris dans les figures représentant le métamodèle) dont héritent ces concepts.

Les propriétés des concepts sont typées. Leur type peut être un type standard (entier, chaîne de caractères, etc.) ou une énumération (i.e. un ensemble fini de valeurs prédéfinies). Dans

ce cas, les éléments constituant l'énumération sont donnés dans chaque vue (classes colorées en vert dans les figures représentant le métamodèle).

Un tableau est présenté pour chacune des vues : il synthétise les concepts de la vue. Nous associons une définition à chaque concept.

3.3.1 Vue du contexte de l'étude

Cette vue permet de cadrer les limites du système à informatiser en représentant les domaines d'activité concernés par l'étude. Le contexte de l'entreprise dans son environnement est également indiqué en précisant les produits/services qu'elle fournit et tous les partenaires externes susceptibles de provoquer un changement dans le fonctionnement de l'entreprise. Les échanges, entrants ou sortants, que l'entreprise établit avec ces derniers sont également représentés.

Tableau 2. Concepts de la vue du contexte.

Désignation du concept	Définition
Enterprise	L'entreprise est l'objet de l'étude et l'élément d'entrée pour le recueil des besoins. L'entreprise est le lieu de création de produits (<i>EnterpriseProduct</i>) ou de services (<i>EnterpriseService</i>), dans un ou plusieurs domaines d'activité (<i>Domain</i>), en interagissant avec des partenaires externes (<i>ExternalPartner</i>) avec lesquelles elle établit des échanges (temporaire ou non) (<i>ExternalEnterpriseInput</i> / <i>ExternalEnterpriseOutput</i>) pour maintenir le bon fonctionnement de ses activités.
Domain	Le domaine d'activité représente, selon l'ISO 19440, la partie de l'entreprise concernée par l'étude et devant de fait être modélisée. Il définit l'activité de l'entreprise dont la finalité est d'apporter des services ou produits (<i>EnterpriseProduct</i> / <i>EnterpriseService</i>) à un marché spécifique. L'entreprise a donc des objectifs (<i>Objective</i>), définis à un haut niveau de gestion, et est soumise aux contraintes liées à ce domaine d'activité. Elle met en place une politique (<i>BusinessPolicy</i>) pour favoriser l'accomplissement de ces objectifs.
EnterpriseProduct / EnterpriseService	Les produits ou services sont des objets d'entreprise (<i>EnterpriseObject</i>) dont la fabrication, la vente, la mise à disposition ou la location font partie du métier de l'entreprise, dans le cadre d'un domaine d'activité (<i>Domain</i>) i.e. dénotent sa valeur ajoutée vis-à-vis du marché. Ils peuvent représenter les sorties finales des processus de l'entreprise. Selon l'ISO 19440, dans leurs états intermédiaires, les produits peuvent être en entrée ou en sortie des activités appartenant aux processus de production. Les produits complexes peuvent également être composés d'autres produits.
ExternalPartner	Le partenaire externe peut être une personne, une entreprise, une catégorie de clients, de fournisseurs, sous-traitants, partenaires autres (Banque, collectivités, etc.) ou concurrents avec lesquels l'entreprise interagit d'une façon directe ou indirecte, de façon temporaire ou non, définissant ainsi le contexte socio-économique de l'entreprise.

	Les partenaires de l'entreprise peuvent différer d'un domaine d'activité (<i>Domain</i>) à un autre. En plus des échanges (<i>ExternalEnterpriseInput</i> / <i>ExternalEnterpriseOutput</i>), les partenaires externes peuvent intervenir de façon locale au niveau de l'activité de l'entreprise en étant clients, fournisseurs ou sous-contractants par exemple.
ExternalEnterpriseInput / ExternalEnterpriseOutput	Les entrées externes et les sorties externes de l'entreprise représentent les relations d'échange qu'entreprend l'entreprise avec ses partenaires externes (<i>ExternalPartner</i>). Selon le domaine d'activité, la relation de l'entreprise avec son environnement externe est définie à un haut niveau de gestion (comme préconisé par l'ISO 19440) grâce à ces concepts.

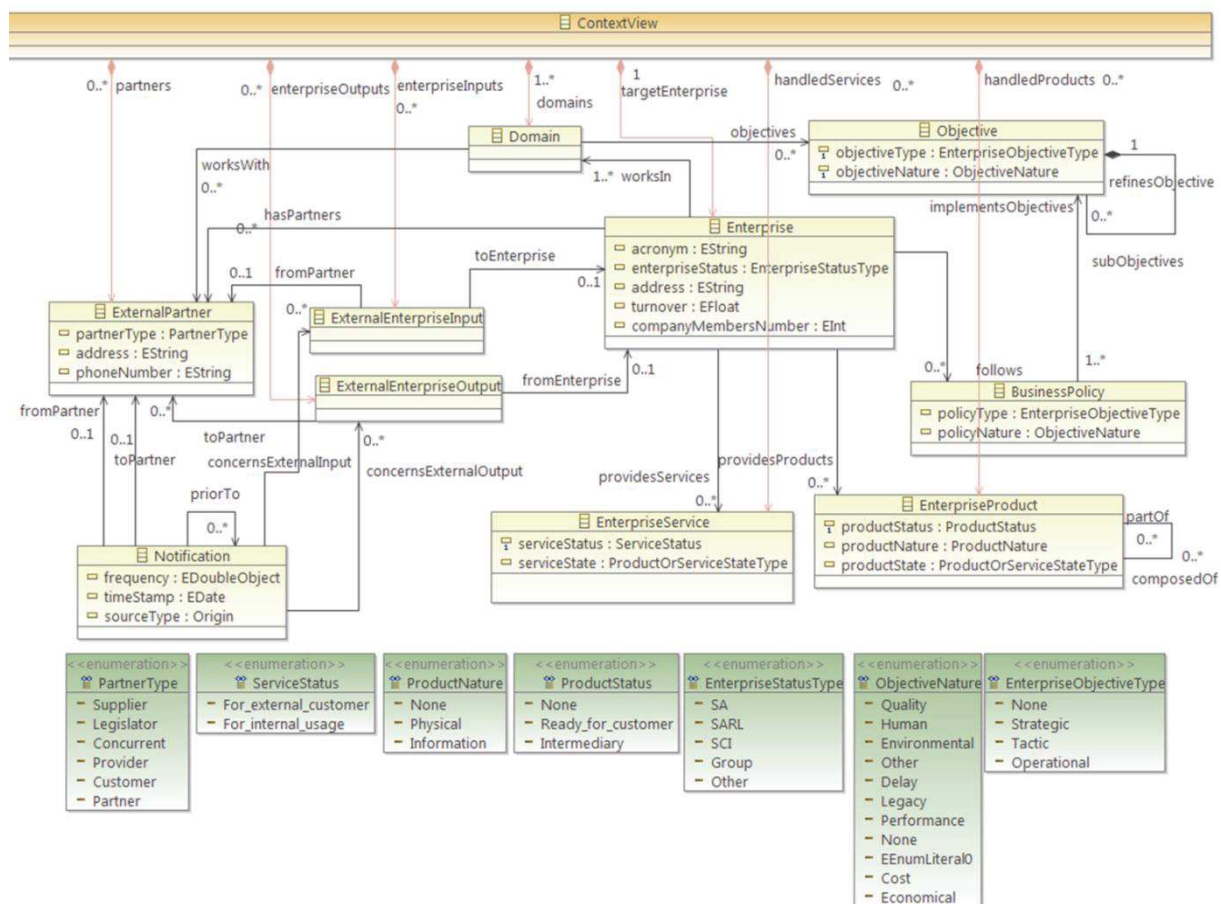


Figure 12. Concepts de la vue du contexte.

3.3.2 Vue organisationnelle

Cette vue permet de représenter la structure interne de l'entreprise. Nous adoptons les concepts « *Organizational Cell* » et « *Organizational Unit* » proposés par l'ISO 19440 pour définir la décomposition organisationnelle de l'entreprise en cellules organisationnelles puis en postes détenus par les collaborateurs de l'entreprise.

Tableau 3. Concepts de la vue organisationnelle.

Désignation du concept	Définition
Organisational-Cell	La cellule organisationnelle représente une entité appartenant à la structure organisationnelle de l'entreprise (service, département...etc.). Elle définit une des fonctions principales de l'entreprise et peut couvrir plusieurs domaines d'activité (<i>Domain</i>). La cellule est constituée d'au moins un poste (<i>Workstation</i>). Un niveau de hiérarchie lui est associé et elle peut être décomposée en d'autres cellules de niveau hiérarchique inférieur. Les PME peuvent ne pas avoir besoin de relations de décomposition hiérarchique en plusieurs cellules organisationnelles et peuvent être structurées en une seule cellule.
Workstation	Le poste ou poste de travail est la brique de base de la structure organisationnelle de l'entreprise. Il représente un groupement de rôles (<i>Role</i>) détenus par un ou plusieurs collaborateurs (<i>CompanyMember</i>) de l'entreprise définissant leurs responsabilités. Le poste fait partie d'une cellule organisationnelle et des relations de « subordonné à », « supérieur de » peuvent exister entre les postes pour définir ici aussi un type de hiérarchie entre postes.

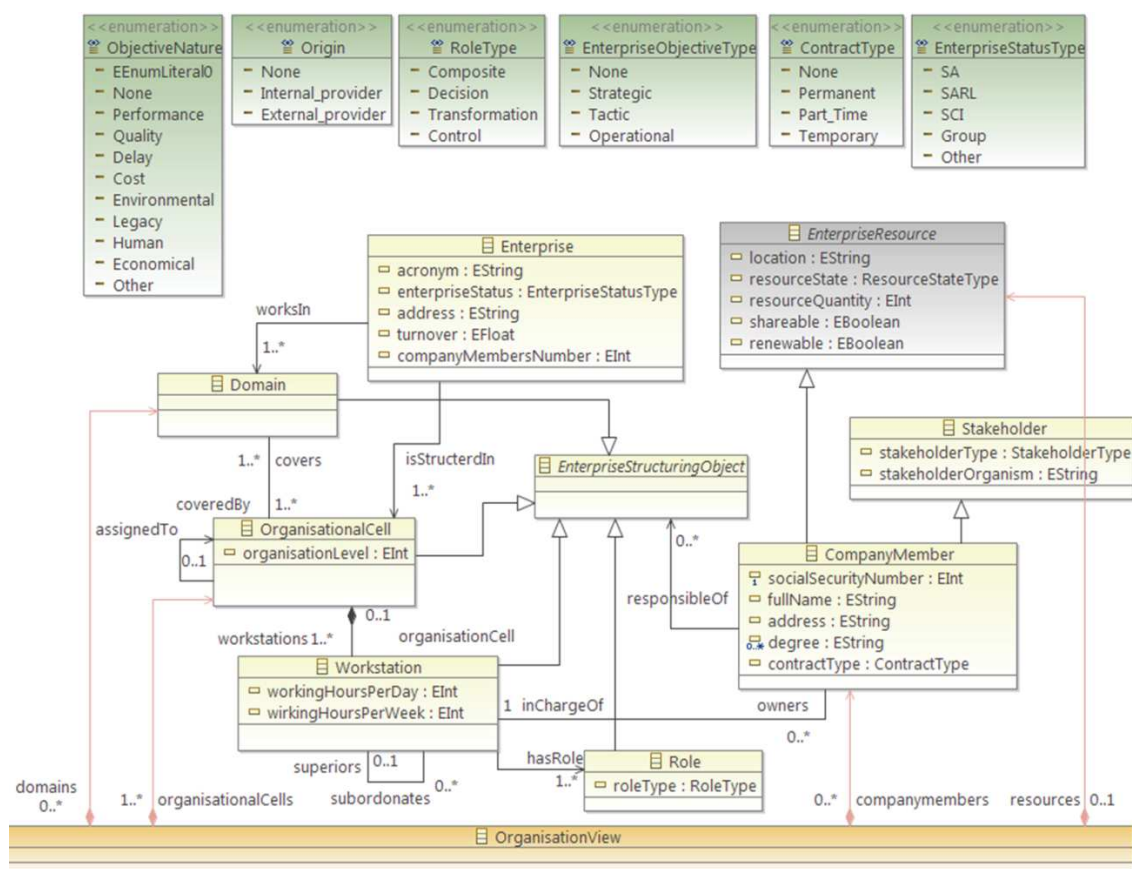


Figure 13. Concepts de la vue organisationnelle.

3.3.3 Vue des ressources

Les concepts de cette vue définissent des objets d'entreprise nécessaires pour la réalisation des activités de l'entreprise. En effet, les ressources sont des objets supports aux activités qui sont utilisées sans être transformées. Elles sont donc souvent réutilisables et peuvent être partageables par plusieurs postes de travail. Il est nécessaire de connaître leur disponibilité, aptitudes et capacités pour pouvoir planifier et tracer leur utilisation.

Tableau 4. Concepts de la vue des ressources.

Désignation du concept	Définition
Enterprise-Resource	La ressource d'entreprise est l'objet d'entreprise (<i>EnterpriseObject</i>) qui exécute ou fournit les aptitudes nécessaires à l'exécution partielle ou totale de chaque activité de l'entreprise (<i>EnterpriseProcessor</i>). Ce concept est spécialisé, comme préconisé par l'ISO 19440, en sous classes selon sa manifestation au niveau de l'entreprise. On peut donc avoir des ressources humaines (<i>CompanyMember</i>) des équipements matériels (<i>Equipment</i>) et des ressources applicatives (<i>Application</i>). Les mainteneurs (<i>Stakeholder</i>) de ces deux dernières ressources peuvent fournir des exigences quant à leur utilisation ou maintenance. La ressource d'entreprise fournit un ensemble d'aptitudes (<i>Capability</i>) caractérisées par des capacités (<i>Capacity</i>). Les ressources humaines détiennent un poste (<i>Workstation</i>) et sont aptes à jouer un ensemble de rôles (<i>Role</i>).
Company-Member	Collaborateur de l'entreprise : ressource (<i>EnterpriseResource</i>) humaine travaillant au sein de l'entreprise, ayant la responsabilité de mener à bien les rôles qui lui sont affectés dans le cadre de son poste de travail (<i>Workstation</i>). Tous les collaborateurs de l'entreprise sont également considérés comme des parties prenantes (<i>Stakeholder</i>) exprimant leur besoins d'informatisation.
Capability	Le potentiel (ou aptitude) d'une ressource est défini en termes d'activités ou d'actions (<i>EnterpriseProcessor</i>) qu'elle permet de réaliser ou dans lesquelles elle est utilisée. Il peut être limité par des capacités (<i>Capacity</i>) en termes de temps, d'espace, de volume, etc. Ce sont des informations à prendre en compte pour une utilisation efficace de la ressource.

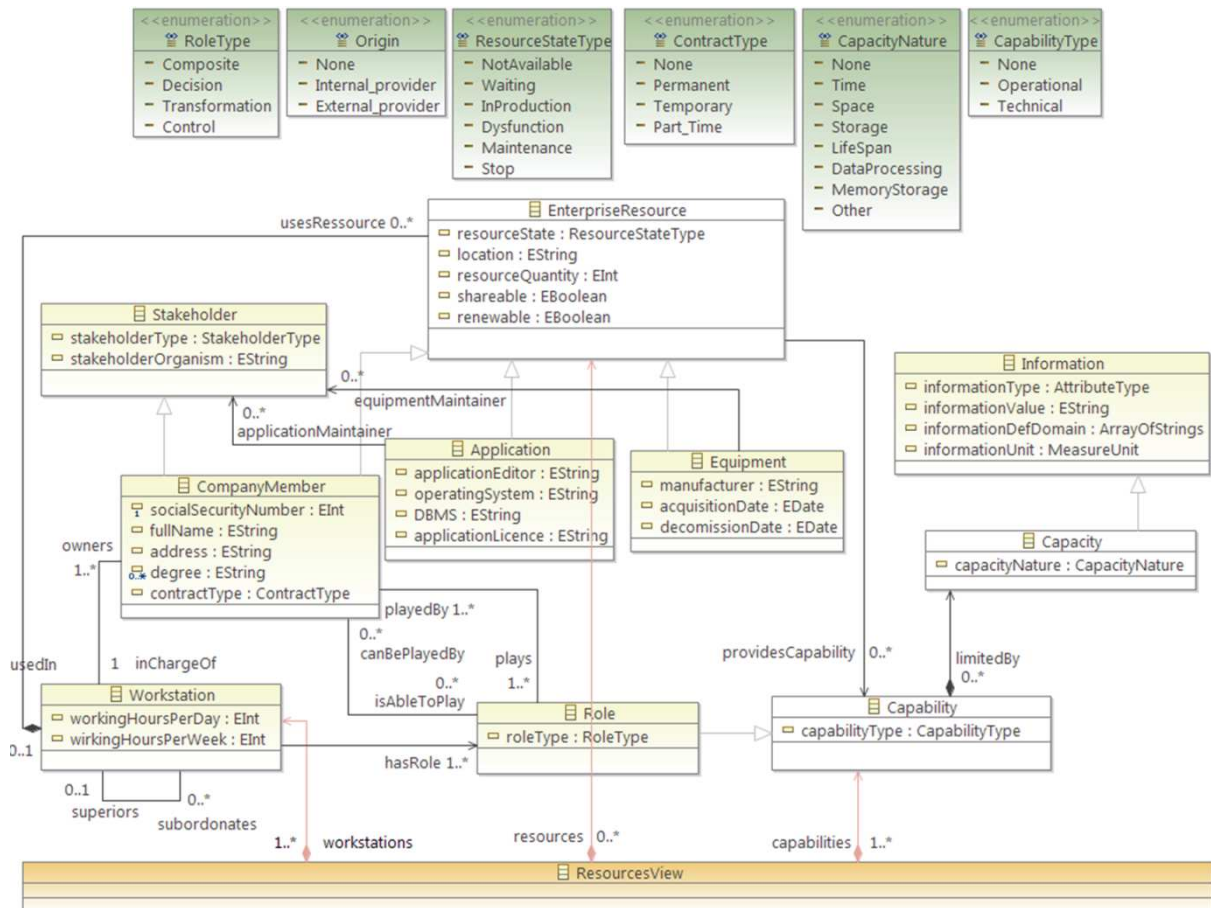


Figure 14. Concepts de la vue des ressources.

3.3.4 Vue fonctionnelle

C'est à ce niveau que le fonctionnement de l'entreprise est précisé. Les rôles joués par les collaborateurs de la PME, l'interaction avec les partenaires externes, l'utilisation des ressources, etc. sont définis en manipulant et en instanciant les concepts suivants.

Tableau 5. Concepts de la vue fonctionnelle.

Désignation du concept	Définition
EnterpriseProcessor	Le processeur est une généralisation d'autres concepts : l'acte de gestion (<i>BusinessAction</i>), l'activité (<i>Activity</i>) et le processus (<i>BusinessProcess</i>). C'est le lieu de production de produits et de services de l'entreprise (<i>EnterpriseProduct</i> /<i>EnterpriseService</i>) en consommant et produisant des objets d'entreprise, représentés par une vue spécifique (<i>EnterpriseObjectView</i>) (telle que perçue par le collaborateur qui réalise l'activité) et ce en utilisant un ensemble de ressources (<i>EnterpriseResource</i>) et en prenant en compte un ensemble de règles et de conseils (<i>BusinessGuidance</i>). Les processeurs peuvent être déclenchés par des notifications (<i>Notification</i>) et en générer, à la fin de leur réalisation, à destination d'autres processeurs ou partenaire externes.
Activity	Les activités définissent les responsabilités d'un collaborateur (<i>Enter-</i>

	<i>priseMember</i>) selon les rôles (<i>Role</i>) qu'il déteint. Les activités définissent donc ce qui est réalisé au niveau de chaque poste de travail (<i>Workstation</i>). C'est aussi le lieu d'utilisation de ressources, de transformation et de production d'objets d'entreprise (<i>EnterpriseObject-View</i>). L'activité est délimitée par deux notifications (<i>Notification</i>) : une notification déclenchante et une notification générée. Une activité peut être détaillée en acte de gestion (<i>BusinessAction</i>). Elle peut également se réaliser en collaboration avec d'autres membres de l'entreprise.
BusinessAction	L'acte de gestion fournit un niveau de détail plus fin pour définir les étapes de réalisation d'une activité (<i>Activity</i>). Il représente une action élémentaire non interrompue, réalisée par un seul collaborateur (<i>EnterpriseMember</i>) sur un seul objet d'entreprise (<i>EnterpriseObject</i>) par exemple, pour le consulter, le transformer, le créer ou le détruire.
Role	Le rôle représente l'ensemble des activités réalisées par un collaborateur (<i>EnterpriseMember</i>) détenant un poste de travail (<i>Workstation</i>). Ces activités sont relatives à un domaine d'activité (<i>Domain</i>) donné ou sont réalisées dans le cadre de sa participation à des processus (métiers, supports ou décisionnels) ayant un objectif commun. En effet, les collaborateurs de la PME peuvent avoir plusieurs rôles distincts. Par défaut, chaque collaborateur détient au moins un rôle. La notion de rôle spécialise le concept de (<i>capability</i>) de la norme ISO 19440. Le rôle est pour la ressource humaine ce que la (<i>capability</i>) est pour la ressource matérielle. Les partenaires externes (<i>ExternalPartner</i>) peuvent être impliqués au niveau des processus métiers ou supports de l'entreprise et agir sur le système. Des rôles peuvent donc leur être affectés.
Notification	Une notification est un événement susceptible de déclencher un processeur de type activité (<i>Activity</i>) ou processus (<i>BusinessProcess</i>). Les notifications déclenchantes font référence à un aspect temporel ou définissent les conditions permettant à l'activité de s'effectuer comme par exemple la réception d'un objet d'entreprise à partir d'un autre poste ou à partir d'un partenaire externe (<i>ExternalPartner</i>). Les notifications générées représentent la fin de la réalisation d'une activité ou le résultat d'un processus.
BusinessProcess	Les processus regroupent un ensemble d'activités (<i>Activity</i>) qui s'enchaînent, en décrivant le flux de contrôle dans l'entreprise. Les activités se déroulent en série ou en parallèle, exécutées par des collaborateurs de l'entreprise (<i>EnterpriseMember</i>) ou des partenaires externes (<i>ExternalPartner</i>). Tout processus est délimité par une notification initiatrice, liée à un partenaire externe, ou à un événement.

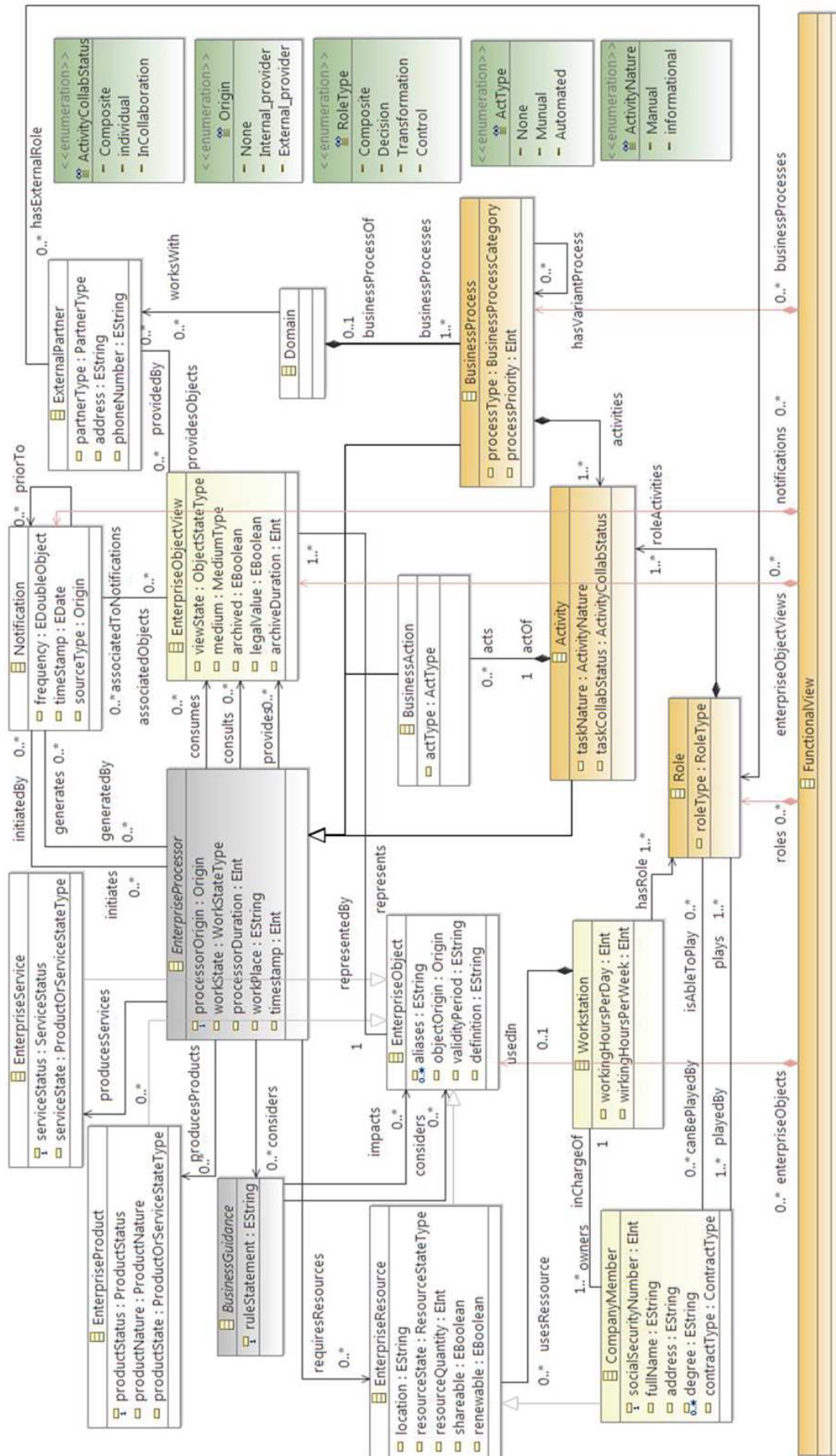


Figure 15. Concepts de la vue fonctionnelle.

3.3.5 Vue informationnelle

Cette vue regroupe toutes les informations relatives aux entités qui circulent entre les activités, les postes, etc., durant le fonctionnement, ou aux entités qui décrivent des propriétés particulières de l'entreprise. L'objet d'entreprise (*EnterpriseObject*) représente l'élément principale de cette vue, il regroupe tous les éléments qui peuvent donner lieu à une modélisation plus fine lors de la conception du modèle conceptuel de données associé aux applicatifs à réaliser lors du projet d'informatisation de la PME.

Tableau 6. Concepts de la vue informationnelle.

Désignation du concept	Définition
EnterpriseObject	L'objet d'entreprise représente des éléments caractérisant l'entreprise et son environnement et permettant de la différencier. Il décrit une généralité ou une entité réelle ou abstraite pouvant être conceptualisée comme étant un tout. Il se réfère à des objets informationnels ou physiques. L'objet d'entreprise regroupe différents éléments du modèle de l'entreprise : les éléments en entrée, en sortie ou manipulés au niveau des activités, les ressources, les produits, les services, les partenaires externes, Les entrées externes et les sorties externes. Il regroupe également des éléments qui servent à structurer l'entreprise : les domaines, les cellules organisationnelles, les postes et les rôles. Les objets d'entreprise qui sont en entrée ou en en sortie des processeurs (<i>EnterpriseProcessor</i>) ou associés à des notifications sont représenté par des vues d'objet d'entreprise (<i>EnterpriseObjectView</i>).
EnterpriseObjectView	La vue d'objet d'entreprise représente toute trace ou manifestation informationnelle des objets d'entreprise (<i>EnterpriseObject</i>). Elle est constituée d'un ensemble de propriétés ou attributs (<i>Information</i>) la définissant. Les vues d'objets peuvent être regroupées en dossier (<i>File</i>). Ce dernier représente un format souvent échangé entre collaborateurs et avec les partenaires pendant les processus de gestion de l'entreprise. NB : Selon l'ISO 19440, la vue de l'objet de l'entreprise est construite selon une vue particulière perçue de l'objet d'entreprise requise par une application ou un utilisateur. C'est ce qui est utilisé en entrée ou sortie ou associés à des notifications des processeurs (<i>EnterpriseProcessor</i>)
Information	L'information est l'élément élémentaire définissant une vue d'objet d'entreprise (<i>EnterpriseObjectView</i>). Elle spécifie une propriété que possède l'objet d'entreprise représenté par la vue d'objet.
AccessRight	L'habilitation définit le droit qu'un collaborateur (<i>CompanyMember</i>) possède au regard d'une vue d'objet ou d'une information. Elle peut être absolue ou dépendante d'une configuration décrite au niveau d'une règle métier.

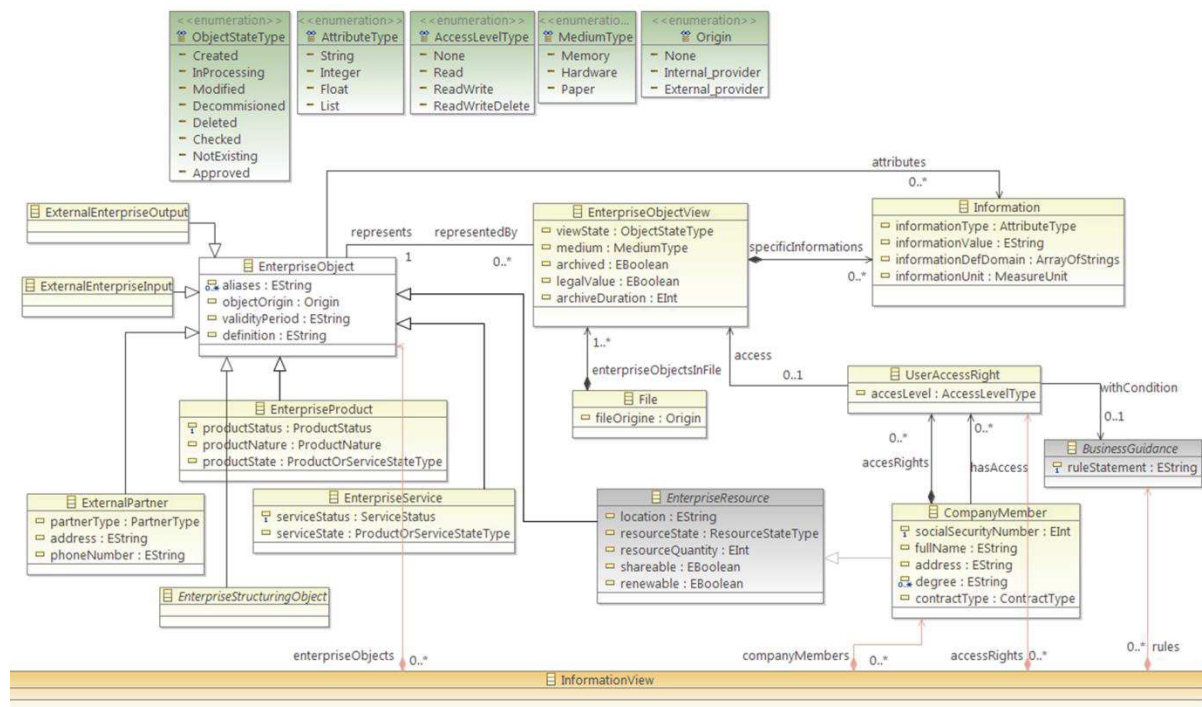


Figure 16. Concepts de la vue informationnelle.

3.3.6 Vue des exigences

Cette vue supporte la définition du référentiel des exigences issues de l'analyse et de la confortation des besoins. Ce dernier est donc défini sur la base des besoins collectés. En effet, les modèles que l'on peut construire selon les vues de modélisation définies précédemment décrivent de fait certains aspects de l'entreprise relatifs à son fonctionnement et aux contraintes prises en compte en termes d'informations, de ressources, d'organisation interne et de relations externes. Ces modèles donnent ainsi une vision des besoins de la MOA sans pour autant les séparer et les formaliser comme il se devrait dans un référentiel d'exigences. Les activités, postes, et rôles sont par exemple un moyen de décrire le fonctionnement attendu et donnent des indications sur le SI à développer pour supporter ces activités. On voit alors apparaître des besoins fonctionnels qui donnent lieu à des exigences fonctionnelles que nous représentons par le concept *Requirement*. De même, des exigences non fonctionnelles peuvent également être exprimées par les parties prenantes du projet (MOA et MOE) comme par exemple des contraintes sur les ressources allouées à certaines activités, etc. Elles sont représentées par le concept *NonFunctionalRequirement*. Nous recensons également au niveau de la vue des exigences les informations qui peuvent potentiellement être supportées par le système à réaliser. Ce sont les objets d'entreprise qui représentent des concepts du métamodèle SI-PME qui peuvent donner lieu à une modélisation plus fine lors de l'élaboration des modèles conceptuels liés aux applicatifs à réaliser. Ces exigences informationnelles sont issues des vues du contexte, informationnelle, organisationnelle et des ressources et reprennent des concepts tels que : les objets d'entreprise, produits, services, partenaires, postes, ressources etc.

Tableau 7. Concepts de la vue des exigences.

Désignation du concept	Définition
Stakeholder	La partie prenante participant de manière directe ou indirecte au projet d'informatisation de la PME est toute personne, groupe de personnes ou organisation pouvant influencer ou être affecté par le système informatique à réaliser ou par le processus de son élaboration. Elle exprime un ensemble de besoins. NB : les parties prenantes du projet comportent entre autres : - La PME jouant le rôle de maîtrise d'ouvrage avec un ensemble de ses collaborateurs (membres de la PME), ses partenaire externes fournissant des entrées, utilisent les résultats, ou ayant des attentes ou contraintes à prendre en compte. ex : clients, fournisseurs ou ceux fournissant des règles et des recommandations que la PME doit ou veut respecter. ex: législateurs, organismes de standardisation et d'audits - La maîtrise d'œuvre, en considérant ses décideurs, testeurs, formateurs, mainteneurs et sous traitiers. La MOE peut aussi prendre en compte des règles et des recommandations d'autres parties prenantes comme le législateur ou les organismes de standardisation.
Requirement	Une exigence représente ce qui doit être potentiellement réalisé par la MOE. Une ou des parties prenantes (<i>Stakeholder</i>) expriment chaque exigence. Nous identifions également pour chaque exigence le poste (<i>Workstation</i>) et le rôle (<i>Role</i>) concernés par l'exigence où l'activité en question est réalisée, la séquence de déroulement (<i>BusinessAction</i>). Aussi, toutes les règles métiers (<i>BusinessGuidance</i>), objets d'entreprises (<i>EnterpriseObject</i>) et ressources liés à cette exigence (<i>EnterpriseResource</i>) sont présentés.
NonFunctional-Requirement	Les parties prenantes peuvent exprimer des exigences non fonctionnelles. Elles peuvent être liées aux exigences fonctionnelles, telles que : des contraintes liées aux ressources utilisées, la performance attendue du système lors de l'exécution de la fonctionnalité ou encore la fréquence avec laquelle elle doit être réalisée. Elles peuvent également être exprimées indépendamment, et être relatives aux exigences non-fonctionnelles communément recensées pour les applicatifs logiciels (Chung et al. 2012) aux contraintes liées à la conduite du projet (délai et coût) ou à des propriétés des applicatifs à réaliser telles que la sécurité, l'interopérabilité, la portabilité, etc.

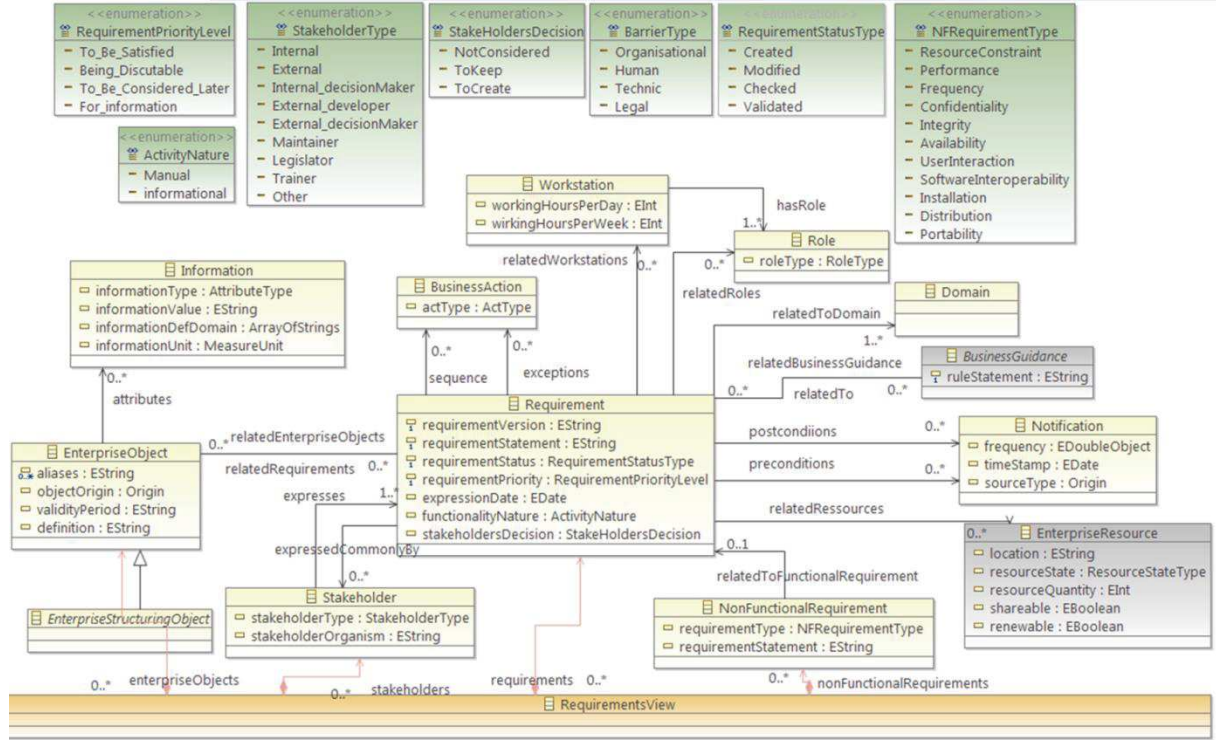


Figure 17. Concepts de la vue des exigences.

3.4 Règles de modélisation

Comme expliqué au chapitre II, des normes du domaine de l'IBE (INCOSE, 2012, IEEE 1993) définissent la notion de qualité d'une exigence et d'un référentiel d'exigences. Ces normes préconisent que chaque exigence doit être : nécessaire, indépendante de l'implémentation, complète, non ambiguë, singulière, identifiable ou traçable, faisable, vérifiable, correcte, conforme, concise et communicable. Le référentiel des exigences doit ensuite être structuré, cohérent, faisable, limité, non-redondant, priorisé et complet.

Nous expliquons au chapitre V comment certaines de ces qualités sont vérifiées au travers des choix de conception des composantes qui constituent la méthode proposée, à savoir : les concepts, les langages, le processus systématisé et l'outillage. Ici, nous définissons des règles qui contraignent les concepts du métamodèle SI-PME. Elles constituent un premier moyen pour affirmer l'existence ou l'absence de certains défauts inhérents aux modèles de besoins : l'omission, la redite et l'incohérence. Ces règles contribuent ainsi à la complétude, la non-redondance et la cohérence d'un référentiel d'exigences.

Pour une meilleure compréhension du niveau d'application de ces règles, nous schématisons le modèle des besoins recueilli, que nous appelons $M(E)$ E étant une PME particulière (Figure 18), selon chaque vue vue_i des vues de la modélisation, tel que $vue_i \in Vues = \{contexte, organisationnelle, informationnelle, ressources, fonctionnelle, exigences\}$, et selon ce qui est décrit par chaque collaborateur appartenant à l'ensemble des collaborateurs $collaborateur(E)$ de la PME. $M(E)$ est constitué de sous modèles chacun résultant de

la description de chaque collaborateur et ce relativement à chaque vue.

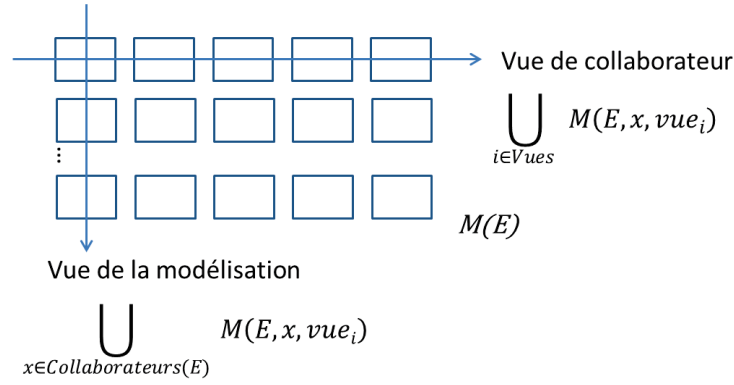


Figure 18. Représentation d'un modèle d'entreprise selon les vues de la modélisation et les vues des collaborateurs.

Le modèle de l'entreprise $M(E)$ est donc défini comme étant l'union des descriptions de chaque collaborateur appartenant à $Collaborateurs(E)$: où chaque description d'un collaborateur x représente l'union de ces descriptions relatives à chaque vue $M(E, x) = \bigcup_{i \in Vues} M(E, x, vue_i)$. Notons que plusieurs collaborateurs peuvent définir les mêmes éléments du modèle de l'entreprise $M(E, x) \cap M(E, y) \neq \emptyset$ et que les modèles partiels de la PME dans différentes vues de modélisation peuvent référer des éléments communs : $M(E, x, vue_i) \cap M(E, x, vue_j) \neq \emptyset$.

Vérifier le respect de certaines règles de modélisation dans le modèle de l'entreprise revient à vérifier les modèles de chaque collaborateur et de ce qui est défini en commun par plusieurs collaborateurs et ce relativement à chaque vue.

Nous listons dans la suite l'ensemble des règles de modélisation non explicites au niveau du métamodèle SI-PME. Elles sont classées relativement à la qualité qu'elles favorisent. Les règles sont groupées par vue (ex. $M(E, x, vue_{fonctionnelle})$) ou sont communes à deux vues (ex. $M(E, x, vue_{fonctionnelle}) \cap M(E, x, vue_{informationnelle})$). Elles sont également relatives au modèle défini par un seul collaborateur (ex. $M(E, x, vue_{fonctionnelle})$) ou défini en commun par deux collaborateurs $M(E, x, vue_{fonctionnelle}) \cap M(E, y, vue_{fonctionnelle})$.

Nous montrons des exemples d'utilisation de ces règles au niveau de la définition de la démarche proposée au chapitre V. Elles sont traduites en contraintes OCL⁹ pour vérifier leur validité.

3.4.1 Règles favorisant la complétude

$M(E, x, vue_{fonctionnelle})$

- R1_{complétude_fonctionnelle} : si l'activité est détaillée en actes de gestion alors il existe au moins un acte de gestion dans le détail d'une activité agissant sur les objets d'entreprise en entrée ou en sortie de l'activité.

⁹ Object Constraint Language

- $R2_{complétude_fonctionnelle}$: les objets d'entreprise consultés, modifié ou créés par les actes de gestion d'une activité appartiennent aux entrées, aux sorties, aux ressources ou sont associés aux notifications de cette activité.
- $R3_{complétude_fonctionnelle}$: il existe au moins un acte de gestion dans le détail d'une activité agissant sur les objets d'entreprise associé aux notifications initiatrices et générées de l'activité.
- $R4_{complétude_fonctionnelle}$: Si l'activité n'est pas détaillée en actes de gestion tout objet d'entreprise en entrée ou en sortie de cette dernière doit faire partie de sa déclaration.

$$M(E, x, vue_{fonctionnelle}) \cap M(E, x, vue_{informationnelle})$$

- $R1_{complétude_fonctionnelle/informationnelle}$: les actes de gestion d'une activité agissent sur des informations définissant les objets d'entreprise en entrée ou sortie de cette activité.
- $R2_{complétude_fonctionnelle/informationnelle}$: il existe au moins un acte de gestion agissant sur chaque objet d'entreprise informationnel

$$M(E, x, vue_{fonctionnelle}) \cap M(E, x, vue_{contexte})$$

- $R1_{complétude_fonctionnelle/contexte}$: il existe au moins une notification en relation avec chaque partenaire externe de type client ou fournisseur.
- $R2_{complétude_fonctionnelle/contexte}$: tous les produits de l'entreprise sont mentionnés au moins une fois comme étant une sortie d'activité.

$$M(E, x, vue_{informationnelle})$$

- $R1_{complétude_informationnelle}$: il existe au moins un objet d'entreprise défini par l'information citée dans les habilitations selon une information discriminante.

$$M(E, x, vue_{fonctionnelle}) \cap M(E, y, vue_{fonctionnelle})$$

- $R1_{complétude_fonctionnelle/fonctionnelle}$: tous les objets d'entreprises représentant des sorties d'activités, sauf celles représentant la finalité du processus métier auquel appartiennent ces activités ou celles destinées à un partenaire externe, font l'objet d'entrées d'autres activités.
- $R2_{complétude_fonctionnelle/fonctionnelle}$: les notifications initiatrices (par réception) d'une activité ne provenant pas d'un partenaire externe sont générées par une autre activité.

$$M(E, x, vue_{organisationnelle})$$

- $R1_{complétude_organisationnelle}$: un collaborateur responsable d'une cellule organisationnelle doit occuper un poste dans cette cellule.

$M(E, x, \text{vue}_{\text{exigences}})$

- $R1_{\text{complétude_exigences}}$: si une partie prenante est de type interne, elle doit faire partie des collaborateurs de l'entreprise.

3.4.2 Règles favorisant la non-reformulation

Nous vérifions par ces règles que les termes utilisés par les différents collaborateurs pour spécifier les objets d'entreprise sont unifiés.

$M(E, x, \text{vue}_{\text{fonctionnelle}}) \cap M(E, y, \text{vue}_{\text{fonctionnelle}})$

- $R1_{\text{non-reformulation_fonctionnelle/fonctionnelle}}$: chaque objet d'entreprise manipulé au niveau de la définition du fonctionnement, lié aux entrées, aux sorties et aux notifications, porte un nom unique.

$M(E, x, \text{vue}_{\text{fonctionnelle}})$

- $R1_{\text{non-reformulation_fonctionnelle}}$: une règle de gestion associée à une activité fait référence à des éléments en relation avec cette activité, à savoir les entrées, sorties, actes de gestion ou ressources.

$M(E, x, \text{vue}_{\text{contexte}}) \cap M(E, y, \text{vue}_{\text{fonctionnelle}})$

- $R1_{\text{non-reformulation_contexte/fonctionnelle}}$: chaque partenaire externe porte un nom unique.
- $R2_{\text{non-reformulation_contexte/fonctionnelle}}$: chaque produits ou service porte un nom unique.

$M(E, x, \text{vue}_{\text{organisationnelle}}) \cap M(E, y, \text{vue}_{\text{organisationnelle}})$

- $R1_{\text{non-reformulation_organisationnelle/organisationnelle}}$: chaque poste de travail détenu par les collaborateurs porte un nom unique.
- $R2_{\text{non-reformulation_organisationnelle /organisationnelle}}$: chaque cellule organisationnelle à laquelle appartiennent les collaborateurs porte un nom unique.

$M(E, x, \text{vue}_{\text{ressources}}) \cap M(E, y, \text{vue}_{\text{fonctionnelle}})$

- $R1_{\text{non-reformulation_ressources / fonctionnelle}}$: chaque ressource porte un nom unique.

3.4.3 Règles favorisant la cohérence

$M(E, x, \text{vue}_{\text{contexte}})$

- $R1_{\text{cohérence_contexte}}$: il existe au moins une entrée ou sortie externe associée au partenaire externe de type client ou fournisseur.

$M(E, x, \text{vue}_{\text{fonctionnelle}}) \cap M(E, x, \text{vue}_{\text{contexte}})$

- $R1_{\text{cohérence_fonctionnelle/contexte}}$: les produits de l'entreprise figurent comme sortie d'au moins un processeur.

- $R2_{\text{cohérence_fonctionnelle/contexte}}$: les entrées et sorties externes associés à un partenaire externe, sont associées aux notifications, entrées ou sorties d'au moins une activité.

$M(E, x, \text{vue}_{\text{fonctionnelle}})$

- $R1_{\text{cohérence_fonctionnelle}}$: les processeurs réalisés par un seul collaborateur ne doivent pas être simultanés (les valeurs données à la propriété *timeStamp* doivent être différentes, si ces dernières sont précisées).
- $R2_{\text{cohérence_fonctionnelle}}$: la durée d'un processeur est supérieure ou égale à la somme des durées des processeurs qui le détaillent (la valeur donnée à la propriété *processorDuration* d'une activité par exemple est supérieure ou égale à la somme de celles données aux actes de gestion qui la détaillent, si ces valeurs sont précisées)

$M(E, x, \text{vue}_{\text{fonctionnelle}}) \cap M(E, x, \text{vue}_{\text{informationnelle}})$

- $R1_{\text{cohérence_fonctionnelle/informationnelle}}$: tout objet d'entreprise d'origine interne est créé au moins une fois au niveau d'un acte de gestion. Il doit figurer en sortie de l'activité à laquelle appartient cet acte de gestion.
- $R2_{\text{cohérence_fonctionnelle/informationnelle}}$: un objet d'entreprise fournit par un partenaire externe doit être d'origine externe.

$M(E, x, \text{vue}_{\text{organisationnelle}})$

- $R1_{\text{cohérence_organisationnelle}}$: une cellule organisationnelle peut être affectée à d'autres cellules organisationnelles différentes de celle-ci.
- $R2_{\text{cohérence_organisationnelle}}$: un poste de travail peut être subordonné ou supérieur à d'autres postes sauf celui-ci.

$M(E, x, \text{vue}_{\text{informationnelle}})$

- $R1_{\text{cohérence_informationnelle}}$: un objet d'entreprise informationnel provenant d'un partenaire externe, d'un collaborateur ou d'un poste externes aux limites des domaines d'activité étudiés est d'origine externe sinon il est d'origine interne.
- $R2_{\text{cohérence_informationnelle}}$: un collaborateur a accès aux informations d'au moins un objet d'entreprise.
- $R3_{\text{cohérence_informationnelle}}$: un collaborateur a accès à toutes les Informations des objets d'entreprise sur lesquels agissent les actes de gestion des activités qu'il réalise.
- $R4_{\text{cohérence_informationnelle}}$: si un dossier a une valeur légale (*legalValue*) alors il doit être archivé.

- $R5_{\text{cohérence_informationnelle}}$: si un dossier est archivé (*archiveDuration*) la durée d'archivage ne doit pas être nulle.

$M(E, x, \text{vue}_{\text{ressources}})$

- $R1_{\text{cohérence_ressources}}$: un équipement est acquis avant d'être mis hors service (*acquisitionDate* doit être inférieure *decommissionDate*)

$M(E, x, \text{vue}_{\text{fonctionnelle}}) \cap M(E, x, \text{vue}_{\text{ressources}})$

- $R1_{\text{cohérence_fonctionnelle/ressources}}$: une ressource est disponible pendant l'exécution de l'activité où elle est utilisée.
- $R2_{\text{cohérence_fonctionnelle /ressources}}$: les ressources utilisées ne dépassent pas la quantité existante.
- $R2_{\text{cohérence_fonctionnelle /ressources}}$: la capacité utilisée d'une ressource ne doit pas dépasser la capacité existante.
- $R4_{\text{cohérence_fonctionnelle /ressources}}$: toute ressource est utilisée au moins une fois au niveau des activités.
- $R5_{\text{cohérence_fonctionnelle /ressources}}$: si une activité est réalisée « en collaboration », une ressource humaine doit être utilisée.

3.5 Conclusion

Nous avons établi dans ce chapitre les concepts et relations que manipulent les utilisateurs de la méthode d'IBE. Grâce à ces notions, les utilisateurs modélisent leur environnement de travail, ce qui indique en grande partie leurs besoins d'informatisation. Ces concepts représentent un socle conceptuel dont la formalisation nous permet d'automatiser la modélisation et certaines étapes de vérification et de transformation de modèles. Un premier moyen de vérification est présenté où les contraintes de modélisation empêchent que les défauts inhérents aux besoins ne se reportent sur les exigences.

Chapitre IV : Langages supports à la Méthode d'IBE

4.1 Introduction

Nous avons présenté au chapitre III le métamodèle SI-PME regroupant les concepts auxquels les collaborateurs de la PME font référence pour définir leur environnement de travail, décrire et identifier leurs besoins d'informatisation. Ces concepts sont manipulés, instanciés et renseignés en respectant des **langages de modélisation** destinés à supporter le processus d'ingénierie des besoins et des exigences que nous proposons. Pour construire ces langages, nous détaillons dans ce chapitre leurs notations, aussi appelés syntaxes concrètes.

Les langages que nous proposons ici sont des langages de type DSL (*Domain Specific Languages*). Les langages de ce type ne sont pas destinés à un usage générique comme le sont les GPL (*General Purpose Languages*), comme par exemple UML, qui sont aptes à potentiellement être utilisés dans plusieurs domaines (Mernik et al. 2005). Nous commençons ce chapitre par exposer l'intérêt d'utiliser des DSL dans une chaîne de développement agile, puis nous présenterons les DSL proposés.

4.2 Les DSL supports de la méthode d'IBE

Comme leur nom l'indique, les DSL sont des langages conçus pour être utilisés pour un domaine particulier (Mernik et al. 2005). Ils sont souvent d'un niveau d'abstraction élevé et correspondent à l'image mentale que se font les utilisateurs du domaine étudié (Vangheluwe & Meyers 2014). Ils sont, de fait, plus faciles à appréhender et à utiliser. En effet, les DSL fournissent un gain considérable en expressivité (Pfeiffer & Pichler 2008) et surtout facilitent la communication entre MOA et MOE.

Les DSL que nous proposons (Amokrane et al. 2015) dans le cadre de la méthode d'IBE sont basés sur la syntaxe abstraite proposée au chapitre précédent. Leur définition est complétée par deux types de syntaxes concrètes qui respectent la syntaxe abstraite :

- (i) Une syntaxe concrète textuelle : il s'agit d'un langage naturel structuré (de type CNL). Il est proposé aux collaborateurs de la PME pour exprimer leurs besoins d'informatisation. En effet, nous avons constaté, suite à l'étude comparative de langages de modélisation d'exigences, présentée au chapitre II, que les langages textuels sont plus appropriés aux collaborateurs qui ne peuvent pas accéder aux notations graphiques détaillées sans formation ni assistance.
- (ii) Une syntaxe concrète graphique : trois des langages utilisés pendant la phase d'IBE sont cependant associés à ce type de syntaxe concrète pour améliorer le rendu des informations lors des restitutions faites aux parties prenantes à des fins de validation. Ce formalisme, bien que graphique, doit donc rester simple, intuitif ou encore compréhensible à moindre effort par les collaborateurs de la PME. Il est de plus approprié pour synthétiser un ensemble important d'informations.

Aussi, les praticiens de la création de DSL fournissent aujourd'hui des environnements de développement d'outils de modélisation (par exemple : Sirius, EMF, Kermeta, etc.) qui favorisent le développement agile en se basant sur des méta-métamodèles uniques permettant

les transformations de modèles et la génération de code. Ceci nous permet de faire passer aux équipes de la MOE des modèles plus facilement exploitables bien qu'incomplets.

Nous utilisons l'environnement de développement Xtext (Xtext 2010) pour créer un éditeur textuel en charge de la manipulation de DSL ayant une syntaxe concrète textuelle baptisé dans la suite DSL textuel. Nous utilisons ensuite l'environnement Sirius (Sirius 2013) développé par Obeo pour créer les éditeurs des DSL ayant une syntaxe graphique baptisés dans la suite DSL graphiques. Ces deux environnements permettent de générer des outils de modélisation basés sur la plate-forme ouverte *Eclipse Modelling Framework* (EMF).

4.2.1 Le DSL textuel

Afin de collecter les besoins d'informatisation de la PME, nous construisons une représentation qui reflète la situation de l'entreprise à informatiser. Nous permettons aux collaborateurs de la PME de partager leurs connaissances métiers par le biais d'un langage naturel structuré. Celui-ci met en œuvre les concepts définis au chapitre III en utilisant le principe de *boilerplate*. Il s'agit d'un langage textuel à base de patrons qui restreint la modélisation textuelle en proposant :

- un vocabulaire prédéfini : une partie de chaque patron textuel du *boilerplate* est fixée et les collaborateurs ne pourront s'exprimer autrement,
- une construction grammaticale prédéfinie : le patron textuel des *boilerplates* est constitué d'éléments fixes et d'un ensemble d'attributs (entre < >) paramétrables,
- une interprétation prédéfinie : la nature des attributs paramétrables est connue. Ces derniers correspondent aux concepts de la syntaxe abstraite. En les renseignant, les collaborateursinstancient ces derniers.

Les collaborateurs de la PME sont également guidés et assistés par l'entremise d'un **questionnaire d'expression des besoins**. En effet, nous avons un ensemble de questions pour couvrir les différentes vues de modélisation de l'entreprise : ses domaines d'activité, ses interactions avec des partenaires, son organisation, ses ressources et son fonctionnement. Les collaborateurs de la PME utilisent ainsi ce DSL textuel pour répondre au questionnaire.

La grammaire de ce DSL est définie avec l'environnement de développement de langages textuels Xtext. Xtext génère un éditeur textuel pour la manipulation du DSL textuel. Ce dernier est doté de fonctionnalités diverses telles que : la coloration syntaxique, l'auto-complétion de mots clés, la vérification et correction de textes et la vérification des contraintes sur le modèle. De plus, nous intégrons à l'éditeur Xtext ainsi développé des fonctionnalités de l'outil de traitement du langage naturel TreeTagger (TreeTagger 1994) pour vérifier certaines constructions grammaticales des modèles textuels. L'usage de cet environnement nous permet de guider les utilisateurs durant la phase de modélisation, ce qui est une stratégie de base de vérification de modèles. Il devient ainsi possible de nous appuyer sur un modèle d'entreprise syntaxiquement correct sur lequel les règles de modélisation présentées au chapitre III sont applicables.

En s'inspirant des principes du métamodèle SBVR (OMG 2008), les patrons textuels de *boilerplates* que nous proposons peuvent être :

- Des concepts : représentés par un nom ou une phrase nominale, ils expriment des concepts individuels (correspondant à un seul objet comme le nom d'un collaborateur) ou des types d'objets (concepts génériques classifiant des objets selon leurs propriétés communes comme un type de ressource matérielle). Leur sémantique est définie par correspondance avec les concepts de la syntaxe abstraite (Tableau 8). Par exemple, le *boilerplate* « Nos produits sont : <EnterpriseProduct>? » inclut un attribut paramétrable correspondant au concept *EnterpriseProduct* de la syntaxe abstraite.
- Des faits : caractérisés par l'utilisation d'un verbe, ils désignent une relation, une situation ou une action impliquant un ou plusieurs concepts. Les faits peuvent également représenter une caractéristique ou un état d'un concept. Pour aider à construire le modèle de l'entreprise, les *boilerplates* que nous proposons incluent des faits prédéfinis à compléter en renseignant un ou plusieurs concepts. Par exemple, le *boilerplate* « <ExternalPartner> <verbe> <ExternalEnterpriseInput> ? » renseigne sur ce que les partenaires externes de l'entreprise lui fournissent.
- Des règles métiers : construites à partir de faits, eux même construits à partir de concepts. Les règles métiers et les conseils (bonnes pratiques) prises en considération par une organisation représentent le mode de gestion de cette dernière et précisent les scénarios opérationnels. Un *boilerplate* est défini pour formaliser la définition de ces règles métiers. Comme dans SBVR, nous permettons aux collaborateurs d'exprimer l'obligation, la nécessité, la possibilité et l'interdiction avec les modalités : « Il est obligatoire », « Il est nécessaire », « Il est possible » et « Il est interdit ». Ceci donne le choix aux collaborateurs de la PME de décrire ces modes de gestion sans pour autant se préoccuper des distinctions, non intuitives, proposées par SBVR où des modalités différentes sont associées à chaque type de règles de gestion (opérationnelles ou structurelles).

La grammaire proposée permet, de plus, plusieurs formulations linguistiques au moyen d'articles, pronoms, quantifieurs, etc. pour former des phrases sémantiquement et grammaticalement correctes.

En plus d'être un moyen d'expression des besoins, ce DSL textuel permet intuitivement aux collaborateurs de construire le vocabulaire spécifique de la PME. Celui-ci est en effet constitué de l'ensemble des termes, définitions et relations qu'elle utilise oralement ou de manière écrite lors de l'exercice de son métier. En levant l'ambiguïté, la redite et l'incohérence entre les termes utilisés, nous pouvons contribuer à l'unification de ce vocabulaire métier parmi les collaborateurs de la PME.

4.2.1.1 Formalisation de la grammaire

Cette grammaire définit au final un sous ensemble restreint et contraint de la grammaire du

Français. Nous présentons ci-dessous, en pseudo code, les constructions de la grammaire nécessaires pour définir les patrons du DSL textuel. Les notations ?, * et + signifient respectivement « 0 ou 1 », « 0 ou plusieurs » et « au moins 1 ». Elles sont utilisées pour montrer les cardinalités des parties composant chaque construction de la grammaire. La barre verticale (|) symbolise les options.

SITUATION = (COMPORTEMENT) ? (QUANTIFIEUR) ?
(PHRASE_VERBALE | PHRASE_NOMINALE_COMPOSEE |
ETAT | (PREPOSITION)? ACTION | (quantité) ? TEMPS)

PHRASE_VERBALE = (SUJET | PRONOM) ACTION

ACTION_UNITAIRE = verbe COMPLEMENT

ACTION = verbe (COMPLEMENT)? (CONNECTEUR COMPLEMENT)*

ETAT = (SUJET | PRONOM) ('ne')? VERBE_ETRE ('pas')? (adjectif
(GROUPE_NOMINAL)* | (COMPARATEUR (quantité) ?
PHRASE_NOMINALE_COMPOSEE))

COMPLEMENT = PHRASE_NOMINALE_SIMPLE

SUJET = PHRASE_NOMINALE_SIMPLE

PHRASE_NOMINALE_COMPOSEE = PHRASE_NOMINALE_SIMPLE (CONNECTEUR
PHRASE_NOMINALE_SIMPLE)*

PHRASE_NOMINALE_SIMPLE = (ARTICLE)? (QUANTIFIEUR)? (quantité)? nom
(GROUPE_NOMINAL | INFINITIF)*

GROUPE_NOMINAL = PREPOSITION (QUANTIFIEUR) ? (ARTICLE)? ((QUANTITE)?
nom | quantité) (CONNECTEUR (ARTICLE)? ((quantité)?
nom | quantité))*

FREQUENCE_TEMPS = (quantité)? (FREQUENCE | QUANTIFIEUR) (ARTICLE)?
(quantité)? TEMPS

INFINITIF = PREPOSITION verbe

MODALITE = 'Il est obligatoire' | 'Il est nécessaire' | 'Il est possible' | 'Il
est interdit'

...

VERBE_ETRE = 'suis' | 'es' | 'est' | 'sommes' | 'sont' | 'soit' | 'soient' |
'être'

TEMPS = 'seconde' | 'minute' | 'heure' | 'matin' | 'jour' | 'se-
maine' | 'mois' | 'année' | 'secondes' | 'minutes' |
'heures' | 'jours' | 'semaines' | 'années'

COMPARATEUR = 'supérieur à' | 'supérieur ou égal à' | 'inférieur à' | 'inférieur ou égal à' | 'égal à' | 'supérieure à' | 'supérieure ou égale à' | 'inférieure à' | 'inférieure ou égale à' | 'égale à'

PREPOSITION = 'à' | 'du' | 'd' | 'pour' | 'de' | 'vers' | 'dans' | 'au' | 'aux' | 'sans' | 'hors' | 'dès' | 'par' | 'avec' | 'sur' | 'en' | 'chez' | 'de chez' | 'entre' | 'contre' | 'devant' | 'derrière' | 'sous' | 'envers' | 'près de' | 'auprès de' | 'à travers' | 'autour de' | 'en face de' ...

COMPORTEMENT = 'avant' | 'avant que' | 'après' | 'après que' | 'pendant' | 'pendant que' | 'durant' | 'durant que' | 'au cours de'

QUANTIFIEUR = 'tout' | 'toute' | 'chaque' | 'aucun' | 'plusieurs' | 'au moins' | 'au plus' | 'exactement'

FREQUENCE = 'fois par' | 'une fois par' | 'chaque fois' | 'plusieurs fois par' | 'une seule fois par' | 'au moins une fois par' | 'au plus une fois par' | 'exactement une fois par' | 'un seul' | 'une seule'

ARTICLE = 'un' | 'le' | 'l' | 'une' | 'la' | 'des' | 'les'

PRONOM = 'je' | 'tu' | 'il' | 'elle' | 'nous' | 'vous' | 'ils' | 'elles'

CONNECTEUR = 'et' | 'ou'

Dans la grammaire proposée, l'élément *quantité* représente une valeur numérique. Nous ne décomposons pas davantage le nom, verbe et adjectif. Pour cela, TreeTagger (TreeTagger 1994) est utilisé pour l'étiquetage morphosyntaxique (*part-of-speech tagging*) qui consiste à associer aux mots composant le texte entré par les collaborateurs, les informations grammaticales correspondantes comme la partie du discours : nom, verbe, article, adverbe, etc. ou encore le genre et le nombre. De cette manière, nous pouvons vérifier la correction grammaticale des modèles textuels. Dans le tableau ci-dessous, nous associons aux attributs des patrons textuels des *boilerplates*, qui représentent des concepts de la syntaxe abstraite, des formulations rendues possibles grâce à la définition de la grammaire.

Tableau 8. Formulations associées aux concepts de la syntaxe abstraite.

Concept de la syntaxe abstraite	Formulations
Domain, ExternalEnterpriseInput ExternalEnterpriseOutput, EnterpriseProduct, EnterpriseService, OrganisationalCell	PHRASE_NOMINALE_COMPOSEE
Enterprise, ExternalPartner, Workstation, EnterpriseResource, CompanyMember, EnterpriseObject, File, Information, Stakeholder	PHRASE_NOMINALE_SIMPLE
Role, Activity	PHRASE_NOMINALE_SIMPLE ACTION

4.2.1.2 Boilerplates d'expression des besoins d'informatisation

Nous présentons à présent les patrons textuels des *boilerplates*. Ils sont utilisés pour répondre au questionnaire d'expression des besoins. Les *boilerplates* sont définis sur la base de la grammaire. Leur définition fait également référence aux concepts de la syntaxe abstraite dont les formulations sont définies dans le Tableau 8. Les patrons textuels sont constitués d'attributs à renseigner entre « < > » et d'éléments fixes de syntaxe qui sont ici soulignés. Les notations ?, * et + sont également utilisés.

Le questionnaire est décomposé en plusieurs parties, présentées dans ce qui suit, chacune utilisée pour éliciter une vue particulière du modèle de l'entreprise.

- la vue du contexte dans les parties « **Quelles sont les lignes directrices de votre entreprise ?** » et « **Qui sont vos partenaires ?** »
- la vue organisationnelle dans la partie « **Quelle est votre organisation ?** »
- la vue des ressources dans la partie « **Quelles sont vos ressources ?** »
- La vue fonctionnelle est élicitée poste par poste dans la partie « **Comment fonctionnez-vous ?** »

Les éléments des vues informationnelle et des exigences sont déduits à partir de ce qui est recensé comme informations par les différentes questions. Ils sont restitués lors de l'étape de validation des exigences présentée au chapitre V.

Nous présentons dans la suite les différentes parties du questionnaire adressé aux collaborateurs de la PME et présentons la définition des *boilerplates* de réponses faisant partie du DSL textuel. Nous donnons aussi des exemples issus du cas de la PME SDT pour montrer l'instanciation du DSL textuel. La démarche opératoire où le questionnaire est utilisé est présentée au chapitre V.

Quelles sont les lignes directrices de votre entreprise ?

Nous élicitons dans cette partie les domaines d'activité concernés par l'étude. La politique que l'entreprise entreprend pour mener à bien son activité et ses principaux objectifs stratégiques.

- Quelle est l'activité de l'entreprise ?

<Enterprise> (a pour activité | fait (du | de la)) : (<Domain> (↓)?)+

Ex : SDT a pour activité : Micronisation et analyse à façon des poudres fines, Formation continue, Expertise et Conseil.

- Quels sont vos objectifs stratégiques ? Associez, s'ils existent, des indicateurs objectifs.

(<ACTION> (↓)?)*

Ex : Minimiser le temps de réponse et le temps de réalisation.

Qui sont vos partenaires ?

Cette partie du questionnaire permet d'indiquer le contexte de l'entreprise d'un point de vue externe en précisant les partenaires externes susceptibles de stimuler un changement dans le fonctionnement de l'entreprise. Les échanges avec ces derniers et les produits/services que fournit l'entreprise.

- Quels sont les types de partenaires avec qui vous travaillez ?

(<ARTICLE>? (clients|fournisseurs|prestataires|partenaires) <VERBE_ETRE> :
(<ExternalPartner> (␣)?)+)

Ex : les prestataires sont : les transporteurs.

Ex : Les partenaires sont : les partenaires de recherche.

- Qu'est-ce que les partenaires vous fournissent ?

(<ExternalPartner> <verbe> (<ExternalEnterpriseInput> (␣)?)+)*

Ex : les transporteurs effectuent le transport des palettes et des colis aux clients.

- Que mettez-vous à disposition de vos partenaires ?

(<Enterprise> verbe (<ExternalEnterpriseOutput> (␣)?)* PREPOSITION (<ExternalPartner> (␣)?)+)

EX : SDT propose des formations aux industriels.

- En plus de ces échanges, veuillez préciser vos produits et services.

(Nos produits sont : (<EnterpriseProduct> (␣)?)+)?

(Nos services sont : (<EnterpriseService> (␣)?)+)?

Nos services sont : la micronisation, l'analyse de poudres, le broyage, le tamisage, la classification des poudres, le mélange de poudres, la formation, le conseil.

Quelle est votre organisation ?

Les questions de cette partie permettent de définir la décomposition organisationnelle de l'entreprise en cellules organisationnelles et en postes.

- Quelles sont les structures et sous-structures qui constituent l'entreprise ?

((<Enterprise> | <OrganisationalCell>) est structuré
en : (<OrganisationalCell> (␣)?)+)*

Ex : SDT est structurée en : direction et administration, production, marketing et commercial, recherche et développement.

Ex : Production est structurée en : atelier et maintenance.

- Quels sont les postes qui constituent chaque structure ?

((<Enterprise> | <OrganisationalCell>) regroupe les postes : ((quantité)? <Workstation> (⌵)?)+)+

Ex : Production regroupe les postes : responsable de production.

Ex : Atelier regroupe les postes : chef d'atelier, 2 responsables d'équipe, 5 opérateurs.

Quelles sont vos ressources ?

Les questions de cette partie permettent de recenser les ressources humaines et toutes autres ressources nécessaires pour mener à bien les activités au niveau de chaque poste.

- Qui sont les collaborateurs qui détiennent chaque poste ?

(<Workstation> est occupé par (<CompanyMember> (" , ")?)+)*

Ex : directeur général est occupé par François Dupond

- Quelles sont les ressources applicatives ou matérielles (documents, équipements, machines, robots, etc.) nécessaires à la réalisation des activités au niveau de chaque poste ?

((<Workstation> (⌵)?)+ nécessite les ressources : (<Enterprise-Resource>(⌵)?)+)*

Ex : Opérateur nécessite les ressources : broyeur, tamiseur, mélangeur, équipements de laboratoire, moyens de manutention.

Comment fonctionnez-vous ?

Dans cette partie du questionnaire le fonctionnement de l'entreprise est précisé. Les rôles joués par les collaborateurs de la PME relativement à chaque poste sont définis et l'ensemble des activités qui y sont réalisées sont précisées.

➤ Considérons le poste « responsable de production »

- Quels sont les rôles attribués à ce poste ?

(<Role> (⌵)?)+

Ex : La gestion de la production, le suivie de la qualité, le respect des consignes de sécurité.

- Listez les principales activités liées à chaque rôle.

(<Role> ⌵ (<Activity> (⌵)?)+)+

Ex : La Gestion de la production : planification des commandes, gestion de la ressource humaine, prise en compte des demandes, gestion des problèmes, création des rapports.

De plus les questions sont posées de façon à définir pour chaque activité : les vues d'objets d'entreprise en entrée et en sortie, les notifications déclenchantes et générées, le déroulement de l'activité en actes de gestions, les ressources manipulés et l'ensemble des règles considérées lors de la réalisation de l'activité. La Figure 19 résume la définition de l'activité et précise à quoi correspondent les questions suivantes.

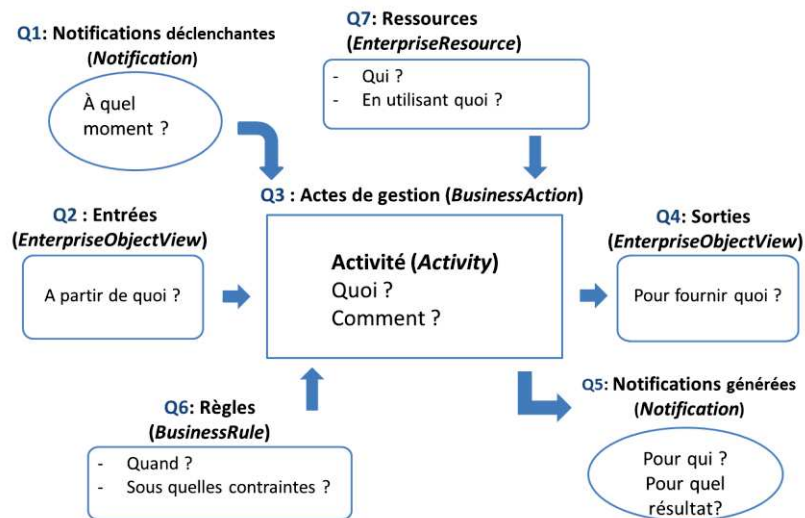


Figure 19. Questions intervenant dans la définition d'une activité.

➤ Considérons l'activité « planification des commandes »

- **Q1** : Quand est-ce que vous réalisez cette activité ?

((réception | arrivée | réceptionner | recevoir | Lorsque je reçois) (PREPOSITION)? <EnterpriseObject> (PREPOSITION)? (<OrganisationalCell> | <Workstation> | <CompanyMember> | <ExternalPartner>)) | SITUATION | ETAT | FREQUENCE_TEMPS)*

Ex : Arrivée de la commande client

- **Q2** : Quels sont les informations / dossiers / produits utilisés, consultés ou transformés durant cette activité ?

(<EnterpriseObjectView> { (<Information> (_)?)* })*

Ex : La commande {produit, quantité, date de livraison souhaitable, CDC technique}

Ex : Planification des opérateurs et des machines.

- **Q3** : Veuillez détailler les étapes de réalisation de cette activité.

```
((<verbe> <EnterpriseObjectView> (,)? ) | ((Si|lorsque) (AC-  
TION|PHRASE_VERBALE|ETAT) (alors|,) (ACTION|PHRASE_VERBALE| ETAT)))+
```

Ex : Choisir les machines, choisir les opérateurs, planifier, définir les étapes du traitement, créer le dossier de production.

- **Q4** : Quels informations / dossiers / produits / services sont produits durant de cette activité ?

```
(<EnterpriseObjectView> { (<Information> (,)?)* })*
```

Ex : Dossier de production

- **Q5** : Qu'est-ce que vous faites de ce résultat ?

```
(((<Transmettre|signaler|envoi|envoyer> (PREPOSITION)? <EnterpriseObject>  
(PREPOSITION)? (<OrganisationalCell> | <Workstation> | <CompanyMember> | <Ex-  
ternalPartner>))) | (<EnterpriseObject> VERBE_ETRE (envoyé|transmis) (PREPOSI-  
TION)? (<OrganisationalCell> | (<Workstation> | <CompanyMember> | <External-  
Partner>))) | SITUATION | ETAT | FREQUENCE_TEMPS)*
```

Ex : le Dossier de production est envoyé à l'atelier.

- **Q6** : Quelles ressources matérielles, applicatives ou informationnelles sont nécessaires pour la réalisation de cette activité ?

```
(<EnterpriseResource> (,)?)+
```

Ex : Le tableau de planification

- **Q7** : Quelles sont les règles métiers ou consignes que vous respectez pour réaliser cette activité ?

```
(((<Modalité>? (<Si|lorsque>) (ACTION|PHRASE_VERBALE|ETAT) (alors|,) (AC-  
TION|PHRASE_VERBALE| ETAT) (FREQUENCE_TEMPS)* (SITUATION)* ) |
```

```
((<Modalité>? (ACTION|PHRASE_VERBALE|ETAT) ((Si|lorsque) (ACTION |  
PHRASE_VERBALE | ETAT)) ? (FREQUENCE_TEMPS)* (SITUATION)*)))
```

Ex : Il est nécessaire que la date de fin de réalisation soit inférieure à la date de livraison souhaitable.

4.2.2 Les DSL graphiques

Des modèles à la syntaxe concrète graphique sont utilisés pour proposer des restitutions aux collaborateurs une fois qu'ils ont répondu aux questionnaires. Il s'agit ici de reformuler leurs

réponses et de restituer les éléments recueillis pour confirmer leur validité. Ceci est une pratique courante lors des interviews.

Les modèles graphiques sont plus parlants pour présenter les relations entre éléments comme, par exemple, les liens d'appartenance ou hiérarchiques. Ils permettent donc de représenter en une seule vue plusieurs informations. La revue de ces modèles par des collaborateurs experts permet alors de constater certains défauts tels que l'oubli ou la redite. Nous proposons pour cela trois DSL simples et éprouvés :

- Un organigramme. Il s'agit d'un diagramme connu par les collaborateurs en entreprise.
- Un diagramme de contexte, usuellement utilisé lors de la conception préliminaire. Par exemple, dans l'étape « capture initiale des besoins », la méthode 2TUP (Roques & Vallée 2011) montre avec ce diagramme l'utilisation ou les attentes d'utilisateurs vis-à-vis du système à réaliser. Nous adaptons ce diagramme pour définir l'environnement et les échanges externes de la PME.
- Un diagramme de processus simplifié, inspiré dans ces grandes lignes par le langage BPMN (OMG 2011).

Nous présentons dans la suite les éléments de la syntaxe concrète de chacun de ces trois DSL graphiques. Nous introduisons des icônes pour représenter les concepts et relations qui peuvent être manipulés grâce à ces DSL graphiques. Elles facilitent la lecture et la compréhension par les collaborateurs non experts en donnant plus de sens aux boîtes et aux flèches des diagrammes. Nous expliquons ensuite les transformations nécessaires à leur construction. Des exemples de ces DSL sont présentés aux chapitre V.

4.2.2.1 Diagramme de contexte

Les diagrammes de contexte permettent de représenter l'entreprise vue de l'extérieur. Il s'agit de graphes recensant d'une part les liens entrants et sortants représentent les échanges qu'a l'entreprise avec ses partenaires externes, ou d'autre part les produits et services qu'elle propose. Ces diagrammes permettent de situer l'entreprise dans son environnement et de distinguer ses partenaires les plus importants.

Tableau 9. Eléments de syntaxe concrète du diagramme de contexte.

Désignation du construit ¹⁰	Elément graphique associé
--	---------------------------

¹⁰ Un construit correspond à un concept, un attribut ou une relation du métamodèle.

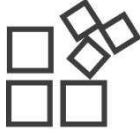
Enterprise	
ExternalPartner	
ExternalEnterpriseInput	
ExternalEnterpriseOutput	
EnterpriseProduct	
EnterpriseService	

4.2.2.2 Organigramme et allocation de ressources

L'organisation interne de l'entreprise en structures, sous structures et postes de travail est schématisée par un organigramme. Nous permettons également de présenter au niveau de cet organigramme les ressources humaines, matérielles et applicatives citées par les collaborateurs au niveau du détail de chaque activité. Ces ressources peuvent être allouées à chaque poste.

Tableau 10. Eléments de syntaxe concrète de l'organigramme

Désignation du construit	Elément graphique associé
OrganisationalCell	
Assigned OrganisationalCell	
Workstation	

Subordinates	
CompanyMember	
EnterpriseResource	
usesResources	

4.2.2.3 Diagramme de processus

Pour une gestion efficace de leur activité et pour plus de réactivité et d'agilité, les PME doivent considérer leurs processus de bout en bout. Ceci suppose une approche transversale qui dépasse les frontières des cellules organisationnelles ou des postes. En effet, qu'il s'agisse de fabriquer un produit ou de fournir un service, les processus permettent de coordonner les activités entre collaborateurs et partenaires pour aboutir à un résultat. L'efficacité avec laquelle une entreprise exécute ses processus détermine souvent son succès, c'est pourquoi la maîtrise des processus est vitale.

Les processus impliquent plusieurs collaborateurs et peuvent être répartis entre différents domaines. Dans l'étape d'expression des besoins, la définition du fonctionnement se fait par poste. Elle ne nécessite pas d'avoir une vision globale des processus dans lesquels le collaborateur détenteur du poste est impliqué. Nous suivons donc volontairement une approche ascendante pour déduire les processus et ne pas avoir à les définir de façon directe.

Les processus transversaux aux postes et aux cellules organisationnelles font apparaître plusieurs éléments : le déroulement des activités de l'entreprise en réponse aux événements déclenchants, les scénarios possibles de ce déroulement, les flux des données et les échanges avec les partenaires externes. Le diagramme que nous utilisons présente horizontalement les postes de travail intervenant dans le processus. Ces derniers englobent les activités qui y sont réalisées. Les activités sont liées par des flux d'objets, représentant la circulation des objets d'entreprise, ou des flux de contrôles montrant les possibilités de déroulement des activités. Le flux de contrôle est délibérément simple. Des notifications déclenchantes et générées annoncent le début et la fin du processus et mettent en évidence les liens avec les partenaires externes.

4.2.2.4 Construction des DSL graphiques

Les modèles sont construits à partir de transformations effectuées sur les modèles textuels de réponses collectées. L'éditeur textuel pour la manipulation du DSL textuel, généré par Xtext, construit des modèles (instances) conformes à la grammaire et aux *boilerplates* du DSL textuel. Le métamodèle Ecore sous-jacent de ce DSL est défini sur la base du graphe de la syntaxe abstraite lié à la définition de la grammaire. Toutefois, ce dernier ne correspond pas dans sa totalité au métamodèle SI-PME : il couvre les concepts mais pas toutes les relations qui les lient alors que ces relations permettent de favoriser certaines qualités telles que la cohérence et la complétude. Nous traduisons donc les liens entre attributs définis par chaque *boilerplates* par les relations correspondantes existantes entre les concepts représentés par ces attributs. La Figure 20 montre un exemple de *mapping* entre le DSL textuel et le métamodèle SI-PME, où les deux *boilerplates* *BpSutstructure_Entreprise* et *BpPostes_Structure* représentent respectivement les relations *isStructuredIn* et *workstations* du métamodèle SI-PME. Nous construisons ainsi des modèles SI-PME, suivant ces correspondances, à partir des informations collectées par l'éditeur textuel au fur et à mesure de la construction des modèles textuels.

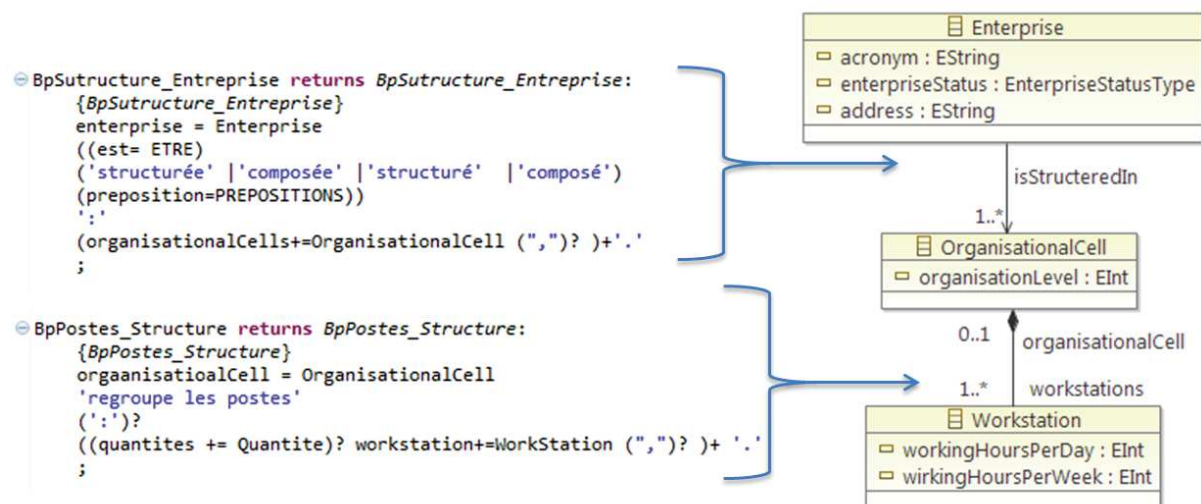


Figure 20. Exemples de correspondance entre le DSL textuel et le métamodèle SI-PME.

A partir du modèle SI-PME ainsi construit, nous utilisons les éditeurs graphiques créés sous Sirius pour lui associer des syntaxes concrètes graphiques : un diagramme de contexte pour la vue du contexte et un organigramme pour la vue organisationnelle. En effet, les syntaxes abstraites du diagramme de contexte (Figure 21) et de l'organigramme sont exactement alignés avec la syntaxe abstraite SI-PME. Ici il s'agit seulement d'interprétation (*rendering*¹¹) par les éditeurs réalisés avec l'environnement Sirius. Nous montrons dans la Figure 22 un exemple de construction de la syntaxe concrète du diagramme de contexte correspondante à la vue du contexte du métamodèle SI-PME. Les éléments graphiques sont mappés avec des

¹¹ Un *mapping* qui affecte des syntaxe concrètes (textuelle, graphique) à chaque élément ou groupe d'éléments de la syntaxe abstraite (Lúcio et al. 2014).

construits du métamodèle SI-PME. Par exemple, l'élément graphique représentant une sortie est mappé au concept du métamodèle SI-PME *ExternalEnterpriseOutput*, il représente un lien allant de l'entreprise au partenaire externe.

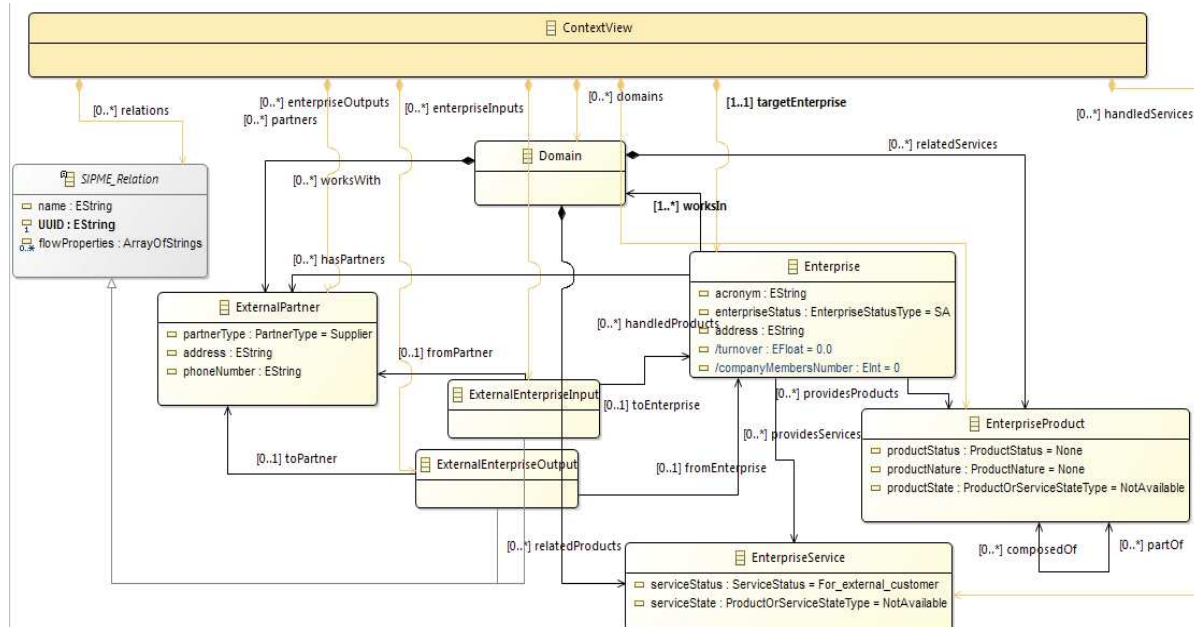


Figure 21. Métamodèle du diagramme de contexte.

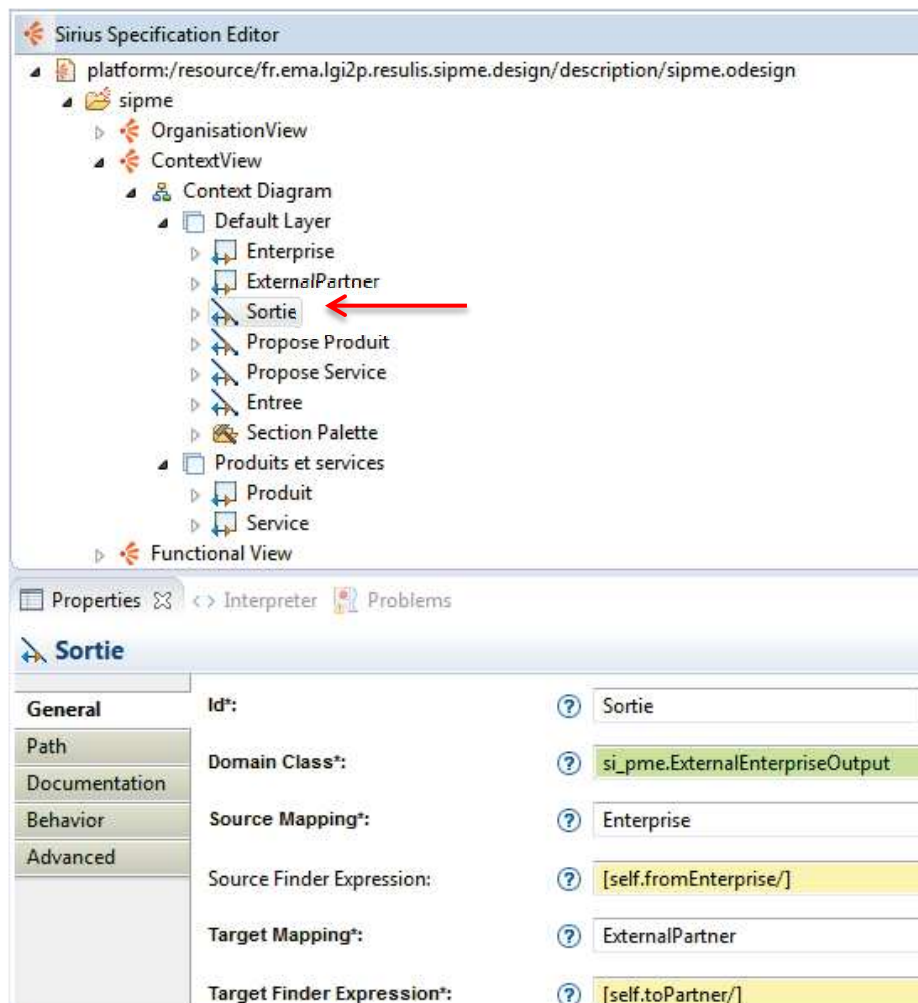


Figure 22. Construction de la syntaxe concrète du diagramme de contexte..

Le diagramme de processus quant à lui présente quelques différences avec le modèle de fonctionnement recueilli. Un ensemble de mises en correspondance est nécessaire entre son métamodèle et la partie du métamodèle SI-PME décrivant des processus. Nous proposons d'utiliser des règles de transformation ATL¹² pour formaliser le mapping nécessaire pour construire des processus transversaux à partir de la description locale du fonctionnement par poste.

Lors de l'expression des besoins, le fonctionnement est décrit au niveau local de chaque poste par l'ensemble des activités qui y sont réalisées. Nous proposons de fusionner l'ensemble des activités des différents postes de façon à faire émerger un ou plusieurs processus. Nous nous basons sur l'utilisation des notifications pour déduire l'ordre des activités les flux d'objets ou de contrôle qui les lient en faisant d'elles des activités d'un même processus qui peut être de ce fait transversal à plusieurs postes.

Rappelons qu'une activité est délimitée par deux notifications : une notification déclenchante et une notification générée. Les notifications **déclenchantes** portent comme informa-

¹² *Atlas Transformation Language* est un langage de transformation de modèles associé à un outil de transformation, disponible sur <https://eclipse.org/atl/>

tion le moment où l'activité doit être effectuée comme par exemple la réception d'un objet d'entreprise à partir d'un autre poste ou à partir d'un partenaire externe. Les notifications **générées** représentent ce qui est réalisé à la fin de l'activité.

Rappelons également qu'au niveau de la grammaire les notifications déclenchantes par exemple sont définies comme suit :

```
(((réception | arrivée | réceptionner | recevoir | Lorsque je reçois) (PREPOSITION)? <EnterpriseObject> ((PREPOSITION) (<OrganisationalCell> | <Workstation> | <CompanyMember> | <ExternalPartner>)) ?)13 | SITUATION14 | ETAT15 | FREQUENCE_TEMPS16)*
```

Nous supposons que l'ensemble des activités appartenant à un même processus sont celles dont l'exécution est conditionnée par une notification déclenchante liée à une autre activité de cet ensemble ou dont la notification générée renvoie vers une autre activité de cet ensemble.

La première activité d'un processus est celle dont aucune des notifications déclenchantes ne correspond à une notification générée par une autre activité. Ces notifications déclenchantes représentent des événements temporels ou proviennent de partenaires externes. Elles forment donc les notifications de début du processus.

La définition des notifications nous permet de définir l'ordre de déroulement des autres activités du processus :

- Les notifications liées à des objets d'entreprise reçus et envoyés précisent la source ou la destination de ces derniers. Nous déduisons donc le lien d'ordre entre les activités et les flux d'objets.
- Pour définir l'ordre de déroulement des activités à partir de notifications représentant un d'état d'un objet d'entreprise ou une situation, nous recherchons dans l'ensemble des notifications celles relatives au même état ou à la même situation.

Nous considérons chaque activité d'un processus comme l'ensemble des actes de gestion réalisés suite à une notification déclenchante dite « externe ». La notification déclenchante externe un nouvel état ou une situation survenue dans un autre poste — différent de celui où l'activité est réalisée, la réception d'un objet d'entreprise de la part d'un autre poste, ou d'un partenaire externe, un moment précis de la journée ou une date précise.

¹³ Exemple de notification par réception : Arrivée de la commande du client X.

¹⁴ Exemple de situation : le voyant de la machine indique la fin du travail.

¹⁵ Exemple d'état : le devis est accepté.

¹⁶ Exemple de fréquence : chaque matin.

Cette granularité des activités de processus peut être différente de celle donnée par les collaborateurs au niveau de l'expression des besoins (en réponse à la question : « Veuillez détailler les étapes de sa réalisation »). Par conséquent, pour former les activités des processus :

- Nous regroupons, en une même activité, les activités données par les collaborateurs relatives à un même poste si aucune notification déclenchante n'est donnée. Nous considérons qu'il s'agit d'une nouvelle activité seulement lorsqu'au moins une de ses notifications déclenchantes est externe.
- Nous scindons en plusieurs activités toute activité dont le détail de déroulement comporte des notifications externes.

La dernière activité d'un processus est celle dont aucune des notifications générées ne correspond à une notification déclenchante d'une autre activité. Ces notifications générées forment donc les notifications de fin du processus.

4.3 Conclusion

Nous avons proposé un DSL textuel utilisant le principe de *boilerplate* pour spécifier les besoins d'informatisation par des collaborateurs non experts. Nous avons également utilisé des DSL graphiques pour assurer la communication et la synthèse des informations recueillies par le DSL textuel. Pour ce faire, nous avons défini les transformations nécessaires pour construire des modèles respectant les DSL graphiques proposés.

Nous profitons de la caractéristique semi-formelle du DSL textuel qui combine la facilité d'utilisation du langage naturel avec la capacité d'automatiser le processus d'expression, d'analyse des besoins et de transformation vers les DSL graphiques. Le chapitre suivant détaille la mise en œuvre de ces langages au sein de la démarche proposée.

Chapitre V : Démarche illustrée et outillages

5.1 Introduction

Le chapitre III présente les premiers éléments de la méthode d'IBE proposée, à savoir, l'ensemble des concepts et relations manipulés par les utilisateurs lors de l'expression des besoins et permettant de gérer la vérification et de validation des exigences. Ces concepts, leurs attributs et les relations qui les lient représentent la syntaxe abstraite des langages, constituant le deuxième élément de cette méthode d'IBE, leurs syntaxes concrètes sont présentées au chapitre IV.

Ces concepts et langages sont mis en œuvre dans ce chapitre pour présenter les deux derniers constituants de cette méthode : la démarche et son outillage. La démarche proposée s'articule en un ensemble d'étapes allant de l'identification des parties prenantes à la validation des exigences. Cette démarche est supportée par un ensemble d'outils qui ont été développés et intégrés dans un prototype démonstrateur de la faisabilité des propositions. Il s'agit d'outils de modélisation, de vérification et de transformation développés sous la forme d'un plug-in Eclipse.

Nous présentons dans la suite le contexte et les attendus de la réalisation de la démarche et les parties prenantes dont les besoins sont à prendre en compte durant le projet d'informatisation. Nous exposons ensuite la démarche et son application.

5.2 Description de la démarche

La démarche que nous proposons s'intègre aux missions d'informatisation réalisées par la SSII RESULIS. L'objectif est de définir un référentiel d'exigences vérifiées i.e. qui présentent un ensemble de qualités telles que la cohérence, la complétude, la non redondance, etc. introduites au chapitre II afin d'être exploitables par la MOE.

Comme précisé au chapitre II, la méthode d'IBE s'applique dans le cas de PME ayant établi une organisation réfléchie et maîtrisant les processus qu'elles souhaitent informatiser. En effet, la démarche d'expression des besoins est descriptive.

La démarche prend donc en entrée les connaissances des collaborateurs de la PME et leur expertise sur le métier de l'entreprise. Ceux-ci décrivent leurs activités au niveau des différents postes pour atteindre les objectifs stratégiques de l'entreprise. L'ensemble forme alors un référentiel des besoins qui couvre également une description du contexte, de la structure, des contraintes, des règles métiers et des ressources de l'entreprise. Le but est ensuite de définir un référentiel d'exigences à partir du référentiel des besoins. Le référentiel des exigences est la sortie attendue de la démarche. Cela nécessite donc une suite d'activités d'analyse et de vérification du modèle des besoins, de validation de ces dernières ainsi et de prise de décision pour la validation du référentiel des exigences.

La démarche proposée (Figure 23) s'articule donc, de façon itérative, en un ensemble d'étapes que sont :

- L'identification des parties prenantes.
- L'expression des besoins.
- La vérification des besoins.
- La validation des restitutions.
- La validation des exigences.

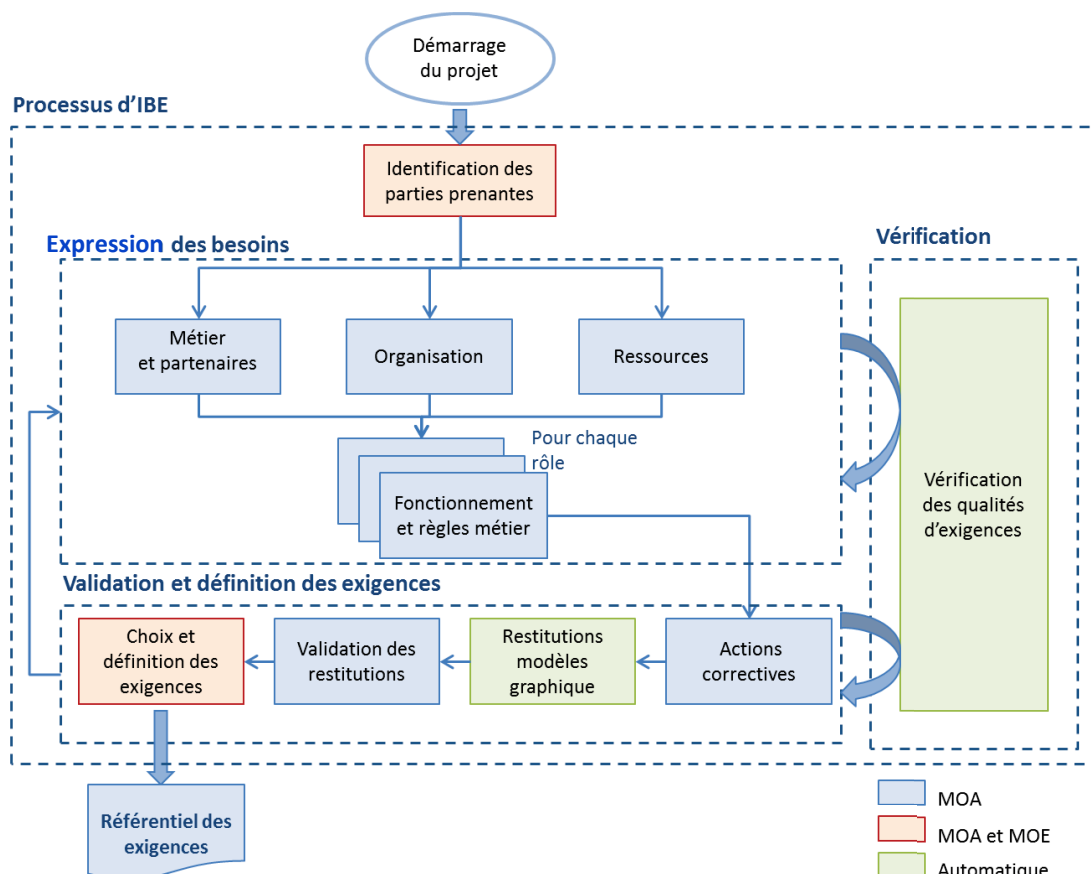


Figure 23. Processus proposé d'ingénierie des besoins et des exigences.

RESULIS entame cette démarche en amont de chaque projet d'informatisation avec les clients souhaitant être accompagnés pour la définition du cahier des charges. RESULIS leur expose la démarche et les accompagne pour la prise de main de l'outillage. Ils choisissent de cette façon les collaborateurs participant à l'expression et à la vérification des besoins et ceux intervenant lors de la validation des restitutions et des exigences, s'ils sont différents. La PME est ainsi fortement impliquée et peut être autonome lors de l'expression de ses besoins.

Nous détaillons dans ce qui suit chacune des étapes de la démarche avec le cas d'étude de la PME SDT et présentons les outils associés.

5.2.1 Identification des parties prenantes

Les parties prenantes du projet d'informatisation sont tout d'abord les collaborateurs de la PME, source principale de connaissances et d'expertise sur le métier de l'entreprise et utilisateurs finaux du résultat de cette informatisation. Ces acteurs représentent ainsi la MOA initiatrice du projet. Les parties prenantes intègrent de même la MOE qui est, dans notre cas, la SSII RESULIS à l'origine de ces travaux, représentée par son responsable et son équipe de développement. Elle est responsable du développement, du test et de la maintenance des applications solutions.

Il est aussi nécessaire de prendre en compte les besoins d'autres parties prenantes en relation avec la MOA et la MOE. Rappelons que pour identifier les parties prenantes, il est important d'identifier toute personne, groupe de personnes ou organisation pouvant influencer ou être affecté par le système ou son élaboration. Les parties prenantes citées ci-dessous sont donc à considérer lors de tout projet d'informatisation :

(i) Celles en relation avec la MOA :

- Les futurs utilisateurs qui interagiront avec le système à réaliser ou utiliseront ses produits (informations, résultats...) qu'ils soient internes ou externes à l'entreprise.
- Les experts métiers du domaine de l'entreprise.
- Les décideurs, responsables de l'acquisition du système, ayant un intérêt financier.
- Les partenaires externes : clients, fournisseurs de l'entreprise et tous ses partenaires ayant des interactions avec le système ou concernés par des informations et des résultats du système.
- Le législateur, les organismes de standardisation et d'audits.

(ii) Celles en relation avec la MOE : la MOE doit prendre en considération les contraintes de différents acteurs :

- Analystes et concepteurs.
- Equipe de développement et chefs de projets;
- Décideurs, responsables de la vente du système ou ayant un intérêt financier;
- Testeurs, formateurs et mainteneurs.
- Sous-traitants.
- Représentants des textes réglementaires et du législateur sur des questions de l'ordre de la solution technique (ex : protections des données personnelles).

Comme expliqué dans le contexte de ces travaux au chapitre I, les PME ne peuvent souvent pas se permettre financièrement de consacrer un budget pour une assistance à maîtrise d'ouvrage pour la représenter, jouer le rôle de coordinateur avec la MOE et l'aider à élaborer ses exigences. Nous avons donc comme contrainte d'assurer l'autonomie des collaborateurs de la PME pour décrire intuitivement et aussi simplement que possible leurs besoins d'informatisation. En offrant des langages accessibles, une démarche guidée et automatisée, la méthode d'IBE que nous proposons peut permettre à la PME de se passer de cette assistance pour l'expression de ces besoins, ce qui peut représenter une économie importante

dans le budget réservé au projet d'informatisation par la PME. Par hypothèse, nous ne considérons donc pas la partie prenante assistance à MOA. La MOE RESULIS accompagne la PME durant les étapes du processus d'ingénierie des besoins et des exigences.

La MOA est donc amenée à mettre à disposition plusieurs **collaborateurs de l'entreprise**, représentatifs de ses besoins de par leurs rôles, leurs responsabilités et leurs attentes en termes d'usage et de mise en œuvre du logiciel de gestion attendu. Ces collaborateurs sont chargés d'exprimer les besoins et participer aux différentes étapes de vérification et de validation. La MOA choisit, avec l'aide de la MOE, un **collaborateur principal** qui sera un expert métier ou tout autre collaborateur :

- pouvant prendre des décisions sur le périmètre du système à réaliser, les domaines d'activité et les postes à prendre en compte,
- ayant une connaissance exacte de la structure interne et des relations externes de la PME,
- pouvant décider ou reporter les décisions des responsables quant aux nouvelles pratiques à apporter en réponses aux choix stratégiques de l'entreprise ou aux contraintes de son environnement.

Pour la PME SDT, le collaborateur principal sera son responsable. La Figure 23 montre l'IHM du prototype dédié au choix des collaborateurs intervenant dans l'étape d'expression des besoins.

Il s'agira ensuite d'identifier un ou plusieurs autres **collaborateurs clés**. Ils auront comme tâche de définir un ensemble de rôles détenus par les collaborateurs de la PME. Ils doivent détailler le fonctionnement de leurs postes et des postes dont ils maîtrisent le fonctionnement. Le nombre de collaborateurs clés dépend de leur maîtrise du fonctionnement des postes intervenants dans les domaines d'activité couverts par le système à réaliser. Nous encourageons la PME à faire intervenir plusieurs collaborateurs pour une couverture maximale des éléments considérés lors de l'expression des besoins. De cette façon, la comparaison des réponses données permet de déceler d'éventuelles divergences et d'affiner l'expression des besoins.

La MOA ainsi représentée doit exprimer de la même manière et implicitement les besoins de ses partenaires externes actuels, leur besoins étant inclus dans les pratiques actuelles de la PME. Dans le cas où la collaboration avec un partenaire externe est étroite et intrusive (par exemple si ce dernier peut intervenir dans le fonctionnement de la PME et agir via le système à développer) son rôle doit être détaillé pareillement aux rôles détenus dans l'entreprise pour montrer ses responsabilités et, en quelque sorte, ses privilèges d'action définis. Les caractéristiques du métier de la PME sont parfois régies par la législation en vi-

gueur, les organismes de standardisation ou d'audit. Les besoins de ces parties prenantes sont donc également implicitement inclus dans la définition des besoins de la PME.

The screenshot shows a software window titled "Choix des collaborateurs". It contains three main input areas: a text field for "Quel est l'effectif global de l'entreprise ?" with the value "30", a list box for "Choisissez le collaborateur principal." containing "François Dupont", and another list box for "Listez les autres collaborateurs à interroger." containing "Thomas, Jules, Claude.".

Figure 24. IHM - Identification des collaborateurs intervenant lors de l'expression des besoins.

Nous considérons l'ensemble des collaborateurs intervenant dans la démarche comme étant qualifiés, crédibles et maîtrisant le métier de la PME, mais au risque d'erreurs ou d'oublis de leur part nous préconisons de choisir un ensemble d'experts pour la revue et la validation des exigences. Il s'agira du collaborateur principal ainsi que d'autres collaborateurs ayant des compétences similaires à celles des collaborateurs clés.

5.2.2 Elicitation des besoins

La MOA est appelée ici, selon son rôle défini ci-dessus, à participer à l'expression des besoins d'informatisation grâce à un questionnaire couvrant les aspects nécessaires à la construction du modèle SI-PME de la PME concernée. Comme proposé au chapitre précédent et dans un souci d'autonomie des collaborateurs, ce questionnaire se présente donc sous la forme d'une suite d'interfaces textuelles pour guider la modélisation en utilisant le DSL textuel. Cela permet de guider et d'assister au fil de l'eau la MOA dans l'expression progressive de ses besoins en filtrant les considérations qui appartiennent à des vues différentes du méta modèle SI-PME. L'interprétation des réponses du questionnaire est possible grâce à l'intégration de l'éditeur Xtext dans les champs des réponses au niveau du prototype de la solution.

L'implication des collaborateurs de la PME dans le processus d'expression des besoins autour de ce questionnaire unifiant assure ou, tout au moins, rassure quant au partage d'une vision commune au sein de l'entreprise et avec les autres parties prenantes du projet d'informatisation.

Les pratiques actuelles de la PME sont ainsi modélisées pour refléter son fonctionnement et son organisation qui sont, rappelons-le, supposés connus, établis et stables. Ces pratiques peuvent inclure de l'existant informatisé. Les collaborateurs ne doivent pas, dans un premier temps, se soucier de la distinction entre ce qui existe (la partie informatisée du système « *as-is* ») et ce qui doit être réalisé (la partie informatisée du système « *to-be* »). Ceci est fait dans l'étape de validation des exigences en collaboration avec la MOE. Le décideur de la PME peut éventuellement, *a posteriori*, définir de nouvelles pratiques selon de nouvelles considérations ou contraintes de l'environnement, du marché ou de la réglementation par exemple. Ce sont des décisions à prendre de manière centralisée. Nous proposons donc que tout changement par rapport à la situation actuelle, décrite par l'ensemble des collaborateurs, soit défini par le collaborateur principal.

Concernant le questionnaire proprement dit, pour chaque question, des indications et explications sont données aux utilisateurs. De plus un exemple de réponse est donné aux collaborateurs. Il constitue un patron syntaxique à suivre puisque la définition des *boilerplates* en utilisant les notations ?, * et + telles que définies au chapitre IV (ex : <OrganisationalCell> regroupe les postes : ((quantité)? <Workstation> (,)?)+) reste non intuitive.

Le questionnaire est décomposé en plusieurs parties, chacune utilisée pour éliciter une vue particulière du modèle de l'entreprise.

- Le questionnaire concerne, dans un premier temps, le collaborateur principal où les questions couvrent : la vue du contexte dans les parties « Quelles sont les lignes directrices de votre entreprise ? » et « Qui sont vos partenaires ? », la vue organisationnelle « Quelle est votre organisation ? » et la vue des ressources « Quelles sont vos ressources ? ».
- Le questionnaire concerne ensuite l'ensemble des collaborateurs jouant le rôle de collaborateurs clés pour décrire le fonctionnement de l'entreprise poste par poste et couvrir ainsi la vue fonctionnelle et ce dans la partie « Comment fonctionnez-vous ? »

Cette décomposition en vues reste néanmoins transparente aux collaborateurs. De plus, les éléments des vues informationnelle et des exigences sont déduits à partir de ce qui est recensé comme informations par les différentes questions. Ils sont restitués lors de l'étape de validation des exigences. Nous exposons dans la suite les questions posées dans chaque partie et illustrons ceci en utilisant des IHM (Figures 25 à 30) tirées du prototype proposé.

5.2.2.1 Domaines d'activité et politique générale de la PME

Nous recensons ici les domaines d'activité de la PME représentant ses spécialités métiers, la politique qu'elle entreprend pour mener à bien son activité et ses principaux objectifs stratégiques (cf. Figure 25). Ceci permettra à la MOE de connaître plus clairement la PME, son niveau de maturité et à la MOA de présenter sa vision et sa culture d'entreprise.

Cette étape permet également de déterminer les frontières du système à informatiser. En effet, le collaborateur principal ne mentionne que les domaines d'activité concernés par l'étude. Le but est aussi de séparer la description du fonctionnement de l'entreprise, dans les étapes suivantes, selon les domaines d'activité. Les collaborateurs devront donc détailler les rôles liés aux domaines d'activité en question.

Quelles sont les lignes directrices de votre entreprise?

Quelle est l'activité de l'entreprise ?

SDTech a pour activité : Micronisation et analyse à façon des poudres fines, Formation continue, Expertise et Conseil.

Quelles politiques entreprenez-vous ?

Dans le cadre d'une vision stratégique à long terme, la filiale apporte aux entreprises industrielles et aux laboratoires de recherche des solutions innovantes parfaitement adaptées à leurs besoins.
De quelques grammes à plusieurs centaines de tonnes, l'entreprise adapte son expertise à toutes les échelles : laboratoire, pilote et production

Quels sont vos objectifs stratégiques ?

Assurer une bonne qualité.
Minimiser le temps de réponse et le temps de réalisation.

Figure 25. IHM – définition des domaines d'activité, politique et objectifs stratégiques.

Les réponses à la question concernant les politiques de l'entreprise est une réponse libre. Elle est optionnelle et ne donne pas lieu à une modélisation par le DSL textuel.

5.2.2.2 Partenaires, produits et services de la PME

Le collaborateur principal définit, pour chaque domaine d'activité, les interactions de la PME avec son environnement en précisant les produits et services qu'elle fournit, ses partenaires externes et les échanges qu'elle établit avec ces derniers (cf. Figure 26).

Ces informations serviront par la suite à vérifier la complétude de la définition du fonctionnement. En effet, le système à réaliser doit, par exemple, réagir à toutes les entrées parvenant de tous les partenaires externes. L'absence de référence à un partenaire révélera inévitablement des omissions.

Qui sont vos partenaires ?

Quels sont les types de partenaires avec qui vous travaillez ?

Les **clients** sont : **des** industriels, **des** établissements **de** recherche.
les **prestataires** sont : les transporteurs.
Les **partenaires** sont : les **partenaires de** recherche.

Qu'est-ce que les partenaires vous fournissent ?

les transporteurs effectuent le transport **des** palettes **et des** colis **aux clients**.
les industriels fournissent **des** commandes **de** services, les produits à traiter.

Que mettez-vous à disposition de vos partenaires ?

SDTech **fournit des** rapports **d'analyses aux clients** établissements **de** recherche
SDTech propose **des** formations **aux** industriels

En plus de ces échanges, veuillez préciser vos produits et services.

Nos services sont : la micronisation, l'analyse **de** poudres, le broyage, le tamisage, la classification **des** poudres, le mélange

Figure 26. IHM – définition des partenaires, échanges, produits et services.

5.2.2.3 Organisation interne de la PME

Indépendamment du fonctionnement de l'entreprise, le collaborateur principal détaille la structure interne de cette dernière et liste les cellules organisationnelles et les postes qui les constituent (cf. Figure 27). C'est ici un moyen de considérer l'entreprise comme un ensemble structuré de postes qui regroupent les responsabilités que doivent assurer les collaborateurs détenant ces postes. La stratégie d'expression que nous entreprenons se base sur cet ensemble de postes pour détailler les rôles qui y sont joués lors de la définition du fonctionnement.

La décomposition de l'entreprise en cellules organisationnelles est optionnelle. Une PME de moins de 10 collaborateurs n'adopte pas forcément une organisation par structures distinctes. Les postes quant à eux doivent être définis. Si l'intitulé d'un poste n'est pas précisé il peut être remplacé par le nom d'un des collaborateurs qui le détient. Pour simplifier nous ne considérons pas les sous-cellules transversales à plusieurs cellules organisationnelles.

Quelle est votre organisation ?

Quelles sont les structures et sous-structures qui constituent l'entreprise ? 

SDTech est **structurée en** : direction **et** administration, production, marketing **et** commercial, recherche **et** développe
Production est **structurée en** : atelier **et** maintenance.

Quels sont les postes qui constituent chaque structure ? 

Production **regroupe les postes** : **responsable de** production.
Atelier **regroupe les postes** : chef **d'**atelier, 2 responsables **d'**équipe, 5 opérateurs.

Figure 27. IHM – définition de l'organisation interne.

5.2.2.4 Ressources de la PME

Suivant la stratégie d'expression qui consiste à récolter des informations concernant les mêmes éléments de l'entreprise à plusieurs niveaux, nous faisons lister au collaborateur principal l'ensemble des ressources humaines, applicatives ou matérielles que possède l'entreprise, et ce relativement à chaque poste, avant de considérer leur utilisation lors de la définition du fonctionnement (cf. Figure 28).

Quelles sont vos ressources ?

Qui sont les collaborateurs qui détiennent chaque poste ? 

Directeur général est **occupé par** Blake Lively
responsable de production est **occupé par** Ross Geller

Quelles sont les ressources applicatives ou matérielles (documents, équipements, machines, robots, etc.) nécessaires à la réalisation des activités au niveau de chaque poste ? 

Opérateur nécessite **les** ressources : broyeur, tamiseur, mélangeur, équipements **de** laboratoire, moyens **de** manutention.

Figure 28. IHM – définition de ressources.

5.2.2.5 Fonctionnement de la PME

Le questionnaire est ici destiné aux collaborateurs clés, sachant que le collaborateur principal peut faire partie de ce groupe de collaborateurs. Ils doivent définir les rôles détenus au niveau de chaque poste (Figure 29) en ne décrivant que les activités réalisées au sein du poste en question. Chaque activité (Figure 30) est définie en précisant : les notifications déclenchantes et générées, les entrées et sorties, le détail de l'activité, les règles de gestion à respecter, les ressources utilisées et les interactions avec d'autres collaborateurs internes ou partenaires externes. Il est aussi possible de définir des rôles pour les partenaires externes si l'on veut donner à ses derniers la possibilité d'agir sur le système.

Le fonctionnement est donc défini par poste. Ce qui signifie qu'une description locale des activités réalisées dans chaque poste est donnée sans préciser le déroulement des activités d'un poste à un autre. Nous ne demandons donc pas aux collaborateurs de définir les processus de bout en bout. Ces derniers sont souvent transversaux à plusieurs postes. Nous choisissons cette stratégie d'expression car, souvent, les collaborateurs n'ont pas une vision globale exacte de ce qui est réalisé au sein de l'entreprise. De plus, malgré la taille de la PME, les structures organisationnelles restent souvent cloisonnées.

Les collaborateurs réalisent des missions différentes, même en exerçant dans un seul poste. En effet, les collaborateurs des PME sont assez polyvalents et peuvent détenir des rôles dont les activités sont réalisées dans le cadre de processus ayant des objectifs différents. Par défaut, si les rôles ne sont pas précisés, nous supposons que chaque collaborateur détient au moins un rôle.

Dans l'exemple de la PME SDT le domaine choisi est celui de la micronisation et de l'analyse à façon des poudres fines. Les Figures 29 et 30 illustrent des réponses du responsable de production sur la partie concernant ses rôles et une des activités de son poste.

Comment fonctionnez-vous ?

Selectionnez un poste.

- Responsable de production
- Chef d'atelier
- Responsables d'équipe
- Opérateur

Quels sont les rôles attribués à ce poste?

La gestion de la production
le suivi de la qualité
le respect des consignes de sécurité

Listez les principales activités liées à chaque rôle.

La Gestion de la production : planification des commandes, gestion de la ressource humaine, prise en compte des demandes, gestion des problèmes, création des rapports.

Figure 29. IHM – définition des rôles et des activités.

Comment fonctionnez-vous ?

Selectionnez un poste.

Responsable de production
Chef d'atelier
Responsables d'équipe
Opérateur

Selectionnez une activité.

planification des commandes
gestion de la ressource humaine
prise en compte des demandes
gestion des problèmes
création des rapports

Quand est-ce que vous réalisez cette activité ? 

Arrivée de la commande client.

Quels sont les informations / dossiers / produits utilisés, consultés ou transformés durant cette activité ? 

Détails de la commande {produit, quantité, date de livraison souhaitable, CDC technique}

Veuillez détailler les étapes de la réalisation de cette activité. 

Choisir les machines,
choisir les opérateurs,
planifier,
définir les étapes du traitement.

Quels informations / dossiers / produits / services sont produits durant de cette activité ? 

Dossier de production.

Qu'est-ce que vous faites de ce résultat ? 

le Dossier de production est envoyé à l'atelier.

Figure 30. IHM – définition du détail d'une activité.

5.2.3 Vérification des besoins

L'étape d'expression des besoins permet de renseigner le modèle SI-PME et ainsi de construire une partie du modèle de la PME, tel qu'il est souhaité par ses collaborateurs à partir des réponses au questionnaire. Ce modèle est partagé par la suite avec les parties prenantes de la MOE, étant à la base du référentiel d'exigences attendu. Rappelons que (INCOSE, 2012, IEEE 1993) préconisent que chaque exigence doit être : nécessaire, indépendante de l'implémentation, complète, non ambiguë, singulière, identifiable ou traçable, faisable, vérifiable, correcte, conforme, concise et communicable. Le référentiel des exigences doit ensuite être structuré, cohérent, faisable, limité, non-redondant, priorisé et complet.

Les besoins élicités peuvent présenter des défauts qui détériorent la qualité des exigences qui en résultent. Nous proposons donc, et ce en parallèle des étapes d'expression, des activités de vérification où les besoins sont analysés et comparés pour détecter d'éventuels défauts, notifier leur existence et permettre aux collaborateurs de la PME d'apporter des corrections.

Nous assurons certaines des qualités citées précédemment au travers de techniques de vérification semi-formelles « *light-weighted* » et ce, à différents niveaux des choix de conception des composantes qui constituent la méthode proposée, à savoir : les concepts, les langages, le processus systématisé et l'outillage. Nous explicitons dans ce qui suit comment nos propositions contribuent à garantir ces qualités.

Nécessaire, utile et justifiable :

L'utilisation du questionnaire pour l'expression des besoins couplé avec un langage contrôlé nous évite de collecter des informations inutiles. En effet, dans les interviews libres, les collaborateurs ont tendance à donner des informations parasites qui dépassent la limite de la question ou, en général, celle du domaine étudié. De par la nature des questions, nous ne recueillons ainsi que les informations nécessaires à la construction du modèle de l'entreprise et à la gestion des exigences, en respectant le métamodèle SI-PME.

Les *boilerplates* précisant le format des réponses constituent un moyen utile pour cadrer l'expression des besoins. Il n'est certes pas suffisant puisque les collaborateurs peuvent, tout en respectant syntaxiquement et grammaticalement les *boilerplates*, donner des réponses redondantes, sans lien avec l'ensemble des réponses ou incomplètes. Nous expliquons plus loin comment nous mettons en œuvre d'autres moyens de vérification pour remédier à ces défauts.

Indépendante de l'implémentation :

Les besoins élicités et les exigences qui en découlent sont par définition indépendants de l'implémentation, puisque les concepts de la syntaxe abstraite des langages dans lesquels ils sont exprimés ne font aucune référence à des concepts liés à l'implémentation.

Complète :

Nous considérons ici la complétude de l'expression de chaque besoin. Les réponses au questionnaire étant contraintes par des *boilerplates*, leur complétude est automatiquement vérifiée par le parseur de l'éditeur Xtext. Des messages d'erreur sont utilisées pour prévenir les collaborateurs s'ils ne respectent pas la formulation de la réponse telle que définie par le *boilerplate*.

Non ambiguë :

Le métamodèle définissant la syntaxe abstraite du langage d'expression des besoins organise les concepts de façon à éviter les interprétations multiples. Aussi, l'alignement existant entre les attributs des *boilerplates* et les concepts de la syntaxe abstraite limite les interprétations possibles de chaque besoin. Ceci permet de diminuer l'ambiguïté des besoins collectés. Il est évident cependant que la revue des experts reste nécessaire pour lever totalement l'ambiguïté.

Singulière, concise et simple :

Les *boilerplates* du DSL textuel permettent de définir des besoins et des exigences singulières et concises ne portant que sur une seule idée. Elles sont simples et ne comportent pas d'expressions complexes ou composées.

Communicable :

Les besoins et les exigences sont exprimés en langage naturel ce qui les rend communicables et compréhensibles par toutes les parties prenantes. Elles sont aussi structurées de façon à simplifier leur lecture.

Identifiable et traçable :

Nous contribuons à l'identification et à la traçabilité des besoins et des exigences avec l'adoption d'une démarche de méta-modélisation outillée pour définir les besoins en instanciant un métamodèle qui englobe des liens de traçabilité entre : les besoins et les parties prenantes qui les expriment, les besoins et les exigences, les exigences fonctionnelles et les exigences non fonctionnelles et entre les exigences et des éléments de modélisation appartenant aux autres vues du modèle.

Correcte :

Nous guidons les collaborateurs dans le processus de formulation de besoins syntaxiquement corrects grâce à l'utilisation du langage contrôlé et de techniques de NLP (outil tree-Tagger) pour vérifier les informations grammaticales des attributs des *boilerplates*. Par exemple, la définition d'un acte de gestion doit commencer par un verbe. La Figure 31 décrit une activité « réalisation des tâches » du poste de travail « opérateur » pour la société SDT.

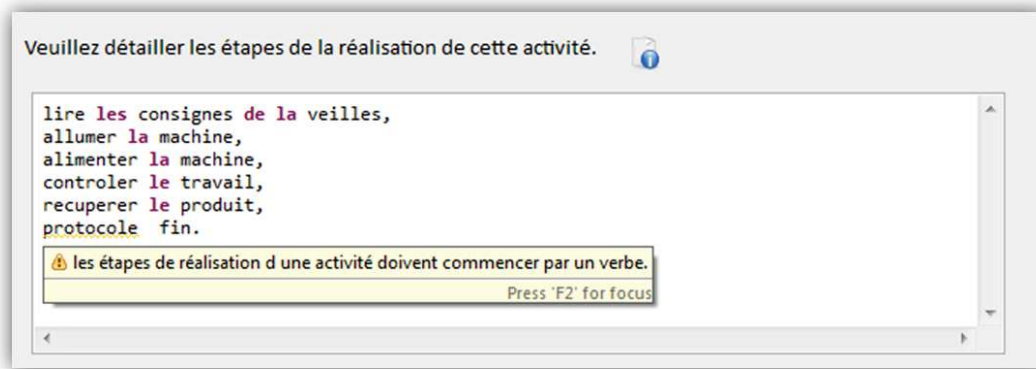


Figure 31. IHM – détection d’une erreur syntaxique.

De plus, en répondant aux questions, les collaborateurs sont supposés renseigner les attributs des *boilerplates* par des éléments relatifs à leur entreprise et à son fonctionnement. Nous ne négligeons pas pour autant les cas où les réponses n’ont pas de sens pour le métier de l’entreprise. La revue par les collaborateurs experts des modèles graphiques du modèle IS-PME (Figure 32 à 35) reste nécessaire pour indiquer si les informations données sont celles que le *boilerplate* est censé collecter et favoriser ainsi la correction et la pertinence des exigences.

Conforme au standard ou au langage appliqué.

L’adoption d’une démarche de méta-modélisation outillée permet de vérifier la conformité des modèles construits à leur métamodèle. Le parseur de l’éditeur Xtext utilisé pour formuler les besoins construit des modèles (instances) conformes à la grammaire et aux *boilerplates* du DSL textuel. Le métamodèle Ecore sous-jacent de ce DSL est défini sur la base du graphe de syntaxe abstrait lié à la définition de la grammaire. Cependant, ce dernier ne correspond pas dans sa totalité au métamodèle SI-PME qui englobe dans sa définition certaines qualités telles que la cohérence et la complétude des modèles construits. Pour pallier ce problème, nous avons établi des correspondances entre le métamodèle du DSL textuel et le métamodèle SI-PME. Nous créons une instance du modèle SI-PME, suivant ces correspondances, au fur et à mesure de la construction des modèles textuels. Nous appelons par la suite une fonction Ecore qui permet de vérifier la conformité de cette instance à son métamodèle.

La qualité de chaque spécification de besoins et d’exigence ne garantit pas la qualité de l’ensemble des exigences et l’absence d’omissions et d’incohérences. L’ensemble d’exigences doit ensuite être :

Complet :

La complétude est un vœu pieux. Elle est par nature difficile à évaluer, il est donc très difficile de s’assurer que les besoins élicités couvrent tout ce qu’il doit être couvert dans chaque domaine d’activité de l’entreprise, autrement dit, qu’ils reflètent la totalité des besoins réels

des parties prenantes et prennent en compte toutes les hypothèses et contraintes recensées sur l'environnement de l'entreprise. Nous ne pouvons pas contrôler l'oubli objectif ou subjectif des collaborateurs.

Cependant, la conformité du modèle des besoins au métamodèle IS-PME permet de s'assurer du respect de plusieurs règles de quasi complétude en utilisant les contraintes de cardinalités. Par exemple : toute partie prenante doit exprimer au moins un besoin ou encore l'entreprise exerce son métier dans au moins un domaine d'activité.

De même, grâce à la stratégie d'expression mise en œuvre, nous minimisons les risques d'omissions en collectant les mêmes informations auprès de plusieurs collaborateurs. Par exemple, le collaborateur principal liste dans un premiers temps l'ensemble des ressources utilisées dans chaque poste. Nous vérifions *a posteriori* que toutes ses ressources sont bien mentionnées dans le détail du fonctionnement des activités réalisées au niveau de ces postes. De la même façon, la complétude peut être vérifiée en considérant toutes les entrées et les contraintes identifiées dans l'environnement du système pour considérer tous les comportements possibles au sein du système d'information. De cette façon, nous listons les partenaires externes et les échanges de l'entreprise avec son environnement pour vérifier *a posteriori* si ceux-ci apparaissent au niveau du détail du fonctionnement.

Cela nous a conduits à définir l'ensemble de règles présentées au chapitre III rassurant la MOE et la MOA sur la complétude du référentiel des exigences. Ces règles sont traduites par des contraintes OCL et des scripts java liés au modèle Ecore définissant la syntaxe abstraite. Elles concernent les informations données par chaque collaborateur et celles définies en commun par plusieurs collaborateurs et ce, relativement à chaque vue. Par exemple, la règle de complétude R1_{complétude_fonctionnelle/contexte}, où l'on précise qu'il existe au moins une notification en relation avec chaque partenaire externe de type client ou fournisseur, est vérifiée par la contrainte OCL suivante définie dans le contexte du concept *ExternalPartner*.

```
invariant
ExternalPartnerIncludedInFunction :
(self.partnerType = 'Customer' or self.partnerType = 'Supplier') implies
(self.notificationFromPartner -> notEmpty() or self.notificationToPartner -> notEmpty());
```

Elle permet de croiser les informations données par le collaborateur principal concernant les partenaires externes avec celles données par les autres collaborateurs. Lorsqu'elle n'est pas vérifiée, il peut s'agir d'une réelle omission au niveau de la description du fonctionnement, ou d'un ajout d'une information inutile ou d'une formulation erronée.

Lorsque les règles de complétude ne sont pas respectées, nous signalons aux utilisateurs le défaut de complétude associé. Nous avons remarqué, lors de l'expérimentation réalisée avec la société SDT, que les collaborateurs ne détaillent pas assez les actes de gestion des activités. La détection des omissions liées à l'utilisation des entrées, sorties et ressources au niveau de la définition des activités, encourage les collaborateurs à fournir plus de détails dans

le déroulement de ces dernières et à contribuer de ce fait à la complétude des exigences fonctionnelles.

Par exemple dans le cas de SDT, au niveau du détail de l'activité « planification des commandes », la règle de complétude suivante n'est pas vérifiée :

R1_{complétude_fonctionnelle} : si l'activité est détaillée en actes de gestion alors il existe au moins un acte de gestion dans le détail d'une activité agissant sur les objets d'entreprise entrées ou sorties de l'activité.

En effet, la sortie produite « Dossier de production » n'est pas citée au niveau du détail de l'activité. Autrement dit aucun acte de gestion n'agit sur cet objet d'entreprise. Nous signalons ceci aux utilisateurs après vérification de l'activité. La règle R1_{complétude_fonctionnelle} est vérifiée par une contrainte OCL dans le contexte du concept *Activity* définie comme suit :

```
invariant
ActivityInputsUsed:
(self.acts.relatedObject) ->
includesAll( self.consumesObjects);

invariant ActivityOutputsUsed:
(self.acts.relatedObject) ->
includesAll( self.providesObjects);
```

Contrairement aux ressources utilisées dans une activité, les entrées doivent être modifiées. Si aucun acte de gestion n'agit sur une entrée, comme « Détails de la commande » dans l'activité « planification des commandes », nous demandons alors au collaborateur si elle est une entrée consultée dans ce cas-là elle sera considérée comme étant une ressource, sinon le collaborateur doit donner plus de détails quant à sa manipulation.

Ces règles de complétude ne permettent tout de même pas de détecter les informations manquantes qui n'ont été mentionnées à aucun niveau et dont l'omission ne cause pas d'incohérences. Nous tentons d'éviter cela avec l'examen des modèles restitués par les experts. Ces derniers peuvent en effet constater les oublis.

Cohérent :

L'adoption d'un métamodèle unique qui structure les concepts manipulés lors de l'expression des besoins nous permet de définir et de vérifier des règles de cohérence, elles aussi données au chapitre III, intra et inter-vues tout en croisant les informations données par les différents collaborateurs. Ces règles permettent de ce fait de renforcer la cohérence globale du référentiel des exigences.

Certaines règles favorisent la cohérence inter-vues. Par exemple, la règle R4_{cohérence_fonctionnelle/ressources} permet de vérifier que toutes les ressources citées par le collaborateur principal pour chaque poste, figurant dans la vue des ressources, sont bien utilisées au niveau de ces postes lors de la définition du fonctionnement. Elle est formalisée par la contrainte OCL *resourcesUsedInWorkstaion* exprimée ci-dessous.

R4_{cohérence_fonctionnelle/ressources} : toute ressource est utilisée au moins une fois au niveau des activités.

```
invariant
resourceIsUsedInWorkstaion:
self.usedIn -> forAll(w : Workstation | w.workstationActivities ->
exists(a : Activity | a.requiresResources ->includes (self))
);
```

D'autres règles sont liées à la cohérence du traitement des données manipulées. Il est important, par exemple, de savoir au niveau de quel poste et durant quelle activité est créé chaque objet informationnel manipulé. Pour ce faire la règle R1_{cohérence_fonctionnelle/informationnelle} permet de vérifier que tout objet d'entreprise d'origine interne est créé au moins une fois. Elle est formalisée par la contrainte OCL *internalEnterpriseObjectIsCreated* ci-dessous.

R1_{cohérence_fonctionnelle/informationnelle} : tout objet d'entreprise d'origine interne est créé au moins une fois au niveau d'un acte de gestion. Il doit figurer en sortie de l'activité à laquelle appartient cet acte de gestion.

```
invariant
internalEnterpriseObjectIsCreated:
(self.objectOrigin = 'Internal_provider')
implies (self.usedInBusinessAction
-> exists ( b : BusinessAction | b.actOfActivity.providesObjects -> includes(self) ));
```

Enfin, certaines règles de cohérence sont liées à la nature des concepts. La règle R1_{cohérence_organisationnelle} vérifie qu'une structure organisationnelle n'est pas affectée à elle-même. La contrainte *subOrganisationalCells* permet de s'en assurer.

R1_{cohérence_organisationnelle} : une cellule organisationnelle peut être affectée à d'autres cellules organisationnelles différentes de celle-ci.

```
invariant subOrganisationalCells:
self.assignedTo -> excludes (self);
```

De plus, la stratégie d'expression par poste favorise la vérification de la cohérence, car nous pouvons examiner si les informations données par chaque collaborateur coïncident en termes d'entrée attendues, de sorties et de séquence des activités. Par exemple, si dans le détail d'une activité, la notification déclenchée de cette dernière précise qu'un objet d'entreprise est envoyé à un autre poste, alors une des activités réalisées au niveau de ce poste doit mentionner l'utilisation de cet objet d'entreprise. La contrainte OCL suivante permet de s'en assurer, elle est définie dans le contexte du concept *Notification*.

```
invariant
objectsRelatedToNotificationsInterPostes :
self.toWorkstation -> forAll (w : Workstation | self.toWorkstation = w and self.associatedToEnterpriseObjects
-> forAll ( o : EnterpriseObject | w.notificationFromWorkstation -> forAll (n : Notification |
(n.associatedToEnterpriseObjects -> union (n.initiates.consultsObjects)
-> union (n.initiates.consumesObjects))
-> includes (o) ));
```

La cohérence des exigences sous-entend que les termes utilisés sont compris de manière unique par toutes les parties prenantes, et que les même termes sont utilisés pour qualifier des *choses* équivalentes. Autrement dit nous ne souhaitons pas avoir, dans l'ensemble des termes qualifiant des concepts de même nature (par exemple l'ensemble des domaines, des partenaires externes, des ressources, etc.) :

- Plusieurs termes utilisés pour qualifier un même élément du SI de l'entreprise. Il s'agit donc de reformulation. La vérification de la complétude peut minimiser l'incohérence causée par la reformulation. On peut par exemple noter qu'une ressource citée par le collaborateur principal n'est pas utilisée dans la description des postes. La détection de cette omission peut révéler que la ressource a bien été citée au niveau du fonctionnement en utilisant un autre terme (comme expliqué pour la règle R1_{complétude_fonctionnelle/contexte}).
- Un même terme pour qualifier des éléments différents du SI de l'entreprise. Nous détectons ceci comme étant une redondance. Si les collaborateurs confirment qu'il ne s'agit pas de redondance, nous leur proposons d'utiliser des termes différents.

Enfin, la revue des experts est nécessaire pour attester l'existence d'incohérences liées à l'utilisation des termes.

Lorsque la même activité est décrite par différents collaborateurs, nous procédons à des comparaisons. Nous commençons par vérifier auprès des collaborateurs qu'il s'agit bien de la même activité. Nous tolérons ce cas de redondance pour permettre l'unification des pratiques au sein de la PME. Si c'est le cas, nous comparons les différents éléments décrivant l'activité. Lorsque des différences sont détectées nous demandons aux collaborateurs concernés (secondés s'il le faut par le collaborateur principal) d'arbitrer sur ce qui doit être gardé pour arriver à un consensus.

Faisable :

Nous préconisons une étape de validation des exigences où la MOE collabore avec la MOA pour juger de la faisabilité des exigences, afin de s'assurer qu'une solution satisfaisant les exigences peut être réalisée en précisant les délais, budget et choix techniques du projet de réalisation. Aussi, nous favorisons la revue par les collaborateurs experts des informations recueillies en les restituant de manière synthétique via des modèles graphiques (décrits dans la section suivante).

Limité :

La stratégie d'expression que nous adoptons commence par définir les domaines d'activité de l'entreprise avant de les détailler. Par conséquent, le collaborateur principal ne mentionne que les domaines d'activité concernés par l'étude. Cette étape permet de cadrer les frontières du système à informatiser. Les exigences n'appartiennent de ce fait qu'au contexte du système à réaliser.

Structuré et homogène :

La syntaxe abstraite des langages utilisés définit les concepts et relations qui permettent de construire des modèles de besoins et d'exigence fondés et en adéquation avec la nature du système à réaliser qui est ici constitué d'un ensemble d'applications supports au système d'information de l'entreprise. Elle permet également de définir des sous-ensembles d'informations en les structurant en vues complémentaires.

Non-redondant :

Pour assurer que le référentiel des exigences ne contient pas la même exigence plus d'une fois ou reformulée d'une manière différente, nous analysons le modèle des besoins. Nous nous assurons par exemple que l'ensemble des domaines, des cellules organisationnelles, des postes, des activités, etc. ne contiennent pas la même information deux fois en utilisant des techniques de comparaisons de racine des mots issues du traitement de langage naturel (NLP). La reformulation est plus difficile à détecter : la détection d'omissions peut contribuer à lever les reformulations. Aussi, la revue d'experts est nécessaire pour attester l'existence d'informations faisant référence à un même élément du SI.

Priorisé :

Pendant la phase de validation des exigences, la MOA en collaboration avec la MOE classe les exigences et définit des priorités de réalisation en fonction de leur importance.

Nous résumons dans les Tableaux 11 et 12 les choix conceptuels et les techniques de vérification et leurs contributions pour assurer les qualités attendues d'une exigence et d'un référentiel d'exigences. Ces techniques impliquent le moins possible les collaborateurs de la PME et restent relativement transparentes pour eux.

Tableau 11. Contributions des choix conceptuels de la méthode aux qualités attendues des exigences.

	Choix des concepts	Méta-modélisation et outillage	CNL + NLP	Stratégie d'expression	Analyse et comparaison	Règle de complétude et de cohérences	Revue d'experts
Nécessaire			x	x			
Indépendante de l'implémentation	x						
Complète			x				
Non ambiguë		x	x				x
Singulière			x				
Communicable	x		x				
Identifiable et traçable	x	x				x	
Correcte			x				x
Conforme au standard ou au langage appliqué		x					

Tableau 12. Contributions des choix conceptuels de la méthode aux qualités attendues du référentiel des exigences.

	Choix des concepts	Méta-modélisation et outillage	CNL + NLP	Stratégie d'expression	Analyse et comparaison	Règle de complétude et de cohérences	Revue d'experts
Complet		x		x	x	x	x
Cohérent		x		x	x	x	x
Faisable							x
Limité			x	x			x
Structuré et homogène	x	x	x				
Non-redondant					x	x	x
Priorisé							x

5.2.4 Validation des restitutions

Nous fournissons aux collaborateurs émetteurs des besoins et aux parties prenantes du projet d'informatisation, une représentation plus synthétique des informations fournies lors de l'étape d'expression des besoins. Il s'agit de restitutions de modèles graphiques construits par le biais de transformations entre le DSL textuel et les DSL graphiques présentés au chapitre IV.

Ces transformations sont réalisées après l'étape de vérification des modèles de besoins élicités. Les modèles restitués sont un moyen de demander aux collaborateurs : « est-ce bien ce que vous vouliez dire ? », ils sont un support de validation où les experts de la MOA effectuent une revue. Nous avons vu dans le paragraphe précédent que la revue était nécessaire pour consolider l'examen de certaines qualités. A partir de leurs constatations, les collaborateurs experts peuvent apporter des modifications aux besoins collectés jusqu'à la validation totale des besoins restitués.

5.2.4.1 Restitution du contexte de l'entreprise

Avec les diagrammes de contexte, nous représentons la PME SDT selon un point de vue externe. La Figure 32 retrace les échanges qu'elle effectue avec ses partenaires externes et la Figure 33 résume les produits et les services qu'elle propose.

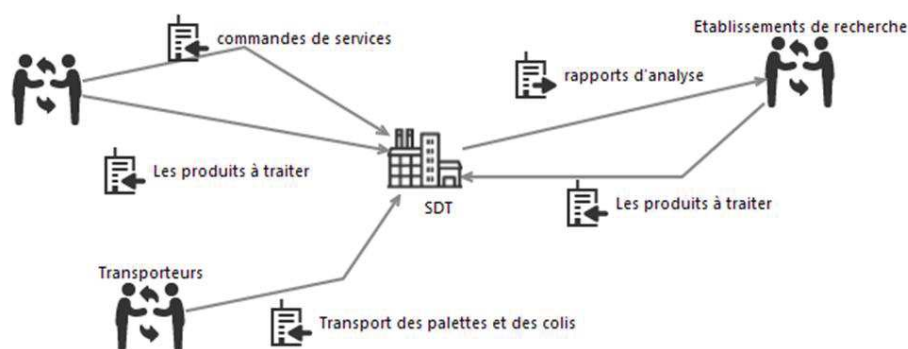


Figure 32. Diagramme de contexte – partenaires de SDT.

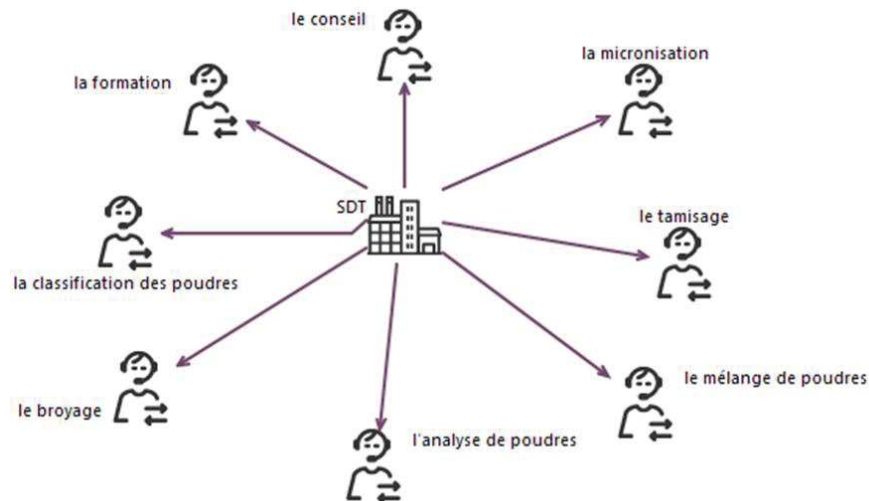


Figure 33. Diagramme de contexte – services de SDT.

5.2.4.2 Restitution de la structure de l'entreprise

L'organigramme (Figure 34) retrace la structure interne de l'entreprise. SDT est en effet structuré en quatre structures, on y retrouve les postes de travail cités pour chaque structure et les relations hiérarchiques qui les lient.

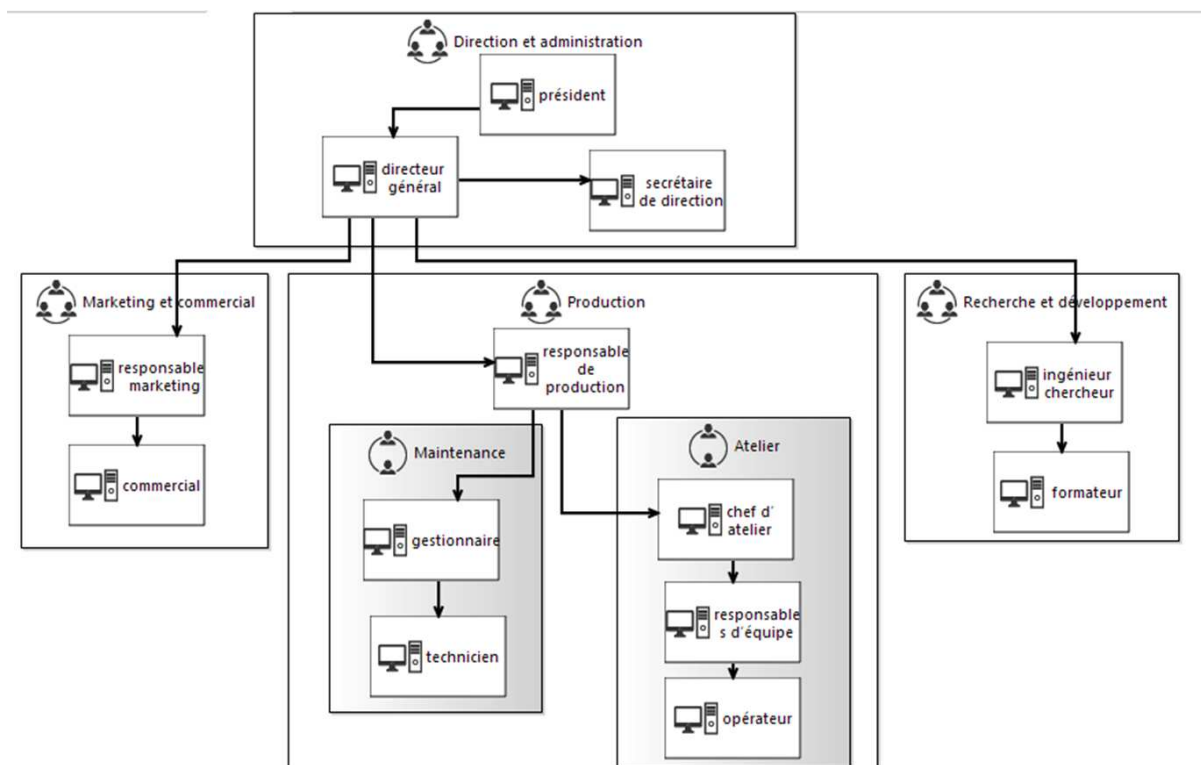


Figure 34. Organigramme partiel de SDT.

5.2.4.3 Restitution du fonctionnement de l'entreprise

En répondant au questionnaire, les collaborateurs de la PME décrivent le fonctionnement au niveau local de chaque poste. Avec les diagrammes de processus, une vision globale du fonc-

tionnement de l'entreprise est donnée. Nous restituons ici (Figure 35) le processus de la prise en compte de la commande client.

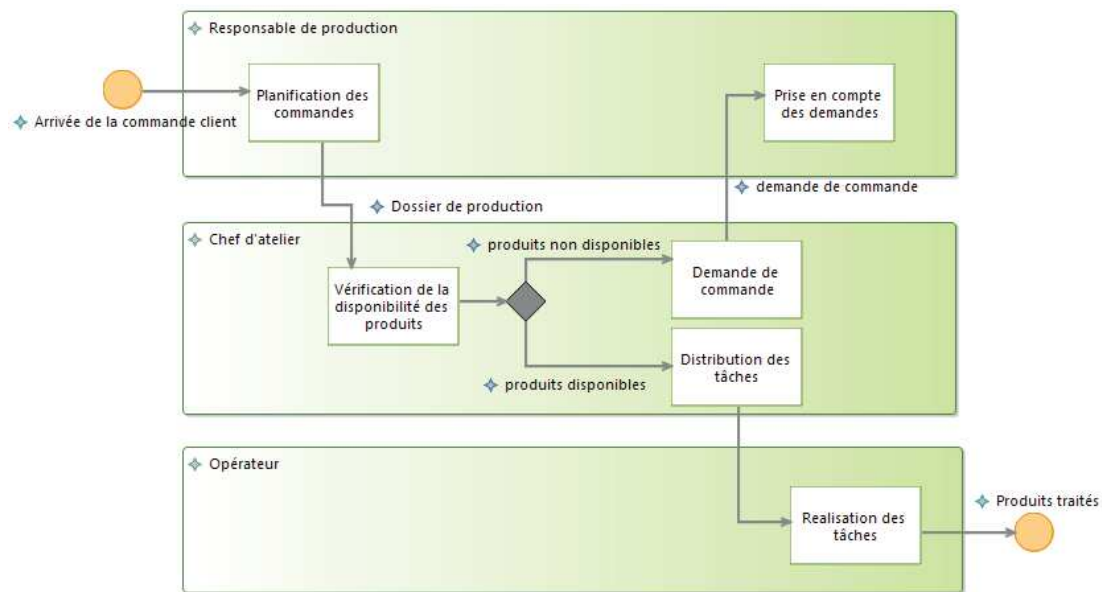


Figure 35. Diagramme de processus « Prise en compte de la commande client ».

5.2.5 Validation des exigences

Après vérification et validation des besoins, nous proposons aux parties prenantes un référentiel des besoins à partir duquel des décisions doivent être prises pour choisir les exigences à réaliser et les valider. Pour ce faire, la MOA est assistée par la MOE pour définir, dans cet ensemble de besoins :

- Ce qui est manuel et ce qui peut être informatisé, car en effet certaines activités décrites par les collaborateurs doivent être exécutées d'une façon totalement ou partiellement manuelle donc non informatisable. Elles doivent tout de même être élicitées et prises en compte pour comprendre le déroulement des processus de la PME. Elles peuvent aussi impacter d'autres activités.

- Préciser, dans ce qui est informatisable, ce qui fait partie de l'existant informatisé. Dans ce cas il faut prendre une décision : garder cet existant ou le reprendre ? Il faut également préciser ce qui n'est pas supporté par une informatisation et décider donc de ce qui va être réalisé par la MOE. Ainsi les parties prenantes du projet définissent ce qui sera effectivement développé (système « *to implement* ») en tenant compte des contraintes du projet.

Ces choix conduisent à l'élaboration du référentiel final des exigences. Il est le résultat des consensus et décisions prises par les deux parties prenantes. Ce référentiel définit les principales caractéristiques des applications supports au système d'information de la PME, à savoir les fonctionnalités qu'elles doivent proposer, représentées par des exigences fonctionnelles, et les informations qu'elles permettent de gérer et de stocker représentées par un ensemble d'objets d'entreprise.

Les besoins collectés concernant les postes, les rôles, les activités et leurs détails décrivent le

fonctionnent de la PME. Ils fournissent donc une indication sur les activités qui doivent être supportées par les applications de gestion à réaliser. Nous définissons donc les exigences fonctionnelles sur la base des activités réalisées dans les différents postes. Les exigences fonctionnelles candidates, à valider, recensent donc toutes les instances du concept *Activity*. Nous regroupons l'ensemble des propriétés et des informations liées à chaque exigence fonctionnelle dans un *Template* (Tableau 13).

Tableau 13. Template d'une exigence fonctionnelle.

Exigence fonctionnelle <requirement UUID> <requirementStatement>	SysReq0011 planification des commandes		
Version <expressionDate>	01/11/2015		
Auteurs <Stakeholder>*	Jules		
Poste de travail <Workstation>*	responsable de production		
Rôle <Role>	La Gestion de la production		
Séquence :	Précondition <Notification>	Arrivée de la commande client	
	Séquence <BusinessAction>*	Choisir les machines, choisir les opérateurs, Planifier, définir les étapes du traitement, créer le dossier de production.	
	Poste conditions <Notification>	le Dossier de production est envoyé à l'atelier	
Exceptions <BusinessAction>*			
Nature de l'activité	manuelle	Automatisable : x	
Règles métiers	Il est nécessaire que la date de fin de réalisation soit inférieure à la date de livraison souhaitable.		
Objets d'entreprise associés <EnterpriseObject>*	Dossier de production, Planification des opérateurs et des machines		
Ressources associées < EnterpriseResource>*	La ressource	Contraintes liées aux ressources	
	Le tableau de planification		
Performance (<quantity>)? <TIME>	Secondes		
Fréquence <TIME_FREQUENCY>	Chaque jour		
Décision des parties prenantes <stakeholdersDecision>	À garder :	À réaliser: X	Ne pas considérer :
Priorité <quantity>	1		
Commentaire	/		

Ce dernier présente l'identifiant de l'exigence, sa version, les collaborateurs faisant partie des parties prenantes l'ayant exprimé, le poste et le rôle concernés par l'exigence où l'activité en question est réalisée, la séquence de déroulement. Y sont associés également les règles métiers, objets d'entreprises et ressources liés à cette exigence.

De plus, la nature manuelle ou automatisable de l'activité liée à l'exigence fonctionnelle est précisée ici. Les parties prenantes doivent préciser également si l'exigence est à réaliser ou

pas. Ils délimitent de ce fait l'étendue des fonctionnalités à réaliser et construisent ainsi le référentiel final des exigences et leur associent également des priorités pour préciser leur l'ordre de réalisation.

Nous permettons aussi à ce niveau de définir des exigences non fonctionnelles liées à chaque exigence telles que : des contraintes liées aux ressources utilisées et que le système doit prendre en compte, la performance attendue du système lors de l'exécution de la fonctionnalité ou encore la fréquence avec laquelle elle doit être réalisée. D'autres exigences non fonctionnelles peuvent également être exprimées indépendamment, elles sont relatives aux contraintes liées à la conduite du projet d'informatisation (coût, délai) ou à des propriétés relatives aux applicatifs à réaliser telles que la sécurité, l'interopérabilité, la portabilité, la confidentialité, l'intégrité, la disponibilité. Elles peuvent aussi être liées aux interfaces des applicatifs, leur installation ou leur distribution.

Nous proposons également à ce niveau de restituer l'ensemble des objets d'entreprise représentant les éléments du modèle d'entreprise qui caractérisent et différencient la PME. En effet, certains concepts des vues organisationnelle, du contexte et des ressources sont repris dans la vue informationnelle et sont généralisé par le concept d'objet d'entreprise (*EnterpriseObject*). Ce dernier regroupe : les éléments en entrée, en sortie ou manipulés au niveau des activités, les ressources, les produits, les services, les partenaires externes, Les entrées et sorties externes. Il regroupe également des éléments qui servent à structurer l'entreprise (*EnterpriseStructuringObject*) : les domaines, les cellules organisationnelles, les postes et les rôles. Ils représentent des « classes candidates » sur lesquelles les applicatifs doivent garder des informations. Comme, par exemple, le concept d'*opérateur* qui est un poste de travail dans le modèle de la PME SDT ou encore le *broyeur* qui est une ressource utilisée au niveau de ce poste. Il ne s'agit pas d'éléments définitifs de bases de données à gérer mais d'une première ébauche montrant les éléments qui peuvent donner lieu à une modélisation plus fine lors de la conception du modèle conceptuel de données associé aux applicatifs à réaliser. La MOA, assistée par la MOE, doit à ce niveau choisir, dans les objets d'entreprise recensés, les informations pertinentes pour son métier qui doivent être gérées et sauvegardées. Nous regroupons dans un *Template* (Tableau 14) les propriétés liées à chaque objet d'entreprise: son identifiant, sa version, les parties prenantes l'ayant exprimé, le poste et le rôle, s'il existe, pour qui elle est pertinente, l'ensemble des attributs qui la caractérisent, sa période de validité représentant le temps de stockage souhaitable, et la priorité liée à son implémentation. Les parties prenantes précisent ici si elles décident de la prendre en compte et de l'inclure à l'ensemble des classes appartenant potentiellement au modèle conceptuel de la solution.

Nous proposons donc à l'issue de cette démarche un référentiel d'exigences structurant des exigences fonctionnelles, non fonctionnelles et les objets d'entreprise que la MOE doit prendre en compte de manière contractuelle. Pour une meilleure lecture du référentiel et pour montrer les liens entre exigences, nous permettons de consulter les exigences liées à chaque poste de travail, à chaque processus ou celles exprimées par chaque partie prenante. Nous

pouvons aussi montrer les objets d'entreprise liés à chaque exigence fonctionnelle ou même lister, pour tout objet d'entreprise, les exigences fonctionnelles où il intervient.

Tableau 14. Template d'un objet d'entreprise.

Objet d'entreprise <Object UUID> <Statement>	InfoReq0012 Détails de la commande		
Version <expressionDate>	01/11/2015		
Auteurs <Stakeholder>*	Jules		
Postes de travail <Workstation>*	Responsable de production		
Nom <EnterpriseObject>	Détails de la commande		
Définition	/		
Ensemble d'informations <Information>*	produit, quantité, date de livraison souhaitable, CDC technique		
Décision des parties prenantes <stakeholdersDecision>	À garder :	À réaliser: X	Ne pas considérer :
Période de validité <quantity> <TIME> <STATE> <FREQUENCY>	toujours		
Priorité <quantity>	1		
Commentaire	/		

Le référentiel des exigences vérifié et validé est utilisé à l'issue du processus d'IBE par la MOE pour développer des prototypes qu'elle proposera à la MOA qui est donc susceptible d'itérer ce processus jusqu'à convergence. La production de ces prototypes sera facilitée grâce aux techniques de transformation de modèles et de génération de code, en effet l'ensemble des modèles que les éditeurs permettent de créer sont formalisés dans un langage (XMI) ouvert aux transformations.

5.3 Conclusion

Nous avons présenté dans ce chapitre une mise en œuvre de la méthode d'ingénierie des besoins et des exigences que nous proposons. Elle permet d'abord, d'élucider chez les collaborateurs de la PME leur besoins d'informatisation en utilisant un questionnaire et un DSL textuel. Ces derniers rendent intuitive la démarche d'expression : les collaborateurs sont guidés, ils décrivent des éléments qui leur sont familiers, relatifs à leur entreprise, de façon indépendante des solutions possibles et ce en utilisant un langage naturel. Lors de l'expression des besoins, des actions correctives sont nécessaires lorsque les règles de cohérence ou de complétude ne sont pas vérifiées et ce pour assurer les qualités attendues du référentiel d'exigences. Après la validation des besoins, grâce à la revue des restitutions, nous offrons le moyen à la PME en collaboration avec la MOE pour définir à partir d'un référentiel des besoins l'ensemble des exigences à réaliser et constituent de fait le référentiel d'exigences. Ce dernier reflète alors les consensus et décisions prises en commun et est à la

base d'un document contractuel définissant les rôles et responsabilités du commanditaire et du prestataire.

Conclusion générale, limites et perspectives

Dans cette thèse, nous avons abordé la phase d'ingénierie des besoins et des exigences (IBE) qui est une phase cruciale de tout projet d'informatisation. L'objectif était de répondre à un contexte industriel particulier : celui de la société RESULIS qui est une société de service et de conseil en informatique qui travaille essentiellement pour des PME souhaitant informatiser leurs processus de gestion. RESULIS est par conséquent maître d'œuvre de projets d'informatisation. La réussite de ces projets dépend fortement de la qualité des cahiers des charges (plus formellement appelé référentiel d'exigences) établis par les PME. Or ces dernières manquent de compétences quant à la définition du cahier des charges, de temps pour se former et de moyens financiers pour se faire assister. Par conséquent, ce sont les équipes de RESULIS qui établissent les exigences, de manière informelle, la plupart du temps ce qui ne permet pas d'assurer un niveau de qualité satisfaisant, et pose problème quant à l'adéquation des exigences avec la demande réelle de la PME. En outre, l'implication des collaborateurs de la PME est primordiale pour une meilleure compréhension des besoins et l'évitement de toute fausse interprétation.

De ce fait, nous proposons de fournir à RESULIS des moyens conceptuels, méthodologiques et des outils techniques pour supporter et formaliser les étapes d'expression, de vérification et de validation des besoins et des exigences tout en impliquant les collaborateurs de la PME. Nous offrons à ces derniers des moyens favorisant leur autonomie lors de l'expression des besoins d'informatisation.

De nombreuses approches ont été proposées pour supporter le processus d'ingénierie des besoins et des exigences. Nous en avons étudié les principales. Toutefois, il n'y a pas à notre connaissance de méthodes d'IBE permettant aux clients non experts d'exprimer leurs besoins de manière quasi-autonome. En effet, la société RESULIS s'est fixée comme objectifs de premièrement accompagner les collaborateurs de la PME pour leur permettre de décrire leurs besoins dans leur langage métier, tout en leur laissant une certaine autonomie. Le second objectif est, à partir des besoins, de pouvoir formuler des exigences possédant certaines propriétés qui en garantissent l'approbation par ces mêmes collaborateurs. Finalement RESULIS souhaite que ses équipes de développement puissent utiliser ces exigences sans perte d'information pour entamer les étapes de réalisation.

Pour ce faire, notre contribution s'articule autour de la proposition d'une méthode d'ingénierie des besoins et des exigences. La méthode repose sur un socle de concepts et de relations, le métamodèle SI-PME, réunissant les concepts et relations de la modélisation d'entreprise d'une part et de l'ingénierie des besoins et des exigences d'autre part. Ce modèle conceptuel permet d'assister voire d'automatiser certaines étapes de modélisation, de vérification et de transformation de modèles, restitués pour la validation.

Les concepts et relations du métamodèle SI-PME sont instanciés grâce aux langages de modélisation que nous avons définis. Les activités de la méthode d'IBE nécessitent de créer et de manipuler des modèles rendant compte des besoins des collaborateurs de l'entreprise et permettant de construire un référentiel d'exigences. Nous avons donc proposé un DSL tex-

tuel pour simplifier l'activité d'expression des besoins par les collaborateurs de la PME et des DSL graphiques à la syntaxe concrète simplifiée permettant la revue des modèles recueillis.

Nous avons également défini une démarche opératoire. Le processus d'ingénierie des besoins et des exigences que nous proposons s'articule en un ensemble d'activités d'expression, de vérification, de validation, et de prise de décision. Il permet d'accompagner la maîtrise d'ouvrage et la maîtrise d'œuvre dans la définition des exigences à réaliser à partir des besoins collectés. Ces derniers sont vérifiés et validés avant toute prise de décision ou utilisation.

Enfin, nous proposons un outillage informatique, supportant cette démarche, composé de plusieurs outils qui ont été développés et intégrés dans un prototype démonstrateur sous forme d'une extension de l'environnement de modélisation d'Eclipse.

Nous avons démontré au travers d'un cas d'étude mené avec un des clients de RESULIS que nos propositions couvraient les attentes de ce client en matière de recueil et de validation des besoins exprimées.

Les travaux réalisés présentent des limites pour lesquelles nous donnons des pistes de solutions, qui feront l'objet de travaux futurs.

Pour améliorer le cadre d'expression des besoins, il serait opportun de prendre en compte les exigences non fonctionnelles. Cela permettra donc d'élargir l'éventail des parties prenantes concernées dans la phase d'expression des besoins et d'y intégrer, de fait, la MOE elle-même qui deviendrait alors apte à décrire ses contraintes de développement. Les exigences non fonctionnelles pourraient faire l'objet d'un DSL textuel constitué d'un ensemble de *boilerplates* spécifiques comme nous l'avons fait pour les exigences fonctionnelles.

Le niveau partiel de généricité du cadre normatif de la norme ISO 19439 (*Framework for Enterprise Modelling*), sur lequel nous nous basons pour définir les concepts et phases de modélisation, peut être davantage exploité et enrichi. En effet, l'axe de généricité représente le processus d'instanciation des modèles génériques et partiels en modèles spécifiques à l'entreprise. Nous pourrions définir au niveau partiel de généricité des typologies d'entreprises ayant un secteur d'activité commun (finances, santé, agroalimentaire, etc.) ou définir des fonctions communes à plusieurs entreprises (facturation, planification, vente en ligne, etc.). Nous constituerons, de ce fait, des modèles partiels de référence réutilisables par le biais de mécanismes de paramétrage.

La couverture fonctionnelle du prototype obtenu à partir des exigences est complétée au fur et à mesure des itérations de la méthode de développement agile. Le référentiel des exigences est donc remis en cause à chaque nouvelle itération (par exemple par ajout d'exigences induites ou modification d'exigences existantes suite à des choix technologiques). Il doit servir de fait de pivot pour le développement d'un système de plus en plus complet, où la maîtrise d'ouvrage est sollicitée continuellement pour valider l'adéquation du système avec ces exigences. Nous devons donc fournir des mécanismes de traçabilité des exigences et de leurs modifications. Il s'agit également de proposer des moyens de rétro-

ingénierie pour gérer les impacts des modifications effectuées au niveau de la conception ou de la réalisation pour actualiser les exigences à chaque itération.

La méthode d'IBE proposée permet actuellement de décrire l'existant. RESULIS peut également aider la PME à formuler sa réflexion sur le modèle cible de l'entreprise lorsque cette dernière souhaite définir de nouvelles pratiques. La PME peut avoir des objectifs d'évolution, elle doit alors identifier les nouveaux ou futurs profils de partenaires externes, de produits ou de services et définir les processus permettant de les réaliser. De plus, l'informatisation peut remettre en question les processus ou activités manuelles existantes ou les rendre obsolètes. Ces derniers peuvent aussi être améliorés et optimisés. Nous devons donc proposer des moyens pour apporter des modifications aux modèles existants. Pour ce faire, les modèles restitués, tels que les modèles de processus, représentent un support adéquat de réflexion et de prise de décisions. Un décideur de la PME peut les utiliser en collaboration avec RESULIS pour définir les nouvelles pratiques ou apporter des modifications nécessaires suite à l'informatisation. Ces modifications doivent également être prises en compte au niveau du référentiel des exigences.

La rigueur et le niveau de qualité des vérifications proposées pourraient être améliorés par une utilisation plus poussée de méthodes formelles : même si un langage formel est difficilement accessible à l'ensemble des parties prenantes car souvent muni d'une syntaxe hermétique, il paraît pertinent de passer par des transformations formelles des modèles actuels vers des modèles tiers supportant des techniques de preuve de propriétés par exemple.

Pour permettre à l'équipe de développement de RESULIS d'exploiter au mieux les modèles établis durant la phase d'IBE, des mécanismes de transformations vers des modèles de type PIM et vers le code pourraient être envisagés. Les modèles que nous construisons sont positionnés au niveau CIM (*Computation Independent model*). Le passage des exigences aux modèles d'analyse (CIM vers PIM (*Platform Independent Model*)) n'est pas couvert par la plupart des méthodes d'ingénierie dirigée par les modèles (Yue et al. 2011) qui se concentrent plutôt sur les transformations PIM vers PSM (*Platform Specific Model*). Ceci est dû au caractère informel et ambigu des exigences écrites en langage naturel libre et qui ne se prêtent donc pas aux transformations automatiques. Notre utilisation d'un langage naturel contrôlé permet de lever cette ambiguïté et favorise donc ces transformations. De plus, Les modèles construits avec les DSL graphiques que nous proposons peuvent couvrir partiellement le passage des CIM vers PIM en transformant les modèles d'exigences en une première version à compléter de modèles d'analyse. Nous devons également automatiser ces transformations comme proposé au chapitre IV. Ces modèles donnent une vision structurée de l'applatif recherché au sein du SI global, toujours indépendamment de la conception technique de la solution.

Références

- Aceituna, D., Walia, G., Do, H., & Lee, S. W. (2014). Model-based requirements verification method: Conclusions from two controlled experiments. *Information and Software Technology*, 56(3), 321-334.
- Acharya, S., Mohanty, H., & George, C. (2005, September). Domain consistency in requirements specification. In *Quality Software, 2005.(QSIC 2005). Fifth International Conference on* (pp. 231-238). IEEE.
- de Almeida Ferreira, D., & da Silva, A. R. (2009, September). A controlled natural language approach for integrating requirements and model-driven engineering. In *Software Engineering Advances, 2009. ICSEA'09. Fourth International Conference on* (pp. 518-523). IEEE.
- AMICE, ESPRIT Consortium AMICE. (Ed.), (1993). CIMOSA: open system architecture for CIM (Vol. 1). Springer.
- Amokrane, N., Chapurlat, V., Courbis, A.L., Lambolais T., Rahhou, M. (2014). Modeling Frameworks, Methods and Languages for Computerizing Small and Medium-Sized Enterprises: Review and Proposal. *Interoperability for Enterprise Systems and Applications (I-ESA)*, (pp. 77–87).
- Amokrane, N., Chapurlat, V., Courbis, A. L., Lambolais, T., & Rahhou, M. (2015). Requirements Authoring and Verification for SMEs' Information Systems Engineering. *IFAC-PapersOnLine*, 48(3), (pp. 2238-2243).
- Annervaz, K. M., Kaulgud, V., Sengupta, S., & Savagaonkar, M. (2013, November). Natural language requirements quality analysis based on business domain models. In *Automated Software Engineering (ASE), 2013 IEEE/ACM 28th International Conference on* (pp. 676-681). IEEE.
- Anton, A. I. (1996, April). Goal-based requirements analysis. In *Requirements Engineering, Proceedings of the Second International Conference on* (pp. 136-144). IEEE.
- Beck, K. (2000). *Extreme programming explained: embrace change*. addison-wesley professional.
- Bell, T.E. & Thayer, T.A., (1976). Software requirements: Are they really a problem? *Proc. ICSE-2: 2nd Intrnational Conference on Software Engineering, San Francisco*, pp.61–68.
- Benner, K. & Feather, M., (1993). Utilizing scenarios in the software development process. *Information System Developpement Process*, pp.117–134.
- Berio, G. & Vernadat, F.B., (1999). New developments in enterprise modelling using CIMOSA. *Computers in Industry*, 40, pp.99–114.
- Bernus, P. & Nemes, L., (1997). The contribution of the generalised enterprise reference architecture to consensus in the area of enterprise integration. *Proceedings of ICEIMT97, Kosanke K. and Nell J.G. (Eds.) Enterprise Engineering and Integration: Building International Consensus.*, pp.175–189.
- Bernus, P., Noran, O., & Molina, A. (2014). Enterprise architecture: twenty years of the GERAM framework. *Annual Reviews in Control*, 39, 83-93.
- Besnard, S., Biré, C., & Victor, P. (2008). L'intégration des TIC est encore incomplète dans les entreprises. *Insee première*, (1184).

- BKCASE Editorial Board, (2015). Guide to the Systems Engineering Body of Knowledge -- G2SEBoK.
- Blanc, X., Mounier, I., Mougnot, A., & Mens, T. (2008, May). Detecting model inconsistency through operation-based model construction. In *Software Engineering, 2008. ICSE'08. ACM/IEEE 30th International Conference on* (pp. 511-520). IEEE.
- Boehm, B. W. (1981). *Software engineering economics* (Vol. 197). Englewood Cliffs (NJ): Prentice-hall.
- Bourey, J. P., Grangel, R., Ducq, Y., Berre, A., Bertoni, M., D'Antonio, F. & Pantelopoulos, S. (2007). Report on model driven interoperability. INTEROP
- Braesch, C., Haurat, A., & Beving, J. M. (1995). L'entreprise-système. *La modélisation systématique en entreprise*, 83-88.
- Brocke, J. Vom, & Rosemann, M. (2010). *Handbook on business process management*. Heidelberg: Springer.
- Brottier, E., (2009). *Acquisition et analyse des exigences pour le développement logiciel : une approche dirigée par les modèles*. UNIVERSITÉ DE RENNES 1.
- Bruza, P. D., & Van der Weide, T. P. (1989). *The semantics of data flow diagrams*. University of Nijmegen, Department of Informatics, Faculty of Mathematics and Informatics.
- Bryant, B. R., Lee, B. S., Cao, F., Zhao, W., & Gray, J. G. (2003). *From natural language requirements to executable models of software components*. INDIANA UNIV-PURDUE UNIV AT INDIANAPOLIS.
- Cao, L., & Ramesh, B. (2008). Agile requirements engineering practices: An empirical study. *Software, IEEE*, 25(1), 60-67.
- De Gea, J. M. C., Nicolás, J., Alemán, J. L. F., Toval, A., Ebert, C., & Vizcaíno, A. (2012). Requirements engineering tools: Capabilities, survey and assessment. *Information and Software Technology*, 54(10), 1142-1157.
- CESAR Consortium. (2011). Definition of RSL and RMM D_SP2_R2.3_M3; Dec.
- Chapurlat V. (2007). Vérification et validation de modèles de systèmes complexes : application à la Modélisation d'Entrepise. HDR Université de Montpellier ;.
- Chapurlat, V., Kamsu-Foguem, B., & Prunet, F. (2006). A formal verification framework and associated tools for enterprise modeling: Application to UEML. *Computers in industry*, 57(2), 153-166.
- Chemuturi, M. (2012). *Requirements engineering and management for software development projects*. Springer Science & Business Media.
- Chen, D., Vallespir, B., & Doumeingts, G. (1997). GRAI integrated methodology and its mapping onto generic enterprise reference architecture and methodology. *Computers in industry*, 33(2), 387-394.
- Chen, P. P. S. (1976). The entity-relationship model—toward a unified view of data. *ACM Transactions on Database Systems (TODS)*, 1(1), 9-36.
- Chung, L., Nixon, B. A., Yu, E., & Mylopoulos, J. (2012). *Non-functional requirements in software engineering* (Vol. 5). Springer Science & Business Media.

- Cragg, P., Caldeira, M. & Ward, J., (2011). Organizational information systems competences in small and medium-sized enterprises. *Information & Management*, 48(8), pp.353–363.
- Doumeingts, G., Vallespir, B., Zanettin, M., & Chen, D. (1992). *GIM: GRAI Integrated Methodology, a methodology for designing CIM systems*, GRAI. LAP, Technical Report.
- Doumeingts, G., 1984. Méthode GRAI, Méthode de Conception et de Spécification des Systèmes de Productique.
- Fanmuy, G., Fraga, A. & Llorens, J., (2012). Requirements Verification in the Industry. In O. Hammami, D. Krob, & J.-L. Voirin, eds. *Complex Systems Design & Management*. Springer Berlin Heidelberg, pp. 145–160.
- Farfeleder, S., Moser, T., Krall, A., Stålhane, T., Omoronyia, I., & Zojer, H. (2011). Ontology-driven guidance for requirements elicitation. In *The semantic web: research and applications* (pp. 212-226). Springer Berlin Heidelberg.
- Faure, P., (2010). Le tableau de bord des TIC dans les entreprises.
- Finkelstein, A., Kramer, J., Nuseibeh, B., Finkelstein, L., & Goedicke, M. (1992). Viewpoints: A framework for integrating multiple perspectives in system development. *International Journal of Software Engineering and Knowledge Engineering*, 2(01), 31-57.
- Fischer, K. (2005). Advanced technologies for interoperability of heterogeneous enterprise networks and their applications. *European Coordination Action for Agent-based Computing*, 34.
- Fox, M.S. & Gruninger, M., (1998). Enterprise Modeling. *AI Magazine*, 19(3), p.109.
- Fuchs, N. E., Kaljurand, K., & Schneider, G. (2006). Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces. In *FLAIRS Conference* (Vol. 12, pp. 664-669).
- Gane, C. P., & Sarson, T. (1979). *Structured systems analysis: tools and techniques*. Prentice Hall Professional Technical Reference.
- Grangel, R., Salem, R. B., Bourey, J. P., Daclin, N., & Ducq, Y. (2007). Transforming GRAI Extended Actigrams into UML Activity Diagrams: A first step to model driven interoperability. In *Enterprise Interoperability II* (pp. 447-458). Springer London.
- Hall, J. (2006, June). Semantics of Business Vocabulary and Business Rules (SBVR). In *Online*, [Retrieved January 16, 2013], http://www.omg.org/news/meetings/ThinkTank/past-events/2006/presentations/04-WS1-2_Hall.pdf.
- Hull, E., Jackson, K., & Dick, J. (2010). *Requirements engineering*. Springer Science & Business Media.
- Ibanez, M., & Rempp, H. (1996). European user survey analysis. *ESPITI project report, February*.
- IEEE Computer Society. Software Engineering Standards Committee, & IEEE-SA Standards Board. (1998). IEEE recommended practice for software requirements specifications. Institute of Electrical and Electronics Engineers.
- INCOSE, (2012). Guide for Writing Requirements.

- INSEE, (2013). *Enquête sur les technologies de l'information et de la communication et le commerce électronique 2012*
- ISO/DIS (2002) CIM Systems Architecture — Framework for enterprise modelling. ISO/DIS 19439, ISO/TC 184/SC 5.
- ISO/DIS (2004) Enterprise integration - Constructs for enterprise modelling. ISO/DIS 19440, ISO/TC 184/SC 5.
- ISO15288, (2008). *Systems Engineering – System Life-Cycle Processes*,
- ISO15704, (2000). *Industrial automation systems – Requirements for enterprise-reference architectures and methodologies*;
- ITU-T, (2008). 151 (11/08), User Requirements Notation (URN)—Language definition. *Geneva, Switzerland, approved November*.
- Jackson, M. (1997). The meaning of requirements. *Annals of Software Engineering*, 3(1), 5-21.
- Jacobson, I., Booch, G., Rumbaugh, J., Rumbaugh, J., & Booch, G. (1999). *The unified software development process* (Vol. 1). Reading: Addison-Wesley.
- Johannessen, V. (2012). CESAR-text vs. boilerplates: What is more efficient-requirements written as free text or using boilerplates (templates)?.
- Junginger, S. (2000). The workflow management coalition standard wpdL: First steps towards formalization. *Proceedings of ECEC*, 163-168.
- Kamalrudin, M., & Sidek, S. (2015). A Review on Software Requirements Validation and Consistency Management. *International Journal of Software Engineering and Its Applications*, 9(10), 39-58.
- Kent, S. (2002, May). Model driven engineering. In *Integrated formal methods*(pp. 286-298). Springer Berlin Heidelberg.
- Kotonya, G., & Sommerville, I. (1996). Requirements engineering with viewpoints. *Software Engineering Journal*, 11(1), 5-18.
- Kruchten, P. (2004). *The rational unified process: an introduction*. Addison-Wesley Professional.
- Kuras, M. L., & White, B. E. (2005, July). Engineering enterprises using complex-system engineering. In *INCOSE Symposium* (Vol. 10, p. 14).
- Lacombe, C., Pochelu, R., Tazi, S., & Ducq, Y. (2014). Model-Driven Enterprise Resource Planning Specifications in SMEs. In *Advances in Production Management Systems. Innovative and Knowledge-Based Production Management in a Global-Local World* (pp. 538-545). Springer Berlin Heidelberg.
- Lamsweerde, A. Van, (2001). Goal-oriented requirements engineering: A guided tour. In *Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on* (pp. 249-262). IEEE.
- Lamsweerde, A. Van, (2000, June). Requirements engineering in the year 00: a research perspective. In *Proceedings of the 22nd international conference on Software engineering* (pp. 5-19). ACM.

- Lamsweerde, A. Van, (2009). Requirements engineering: from system goals to UML models to software specifications.
- Lamsweerde, A. Van, Dardenne, A., Delcourt, B., & Dubisy, F. (1991). The KAOS project: Knowledge acquisition in automated specification of software. In *Proceedings of the AAAI Spring Symposium Series*.
- Leite, J. C. S. (1989, April). Viewpoint analysis: a case study. In *ACM SIGSOFT Software Engineering Notes* (Vol. 14, No. 3, pp. 111-119). ACM.
- Le Moigne, J. L. (1994). *La théorie du système général: théorie de la modélisation*.
- Lemrabet, Y., Liu, H., Bourey, J. P., & Bigand, M. (2012). Proposition of business process modelling in model driven interoperability approach at CIM and PIM levels. In *Enterprise Interoperability V* (pp. 203-215). Springer London.
- Litvak, B., Tyszberowicz, S., & Yehudai, A. (2003, September). Behavioral consistency validation of UML diagrams. In *Software Engineering and Formal Methods, 2003. Proceedings. First International Conference on* (pp. 118-125). IEEE.
- Liu, L., & Yu, E. (2001, May). From requirements to architectural design-using goals and scenarios. In *ICSE-2001 Workshop: From Software Requirements to Architectures (STRAW 2001) May* (pp. 22-30).
- Loucopoulos, P., & Champion, R. E. M. (1989). Knowledge-based support for requirements engineering. *Information and software technology*, 31(3), 123-135.
- Lúcio, L., Amrani, M., Dingel, J., Lambers, L., Salay, R., Selim, G. M., & Wimmer, M. (2014). Model transformation intents and their properties. *Software & Systems Modeling*, 1-38.
- Luoma, J., Kelly, S., & Tolvanen, J. P. (2004, October). Defining domain-specific modeling languages: Collected experiences. In *4 th Workshop on Domain-Specific Modeling*.
- Marca, D. A., & McGowan, C. L. (1987). SADT: structured analysis and design technique. McGraw-Hill, Inc.
- Mavin, A., Wilkinson, P., Harwood, A., & Novak, M. (2009, August). Easy approach to requirements syntax (EARS). In *Requirements Engineering Conference, 2009. RE'09. 17th IEEE International* (pp. 317-322). IEEE.
- Menzel, C., & Mayer, R. J. (1998). The IDEF family of languages. In *Handbook on architectures of information systems* (pp. 209-241). Springer Berlin Heidelberg.
- Mernik, M., Heering, J., & Sloane, A. M. (2005). When and how to develop domain-specific languages. *ACM computing surveys (CSUR)*, 37(4), 316-344.
- Millet, P. A. (2008). Une étude de l'intégration organisationnelle et informationnelle Application aux systèmes d'informations de type ERP (Doctoral dissertation, INSA de Lyon).
- Morley, C., Hugues, J. & Leblanc, B., (2005). *Processus Métiers et systèmes d'information: Evaluation, modélisation, mise en oeuvre*. Dunod.
- Al Mrayat, O.I., Norwawi, N. & Basir, N., (2013). Requirements Elicitation Techniques : Comparative Study. *International Journal of Recent Development in Engineering and Technology*, 1(3), pp.1-10.

- Mullery, G. P. (1979). CORE-a method for controlled requirement specification. In *Proceedings of the 4th international conference on Software engineering* (pp. 126-135). IEEE Press..
- Nikula, U., Sajaniemi, J., & Kälviäinen, H. (2000). Management view on current requirements engineering practices in small and medium enterprises. In *Queensland University of Technology*.
- Njonko, P. B. F., Cardey, S., Greenfield, P., & El Abed, W. (2014). RuleCNL: A Controlled Natural Language for Business Rule Specifications. In *Controlled Natural Language* (pp. 66-77). Springer International Publishing.
- OMG, (2008). Semantics of Business Vocabulary and Business Rules (SBVR), v1.0. , (January).
- OMG, (2011). BPMN. Available at: <http://www.bpmn.org/>.
- OSEO, (2012). *Rapport sur l'évolution des PME*.
- Pagès, R.R., (2009). An object-oriented approach to the translation between MOF Metaschemas.
- Pfeiffer, M. & Pichler, J., (2008). A comparison of tool support for textual domain-specific languages. *Proceedings of the 8th OOPSLA Workshop on Domain-Specific Modeling*, pp.1–7.
- Planson, S., Jean-Philippe, R. & Tagnani, S., (2013). *Les entreprises en France*.
- Pourcel, G., & Gourc, D. (2002). Modélisation d'entreprise: la méthode MECL. *Ecole de printemps Modélisation d'entreprise d'Albi-Carmaux*, 28-30.
- Rolland, C., (2003). Ingénierie des Besoins: L'Approche L'Ecritoire. *Journal Techniques de l'Ingénieur*, pp.1–45.
- Rolland, C., Grosz, G., & Kla, R. (1999). Experience with goal-scenario coupling in requirements engineering. In *Requirements Engineering, 1999. Proceedings. IEEE International Symposium on* (pp. 74-81). IEEE.
- Roques, P., & Vallée, F. (2011). *UML 2 en action: de l'analyse des besoins à la conception*. Editions Eyrolles.
- Ross, D. T., & Schoman Jr, K. E. (1977). Structured analysis for requirements definition. *Software Engineering, IEEE Transactions on*, (1), 6-15.
- Sabetzadeh, M., Nejati, S., Liaskos, S., Easterbrook, S., & Chechik, M. (2007). Consistency checking of conceptual models via model merging. In *Requirements Engineering Conference, 2007. RE'07. 15th IEEE International* (pp. 221-230). IEEE.
- Scheer, A. W. (1994). ARIS toolset: a software product is born. *Information Systems*, 19(8), 607-624.
- Schneider, F., Naughton, H., & Berenbach, B. (2012). A modeling language to support early lifecycle requirements modeling for systems engineering. *Procedia Computer Science*, 8, 201-206.
- Schwaber, K. & Beedle, M., (2002). *Scrum*.
- Scukanec, M. & Gaasbeek, J. van, (2007). A day in the life of a verification requirement. *2008 US Air Force T&E Days*.

- Scukanec, S. J., & Gaasbeek, J. R. (2010, July). A day in the life of a verification requirement. In *INCOSE International Symposium* (Vol. 20, No. 1, pp. 2524-2542).
- Sharp, H., Finkelstein, A., & Galal, G. (1999). Stakeholder identification in the requirements engineering process. In *Database and Expert Systems Applications, 1999. Proceedings. Tenth International Workshop on* (pp. 387-391). IEEE.
- Sinha, A., Kaplan, M., Paradkar, A., & Williams, C. (2008). Requirements modeling and validation using bi-layer use case descriptions. In *Model Driven Engineering Languages and Systems* (pp. 97-112). Springer Berlin Heidelberg.
- Sirius, (2013). Sirius. Available at: <https://eclipse.org/sirius/> [Accessed August 5, 2015].
- Stålhane T., Farfeleder S. & Daramola O. (2011). "Safety analysis based on requirements". Norwegian University of Science and Technology and Vienna University of Technology.
- Stapleton, J. (1997). *DSDM, dynamic systems development method: the method in practice*. Cambridge University Press.
- Sutcliffe, A. (1998). Scenario-based requirements analysis. *Requirements engineering*, 3(1), 48-65.
- The Standish Group, (2009). *The Chaos Report*, Available at: <http://www.standishgroup.com/>.
- Tommila, T., Pakonen, A., & Valkonen, J. (2013). Ontology-Driven Natural Language Requirement Templates for Model Checking I&C Functions. In *Enlarged Halden programme group meeting, EHPG-2013, Storefjell, Norway, 10th–15th March*.
- TreeTagger, (1994). TreeTagger. Available at: <http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/>.
- Vangheluwe, H. & Meyers, B., 2014. Modelling Language Evolution. , (August), pp.1–29.
- Vernadat, F. (1999). Techniques de modélisation en entreprise: applications aux processus opérationnels. Economica.
- Vernadat, F. (2002). UEML: towards a unified enterprise modelling language. *International Journal of Production Research*, 40(17), 4309-4321.
- Wikipédia, Méthode agile. Available at: https://fr.wikipedia.org/wiki/M%C3%A9thode_agile [Accessed August 4, 2015].
- Williams, S., Power, R., & Third, A. (2014). How easy is it to learn a controlled natural language for building a knowledge base?. In *Controlled Natural Language* (pp. 20-32). Springer International Publishing.
- Williams, T. J. (1994). The Purdue enterprise reference architecture. *Computers in industry*, 24(2), 141-158.
- Xtext, (2010). LANGUAGE DEVELOPMENT MADE EASY! Available at: <https://eclipse.org/Xtext/>.
- Yan, R., Cheng, C. H., & Chai, Y. (2015). Formal consistency checking over specifications in natural languages. In *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015* (pp. 1677-1682). IEEE.

- Yue, T., Briand, L. C., & Labiche, Y. (2011). A systematic review of transformation approaches between user requirements and analysis models. *Requirements Engineering*, 16(2), 75-99.
- Zelm, M., Vernadat, F. B., & Kosanke, K. (1995). The CIMOSA business modelling process. *Computers in industry*, 27(2), 123-142.

Résumé : Une entreprise, et particulièrement une PME ou une PMI, doit être apte à évoluer sur des secteurs d'activités souvent très concurrentiels qui évoluent rapidement, par exemple, en fonction d'une clientèle volatile et soucieuse de trouver des produits et des services moins chers et plus adaptés à ses besoins. La PME se trouve alors confrontée à des problèmes de réactivité et de flexibilité face à cette clientèle. Par effet direct, elle recherche à réduire les délais et les coûts de réalisation tout en privilégiant aussi la qualité et le degré d'innovation des biens et des services qu'elle propose. Le système d'information de cette PME est un enjeu essentiel pour mettre en œuvre cette stratégie et maximiser donc la réactivité et la flexibilité mais aussi la rentabilité et la qualité recherchées. Ce sont des qualités incontournables, garantes d'une autonomie et d'une reconnaissance dont la PME a grand besoin. Le système d'information est en effet la courroie de transmission entre le système de décision et le système productif qui génère la valeur ajoutée de la PME. Il fait le lien entre les différents acteurs de l'organisation mais aussi avec les partenaires externes de l'entreprise. Une partie de ce système d'information est de fait informatisée. Celui-ci supporte, mémorise et traite les informations nécessaires aux différents processus de décision, métiers et supports qui tapissent l'organisation pour servir la stratégie de l'entreprise. Les fonctionnalités, les interfaces et les données qui forment ce système informatisé sont donc cruciales à comprendre, à développer en accord avec les besoins de la PME, à améliorer au fur et à mesure de l'évolution de ces besoins. La PME est donc tentée de se lancer, seule ou accompagnée, dans des projets dits d'informatisation i.e. des projets visant le développement ou l'amélioration de son système informatisé. Nous nous intéressons ici à des projets visant à développer des applicatifs de gestion et de pilotage de la PME. La PME – prenant alors le rôle de la maîtrise d'ouvrage (MOA) – tout comme la société de services qui l'accompagne – prenant alors en charge le rôle de maîtrise d'œuvre (MOE) – doivent partager une vision commune des besoins d'informatisation. Elles sont alors appelées à mener en commun des activités d'ingénierie des besoins et des exigences (IBE). L'IBE guide et accompagne la PME pour arriver à décrire et formaliser ses **besoins**. Elle permet ensuite à la société de service de spécifier de manière plus formelle ces besoins sous forme d'**exigences** qui définissent alors les travaux de développement souhaités. L'IBE est souvent réalisée avec une assistance à maîtrise d'ouvrage. Cette étape cruciale reste cependant difficile pour une PME. Elle est de plus souvent réalisée par la MOE elle-même pour faire face au manque de moyens, de temps et de compétences de la PME. Or, l'implication des collaborateurs de la PME est primordiale pour la réussite de tout projet d'informatisation, surtout si celui-ci impacte durablement le fonctionnement de la PME. Ces travaux, développés dans le cadre d'une collaboration Industrie/recherche avec la SSII RESULIS, ont consisté à développer une méthode d'IBE qui offre aux PME des concepts, des langages et des moyens de modélisation et de vérification simples mais suffisants tout en tant aisément manipulables de manière intuitive et donnant lieu à une formalisation pertinente pour la MOE. Cette méthode est basée sur le croisement et la complémentarité de principes issus de la Modélisation d'Entreprise et de l'Ingénierie Système pour l'expression de besoins. Des moyens de vérification et de validation semi-formels sont appliqués pour garantir certaines qualités attendues des exigences résultantes. La méthode s'intègre également au cycle de développement basé sur les modèles pour permettre *a posteriori* d'accélérer la production de prototypes et de rendre interoperables les langages et outils de la MOA et de la MOE.

Abstract: Most enterprises, and especially SMEs, must develop their business in very competitive and rapidly changing fields. Where they have to adapt to volatile customers who want to find cheaper products and services and that are more corresponding to their needs. The SME is then confronted with problems of responsiveness and flexibility in responding to these customers. As an effect, it seeks to reduce the costs and time to market and to provide high quality and innovative goods and services. The SME's information system is an asset on which it can rely to implement this strategy and so to maximize its responsiveness and flexibility but also to reach the sought profitability and quality. These are key qualities that guarantee autonomy and recognition, qualities that are highly needed by any SME. The Information system is indeed the drive belt of information not only inside the enterprise, between decision and operational systems proving added value of the enterprise, but also within its environment that includes its external partners. Part of this information system is computerized. It stores and processes the information needed by the different decision-making, business and support processes that serve the enterprise's strategy. It is crucial to understand the features, interfaces and data that make up this computerized system and develop them according to the needs of SME. The SME is therefore tempted to embark, alone or accompanied, in so-called computerization projects i.e. projects for the development or improvement of its computerized system. We are interested in projects aimed at developing management applications of SMEs. The SME – then assuming the role of project owner – along with the development team – supporting the role of project management – have to share a common vision of the computerization needs. They are then called upon to carry out jointly requirements engineering (RE) activities. RE guides the SMEs to be able to describe and formalize its needs. It then allows the development team to specify more formally these needs as requirements which then define the required development work. RE is often carried out with the assistance of project owner support. This crucial step remains difficult for SMEs. It is most often performed by the development team itself to address the lack of resources, time and skills of SMEs. However, the involvement of the SME's members is vital to the success of any computerization project, especially if it permanently affects the functioning of the enterprise. This work, developed through a collaborative with the company RESULIS, consisted in developing a requirements engineering method which offers SMEs concepts, simple languages, modeling and verification means that are easily and intuitively manipulated and provide sufficient and relevant formalization of the SME's requirements. This method is based on principles derived from both enterprise modeling and systems engineering fields for requirements elicitation. Semi-formal verification and validation means are applied to guarantee some expected qualities of the resulting requirements. The method is also integrated in the model driven development cycle to enable *a posteriori* the production of prototypes and make interoperable the languages and tools used by both the SME and the development team.