

Université  
de Toulouse

# THÈSE

En vue de l'obtention du  
**DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE**

**Délivré par :**  
Institut National Polytechnique de Toulouse (INP Toulouse)

**Discipline ou spécialité :**  
Systèmes industriels

---

**Présentée et soutenue par :**  
Jean-Stéphane ULMER

**le :** vendredi 11 février 2011

**Titre :**

Approche générique pour la modélisation et l'implémentation des processus.

---

**JURY**

Jean-Pierre Belaud  
Hervé Pingaud  
Pascale Zarate  
Bruno Traverson

---

**Ecole doctorale :**  
Systèmes (EDSYS)

**Unité de recherche :**  
Laboratoire de Génie Chimique UMR 5503

**Directeur(s) de Thèse :**

Jean-Marc Le Lann

**Rapporteurs :**  
Xavier Boucher  
Jean-Claude Bocquet



# Résumé

Une entreprise doit être capable de décrire et de demeurer réactive face à un événement endogène ou exogène. Une telle flexibilité peut s'obtenir par la gestion des processus d'entreprise (Business Process Management - BPM). Lors d'une démarche BPM, différentes transformations interviennent sur les modèles de processus développés par l'analyste métier et l'expert en technologies de l'information. Un non-alignement se crée entre ces modèles hétérogènes lors de leurs manipulations : il s'agit du "fossé métier-TI" tel que décrit dans la littérature. L'objectif de notre travail est de proposer un cadre méthodologique permettant un meilleur pilotage des processus métier, afin de tendre vers un alignement systématique de leur modélisation à leur implémentation au sein du système cible. A l'aide de concepts issus de l'ingénierie d'Entreprise et des Systèmes d'Informations dirigée par les modèles et des TI, nous définissons une démarche générique assurant une cohérence intermodèle. Son rôle est de conserver et de fournir toutes les informations liées à la structure et à la sémantique des modèles. En permettant la restitution intégrale d'un modèle transformé au sens de l'ingénierie inverse, notre plateforme permet une synchronisation entre modèle d'analyse et modèle d'implémentation. Le manuscrit présente également l'adéquation possible entre l'ingénierie des procédés et le BPM à travers un point de vue multi-échelle.

## Mots-clés

Ingénierie de systèmes d'information - Gestion des processus d'entreprise - Business Process Modeling Notation - Ingénierie dirigée par les modèles - Alignement de modèles – Métamodélisation - Génie des procédés



# Remerciements

Ces quelques lignes marquent le point final de ce travail de thèse ainsi que l'aboutissement de trois années (et quelques mois) de recherche, de découvertes et de collaborations réalisées au sein du département Procédés et Systèmes Industriels du Laboratoire de Génie Chimique.

Ce travail de thèse a été accompli sous la direction de Messieurs Jean-Pierre Belaud, Maître de Conférence au département PSI, et Jean-Marc Le Lann, Professeur d'Université et directeur de l'ENSIACET. Je tiens particulièrement à les remercier de m'avoir permis d'avoir la plus liberté dans la conduite de mes travaux et encouragé dans cette difficile démarche consistant à explorer des approches nouvelles à la frontières entre disciplines distinctes.

Je remercie mes rapporteurs pour toute l'attention qu'ils ont porté à ma thèse : Messieurs Xavier Boucher, Professeur associé à l'ENSM et Jean-Claude Bocquet, Professeur d'Université et directeur du LGI de l'Ecole Centrale Paris. Je remercie également les membres du jury : Monsieur Hervé Pingaud, Madame Pascale Zaraté, et Monsieur Bruno Traverson, qui me font l'honneur de discuter de mon travail et participer à ce jury.

Je remercie toutes ces personnes que j'ai rencontrées au cours de ces trois années de thèse, aux thésards et à l'ambiance cosmopolite qui règne au sein du LGC PSI. Je tiens spécialement à remercier Mujtaba Agha et ses pauses-déjeuner à base de chips et soda, El-Awady Attia pour son écoute, ses questions et le café, et Jesús Manuel Barragan Ferrer pour ses conseils avisés sur les méthodes et les techniques de travail.

J'aimerais exprimer toute ma gratitude envers mes parents, Luzviminda et Laurent, pour leur soutien dans la réalisation de mes études, leurs encouragements et leurs conseils avisés dans tous mes choix, pour m'avoir transmis leur motivation et leur volonté, essentielles à cette aventure. « Maraming salamat po » !

Enfin, j'aimerais remercier tout particulièrement Sylvie, pour ton indéfectible présence et affection. Tu as réussi à me supporter pendant ces dernières étapes difficiles de la réalisation de cette thèse. Je t'en serai éternellement reconnaissant.



# Sommaire

<b>A. PREAMBULE .....</b>	<b>15</b>
<b>1. INTRODUCTION .....</b>	<b>17</b>
1.1 CADRE DES TRAVAUX DE RECHERCHE .....	18
1.2 PRESENTATION DU MANUSCRIT .....	19
<b>2. CONTEXTE ET PROBLEMATIQUE .....</b>	<b>23</b>
2.1 CONTEXTE .....	24
2.2 PROBLEMATIQUE .....	27
2.3 OBJECTIFS DE RECHERCHE .....	29
<b>B. CADRE METHODOLOGIQUE.....</b>	<b>31</b>
<b>3. DE L'ENTREPRISE AU PROCESSUS.....</b>	<b>33</b>
3.1 ENTREPRISE .....	34
3.2 SYSTEME D'INFORMATION.....	36
3.3 MODELISATION D'ENTREPRISE .....	38
3.4 CADRE DE MODELISATION D'ENTREPRISE .....	39
3.5 CONCLUSION.....	42
<b>4. NOTIONS AUTOUR DU TERME « PROCESSUS » .....</b>	<b>43</b>
4.1 PROCESSUS.....	44
4.2 PROCESSUS MÉTIER.....	45
4.3 CYCLE DE VIE ET ACTEURS .....	49
4.4 MODELISATION PAR LES PROCESSUS : IMPORTANCE DE LA VUE FONCTIONNELLE .....	50
4.5 CONCLUSION.....	51
<b>5. INGENIERIE ET ARCHITECTURE DIRIGÉES PAR LES MODELES .....</b>	<b>53</b>
5.1 INGENIERIE DIRIGÉE PAR LES MODELES .....	54
5.2 ARCHITECTURE DIRIGÉE PAR LES MODELES.....	55
5.3 CONCLUSION.....	59
<b>6. INGENIERIE DES PROCESSUS METIER.....</b>	<b>61</b>
6.1 DE LA GESTION DU WORKFLOW A LA GESTION DES PROCESSUS METIER.....	62
6.2 LE CYCLE DE VIE DU PROCESSUS SELON L'APPROCHE BPM .....	64
6.3 DU MDA AU BPM .....	67
6.4 SUITES BPM.....	68
6.5 LES LIMITES DU BPM .....	72

6.6	CONCLUSION.....	72
-----	-----------------	----

<b>C. DEFINITION DE L'APPROCHE.....</b>	<b>75</b>
---	-----------

<b>7. DE LA NECESSITE DE L'APPROCHE .....</b>	<b>77</b>	
7.1	VERS UN ALIGNEMENT OPERATIONNEL .....	78
7.2	HETEROGENEITE ET DIFFERENTES ABSTRACTIONS.....	81
7.3	CONCEPTS ET APPROCHES POUR UNE GESTION AGILE.....	82
7.4	CONCLUSION.....	86
<b>8. CARACTERISATION .....</b>	<b>89</b>	
8.1	VERS UNE APPROCHE CENTREE PIVOT.....	90
8.2	VUES.....	91
8.3	GENERICITE .....	92
8.4	ACTEURS.....	92
8.5	DONNEES.....	92
8.6	CONCLUSION.....	96
<b>9. CONCEPTION .....</b>	<b>97</b>	
9.1	CONFORMITE ENTRE MODELE ET METAMODELE.....	98
9.2	DE LA TRANSFORMATION DE MODELES BIDIRECTIONNELLES A LA NOTION DE PIVOT .....	100
9.3	TRANSFORMATIONS HORIZONTALES, TRANSFORMATIONS VERTICALES ET INTEROPERABILITE.....	103
9.4	METAMODELE ET METAMODELES PIVOT .....	104
9.5	DEFINITION DU METAMODELE PIVOT.....	107
9.6	CARACTERISATION DES LIENS SEMANTIQUES .....	115
9.7	FORMATION DES METAMODELES BPA ET BPI.....	116
9.8	CONCLUSION.....	117

<b>D. MISE EN ŒUVRE.....</b>	<b>119</b>
------------------------------	------------

<b>10. PLATEFORME SOLUTION POUR UNE COHERENCE ET UN ALIGNEMENT DES PROCESSUS - SCALP .....</b>	<b>121</b>	
10.1	ARCHITECTURE GENERALE DE LA PLATEFORME SCALP .....	122
10.2	ENVIRONNEMENT PIVOT .....	122
10.3	ENVIRONNEMENT DE MODELISATION.....	125
10.4	ENVIRONNEMENT D'IMPLEMENTATION .....	126
10.5	IMPLEMENTATION DES MAPPINGS ET TRANSFORMATIONS .....	129
10.6	CONCLUSION.....	134
<b>11. DEMONSTRATION .....</b>	<b>137</b>	
11.1	PRESENTATION DU PROCESSUS ETUDIE .....	139

11.2	SCENARIO DE VALIDATION.....	139
11.3	PREMIERE PHASE : DU DIAGRAMME DE PROCESSUS AU MODULE ERP.....	143
11.4	DEUXIEME PHASE : DU MODULE ERP AU DIAGRAMME DE PROCESSUS.....	149
11.5	TROISIEME PHASE : DU DIAGRAMME DE PROCESSUS AU MODULE ERP.....	154
11.6	CONCLUSION.....	155
<b>12.</b>	<b>INGENIERIE DES PROCESSUS AU SERVICE DE L'INGENIERIE DES PROCESSES.....</b>	<b>157</b>
12.1	NOTIONS AUTOUR DE L'INGENIERIE DES PROCESSES.....	158
12.2	VERS UNE APPROCHE PROCESSUS-PROCEDE.....	159
12.3	CONCLUSION.....	163

<b>E.</b>	<b>EPILOGUE.....</b>	<b>165</b>
-----------	----------------------	------------

<b>13.</b>	<b>CONCLUSION ET PERSPECTIVES.....</b>	<b>167</b>
13.1	CONCLUSION.....	167
13.2	BILAN.....	168
13.3	SYNTHESE DES CONTRIBUTIONS.....	170
13.4	LIMITES ET PERSPECTIVES.....	171
<b>14.</b>	<b>BIBLIOGRAPHIE.....</b>	<b>175</b>
<b>15.</b>	<b>ANNEXES.....</b>	<b>185</b>
15.1	GLOSSAIRE.....	186
15.2	TRADUCTION DES TERMES ANGLOPHONES.....	190
15.3	ACRONYMES.....	192
15.4	DEMONSTRATION : EXTRAITS DE FICHIERS.....	194
15.5	SOMMAIRE DETAILLE.....	203



# Table des figures

FIGURE 1. PRINCIPAUX THEMES DE RECHERCHES .....	18
FIGURE 2. CONTEXTE SCIENTIFIQUE DES TRAVAUX DE RECHERCHE.....	19
FIGURE 3. CARICATURE DE LA CONCEPTION D'UN SYSTEME .....	19
FIGURE 4. PRESENTATION D'UNE PARTIE .....	20
FIGURE 5. STRATEGIC ALIGNMENT MODEL (SAM) SELON (HENDERSON AND VENKATRAMAN 1993) .....	25
FIGURE 6. PROCESSUS DE TRANSFORMATION D'UN MODELE METIER VERS UN MODELE TIC.....	28
FIGURE 7. L'ENTREPRISE PERÇUE COMME UN SYSTEME DE SYSTEMES SELON (LE MOIGNE 1984) .....	35
FIGURE 8. SI « BOITE A OUTILS » .....	36
FIGURE 9. VISION SYSTEMIQUE SIMPLIFIEE D'UN SI (BLOCH AND KROB 2005) .....	37
FIGURE 10. SYSTEME D'INFORMATION ET SYSTEME INFORMATIQUE .....	37
FIGURE 11. VUES D'ENTREPRISES (VERNADAT 1999) .....	42
FIGURE 12. PROCESSUS D'ENTREPRISE (MORLEY ET AL. 2007) .....	44
FIGURE 13. TYPOLOGIE DES PROCESSUS.....	45
FIGURE 14. METAMODELE DU PROCESSUS METIER SELON (MORLEY, HUGUES, LEBLANC, & HUGUES 2007) .....	46
FIGURE 15. EXEMPLE DE PROCESSUS METIER .....	47
FIGURE 16. PROCESSUS METIER, PROCESSUS SI, PROCESSUS INFORMATIQUE.....	47
FIGURE 17. NIVEAUX D'ABSTRACTION DU PROCESSUS METIER .....	48
FIGURE 18. CYCLE DE VIE DU PROCESSUS METIER, INSPIRE DE (ZUR MUEHLEN 2004) .....	50
FIGURE 19. SYSTEME REEL, MODELE, METAMODELE .....	54
FIGURE 20. LES MODELES MDA SELON L'APPROCHE 2TUP.....	55
FIGURE 21. MDA ADAPTE DE MODEL-DRIVEN.ORG .....	56
FIGURE 22. DOMAINE METIER, DOMAINE SI ET MDA.....	57
FIGURE 23. COUCHES DE META-MODELISATION SELON MOF.....	58
FIGURE 24. MDA, FRACTURE ENTRE DOMAINE METIER ET DOMAINE SITI .....	59
FIGURE 25. GESTION WORKFLOW ET BPM.....	63
FIGURE 26. CYCLE D'INGENIERIE CONTINUE DES PROCESSUS, BPM (DEBAUCHE AND MEGARD 2004) .....	64
FIGURE 27. MDA ET BPM .....	67
FIGURE 28. D'UNE ARCHITECTURE N-TIERS A UNE ARCHITECTURE N-TIERS DIRIGEE PAR LES PROCESSUS .....	68
FIGURE 29. EVOLUTION DU MARCHE BPMS.....	69
FIGURE 30. STRUCTURE BPMS.....	70
FIGURE 31. TRANSFORMATION D'UN MODELE BPMN A UN MODELE BPEL.....	79
FIGURE 32. DOMAINES, ACTEURS, MODELES.....	81
FIGURE 33. EVITER LE COUPLAGE FORT DU BPMS .....	84
FIGURE 34. TRANSFORMATION DE MODELES SELON IDM .....	85
FIGURE 35. UNE « BONNE ARCHITECTURE BPM » (HAVEY AND HAVEY 2005).....	86
FIGURE 36. MODELES D'ANALYSE ET ESPACES DES MODELES D'ENTREPRISES (D'APRES (TOUZI 2007) .....	91
FIGURE 37. ROLES ET DONNEES.....	94

FIGURE 38. DE L'ANALYSE A L'IMPLEMENTATION A L'AIDE D'UN PIVOT .....	95
FIGURE 39. RELATION REPRESENTATION DE .....	98
FIGURE 40. RELATION EST CONFORME A .....	98
FIGURE 41. RELATION ENTRE $M_{BPA}$ ET $MM_{BPA}$ .....	100
FIGURE 42. RELATION ENTRE $M_{BPI}$ ET $MM_{BPI}$ .....	100
FIGURE 43. MODELES, METAMODELES, METAMODELE PIVOT .....	103
FIGURE 44. ENVIRONNEMENTS DE MODELISATION ET TRANSFORMATIONS HORIZONTALES .....	104
FIGURE 45. CLASSIFICATION DES STANDARDS LIES A BPM SELON (HOLLINGSWORTH 2004) .....	105
FIGURE 46. METAMODELE PIVOT .....	110
FIGURE 47. MODELE BPA ET MODELE BPI .....	111
FIGURE 48. EXTRAIT DU MODELE PIVOT .....	112
FIGURE 49. PROPRIETES GRAPHIQUES DE L'ELEMENT « TASK A » .....	112
FIGURE 50. FORMATION D'UN $MM_i$ .....	117
FIGURE 51. PLATEFORME SCALP .....	123
FIGURE 52. INTERFACE EMF .....	123
FIGURE 53. METAMODELE ECORE .....	124
FIGURE 54. DIAGRAMME INTALIO DESIGNER .....	125
FIGURE 55. EXTRAIT DU FICHIER D'ENTREE « DESCRIPTION LOGIQUE » .....	125
FIGURE 56. EXTRAIT DU FICHIER D'ENTREE « ATTRIBUTS GRAPHIQUES » .....	126
FIGURE 57. FICHIERS DE SORTIE POUR LE MODULE OPENERP .....	127
FIGURE 58. ARCHITECTURE DU FRAMEWORK LOGICIEL .....	128
FIGURE 59. METAMODELE BPA .....	130
FIGURE 60. METAMODELE BPI .....	131
FIGURE 61. DIAGRAMME DE CLASSE DU PATTERN VISITEUR SELON (GAMMA ET AL. 1999) .....	132
FIGURE 62. PATTERN DE CONCEPTION VISITEUR, EN LANGAGE KERMETA .....	133
FIGURE 63. PROCESSUS DE REINTEGRATION DE PARFUMS .....	138
FIGURE 64. SCENARIO DE VALIDATION .....	142
FIGURE 65. DEROULEMENT DE LA PREMIERE PHASE .....	143
FIGURE 66. DU DIAGRAMME DE PROCESSUS AU MODELE $M_{BPA}$ .....	143
FIGURE 67. EXTRAIT DU FICHIER « LOGIQUE » .....	144
FIGURE 68. EXTRAIT DU FICHIER « GRAPHIQUE » .....	144
FIGURE 69. EXTRAIT DE $M_{BPA}$ .....	144
FIGURE 70. DU $M_{BPA}$ AU $M_{PIVOT}$ .....	145
FIGURE 71. EXEMPLE DE $M_{PIVOT}$ .....	146
FIGURE 72. DU $M_{PIVOT}$ AU $M_{BPI}$ .....	147
FIGURE 73. EXTRAIT DE $M_{BPI}$ .....	148
FIGURE 74. DU $M_{BPI}$ AU MODULE OPENERP .....	148
FIGURE 75. DEROULEMENT DE LA SECONDE PHASE .....	149
FIGURE 76. FONCTION PYTHON EXTRAITE DU FICHIER NOMMODULE.PY .....	149
FIGURE 77. EXEMPLE D'INTERFACE POUR LE MODULE « PROCESSUS DE REINTEGRATION » SOUS OPENERP .....	150

FIGURE 78. DU MODULE OPENERP AU LE M <sub>BPI</sub> .....	151
FIGURE 79. EXTRAIT DU M <sub>BPI</sub> MODIFIE .....	151
FIGURE 80. DU M <sub>BPI</sub> AU M <sub>PIVOT</sub> .....	152
FIGURE 81. DU M <sub>PIVOT</sub> AU M <sub>BPA</sub> .....	152
FIGURE 82. DU M <sub>BPA</sub> AU DIAGRAMME DE PROCESSUS .....	153
FIGURE 83. EXTRAIT DU FICHIER « GRAPHIQUE » OBTENU .....	153
FIGURE 84. EXTRAIT DU DIAGRAMME DE PROCESSUS OBTENU .....	154
FIGURE 85. CONCEPTS-CLES D'ONTOCAPE POUR LA MODELISATION DE PROCEDE (YANG ET AL. 2008) .....	159
FIGURE 86. DECOUPAGE DU PSE (BELAUD 2002) ET INFLUENCE POTENTIELLE DU BPA SUR LE CYCLE DE VIE DU PROCEDE.....	160
FIGURE 87. BPD, BFD, PFS A PARTIR DU « CHEMICAL SUPPLY-CHAIN » (MARQUARDT ET AL. 1999) .....	161
FIGURE 88. CARDINALITES ENTRE ENTREPRISE, PROCESSUS, SOUS-PROCESSUS ET PROCEDE .....	162
FIGURE 89. APPROCHE MULTI-EHELLE PROCESSUS-PROCEDE (SOUS FORMAT BPMN).....	162
FIGURE 90. EQUIVALENCES SEMANTIQUES POSSIBLES ENTRE ONTOLOGIE DE DOMAINE ET REGLES METIER ? .....	163
FIGURE 91. APPROCHE PIVOT POUR LA GENERATION D'UN PFS A PARTIR D'UN BPD.....	164
FIGURE 92. EXTRAIT DU FICHIER M <sub>BPA</sub> .....	194
FIGURE 93. EXTRAIT DU FICHIER M <sub>BPI</sub> .....	195
FIGURE 94. FICHIER __INIT__.PY .....	196
FIGURE 95. FICHIER __TERP__.PY .....	196
FIGURE 96. EXTRAIT DU FICHIER REINTEGPROCESS.PY.....	197
FIGURE 97. EXTRAIT DU FICHIER REINTEGPROCESS_WORKFLOW.XML .....	198
FIGURE 98. EXTRAIT DU FICHIER REINTEGPROCESS_VIEW.XML .....	199
FIGURE 99. DIAGRAMME INTALIO OBTENUE EN FIN DE SECONDE PHASE .....	200
FIGURE 100. DIAGRAMME INTALIO MODIFIE AU DEBUT DE LA TROISIEME PHASE .....	201

# Table des tableaux

TABLEAU 1. PRINCIPALES APPROCHES DE MODELISATION D'ENTREPRISE .....	39
TABLEAU 2. NOTATIONS .....	99
TABLEAU 3. EVOLUTION DE LA STANDARDISATION DU WORKFLOW .....	104
TABLEAU 4. RECAPITULATIFS DES STANDARDS .....	105
TABLEAU 5. FREQUENCE D'OCCURRENCE DES CONSTRUCTS BPMN .....	108
TABLEAU 6. ELEMENTS CONSTITUANT LE METAMODELE PIVOT .....	109
TABLEAU 7. ATTRIBUTS COMMUNS A TOUS LES ELEMENTS .....	109
TABLEAU 8. ATTRIBUT DE L'ELEMENT PROCESS .....	111
TABLEAU 9. ATTRIBUT DE L'ELEMENT BPELEMENT .....	112
TABLEAU 10. ATTRIBUT DE L'ELEMENT EVENT .....	113
TABLEAU 11. ATTRIBUT DE L'ELEMENT TASK .....	113
TABLEAU 12. ATTRIBUT DE L'ELEMENT ACTIVITY .....	113
TABLEAU 13. ATTRIBUT DE L'ELEMENT SUBPROCESS .....	113
TABLEAU 14. ATTRIBUT DE L'ELEMENT LOGICAL .....	113
TABLEAU 15. ATTRIBUT DE L'ELEMENT EDGE .....	114
TABLEAU 16. ATTRIBUT DE L'ELEMENT POOL .....	114
TABLEAU 17. ATTRIBUT DE L'ELEMENT LANE .....	114
TABLEAU 18. EXEMPLES D'EQUIVALENCE SEMANTIQUE ENTRE BPMN, XPDL ET BPEL .....	116
TABLEAU 19. APPLICATIONS ET TECHNOLOGIES UTILISEES PAR LA PLATEFORME SCALP .....	134
TABLEAU 20. RECAPITULATIFS DU DEROULEMENT ET DES OBJECTIFS DU SCENARIO .....	140
TABLEAU 21. CORRESPONDANCE SEMANTIQUE ENTRE $MM_{BPA}$ ET $MM_{PIVOT}$ .....	145
TABLEAU 22. MAPPING DE L'ELEMENT ACTIVITY ENTRE $MM_{BPA}$ ET $MM_{PIVOT}$ .....	146
TABLEAU 23. CORRESPONDANCE SEMANTIQUE ENTRE $MM_{PIVOT}$ ET $MM_{BPI}$ .....	147
TABLEAU 24. SYNTHESE DES CONTRIBUTIONS .....	171
TABLEAU 25. NOMS ET ROLES DES FICHIERS DU MODULE OPENERP .....	196

# PREMIERE PARTIE

---

## A. PREAMBULE

1.	Introduction	.....	17
2.	Contexte et problématique	.....	23

## RESUME

---

Nous présentons dans un premier chapitre les thèmes de recherches abordés dans le manuscrit ainsi que son organisation. Dans un deuxième temps cette partie présente le contexte et la problématique dans lesquels s'inscrivent nos travaux. Nous commençons par décrire l'importance prise par les systèmes et les technologies d'information et du besoin d'alignement sous-jacent entre le domaine métier d'une organisation et son domaine technologique lié aux systèmes d'information. Puis nous décrivons notre problématique d'alignement opérationnel entre les processus métier et les systèmes et technologies d'information d'une entreprise. Enfin, cette partie se conclut par les objectifs de recherche permettant la réalisation d'un tel alignement.

---

---

## Introduction

---

Nous décrivons le cadre dans lesquels se sont réalisés nos travaux. Puis nous donnons une rapide présentation des éléments inclus dans ce manuscrit, ainsi que la manière dont ils s'enchaînent.

## 1.1 CADRE DES TRAVAUX DE RECHERCHE

A travers ce manuscrit de thèse, nous souhaitons mettre en place un cadre méthodologique permettant le pilotage des processus métier selon un cycle BPM en y intégrant des concepts issus de l'IDM (Ingénierie Dirigée par les Modèles) et de l'IESI (Ingénierie de l'Entreprise et des Systèmes d'Information). Cette thèse s'inscrit dans la lignée du groupe de travail Easy-DIM<sup>1</sup> du groupe de recherche MACS<sup>2</sup> et du groupe de recherche I3<sup>3</sup>. Elle en reprend donc logiquement les thèmes principaux de recherche, à savoir les problématiques de recherche autour des modèles dans l'ingénierie d'entreprise et de système d'information au cours de leur cycle de vie, de la conception aux usages. La Figure 1, issue du livre blanc en préparation d'Easy-Dim, indique et positionne ces différents thèmes de recherches.

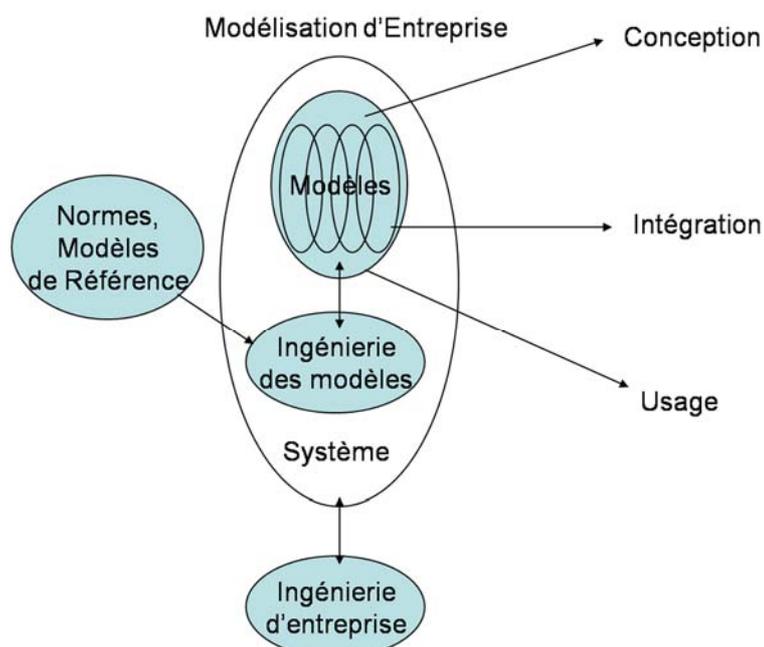


Figure 1. Principaux thèmes de recherches

Nous nous inscrivons tout naturellement dans les thématiques du génie industriel. Nous nous intéressons en particulier à déterminer l'adéquation entre le cycle de vie du processus métier et celui du procédé industriel au sens physico-chimique. La modélisation, au sens IDM, possède les mêmes objectifs que la modélisation au sens de l'ingénierie des procédés ou *Process Systems Engineering* (PSE), à savoir « l'acquisition et la capitalisation des connaissances [...] (et) le contrôle et la supervision du procédé (ou du processus) » (Truong Meyer 2009). Nous pensons qu'une concordance existe entre ces cycles de vie et notamment que l'ingénierie des processus d'entreprise peut interagir sur le cycle de vie du procédé et en permettre un meilleur pilotage. La Figure 2 récapitule l'ensemble des thématiques abordées et notre positionnement.

<sup>1</sup> <http://www.easy-dim.org>

<sup>2</sup> <http://www.univ-valenciennes.fr/GDR-MACS>

<sup>3</sup> <http://www.irit.fr/GDR-I3>

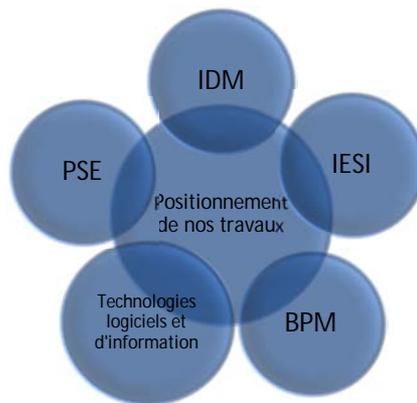


Figure 2. Contexte scientifique des travaux de recherche

## 1.2 PRESENTATION DU MANUSCRIT

### 1.2.1 Démarche proposée

De manière simpliste et humoristique, la Figure 3 représente une caricature de la conception d'un système.

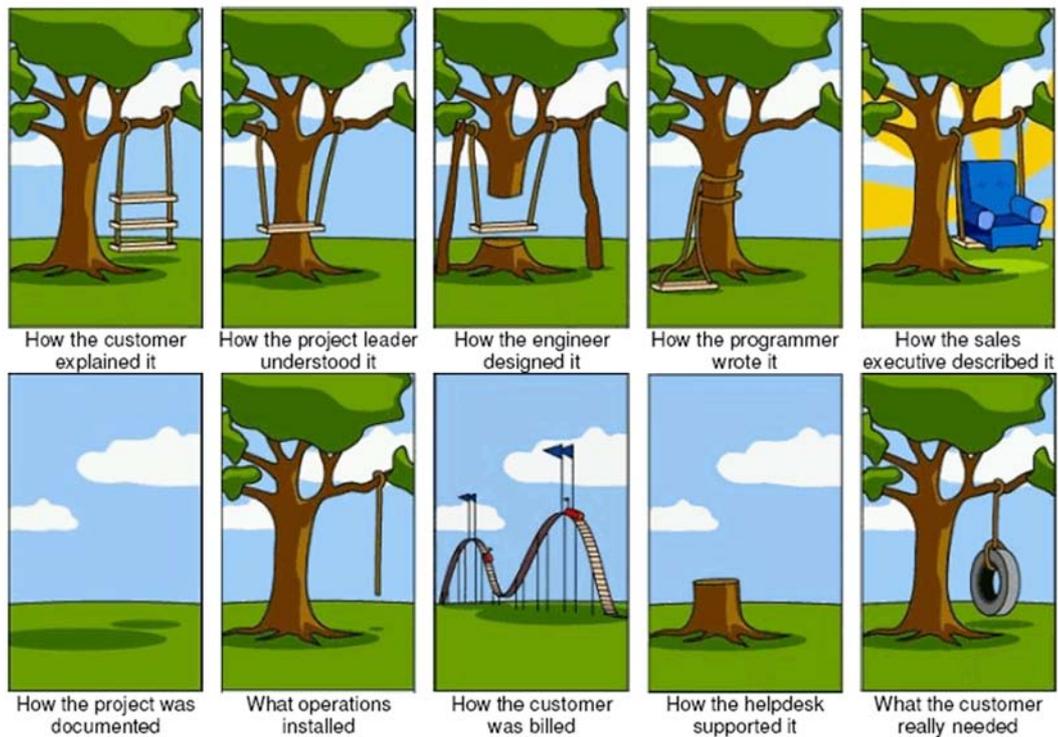


Figure 3. Caricature de la conception d'un système

Nous constatons :

- La prolifération d'acteurs, de métier différent,
- L'absence de modèle-référence, ou encore de documentation,
- Une certaine forme d'incompréhension et de lacunes de communication entre acteurs.

La démarche que nous proposons ne cherche pas à résoudre l'ensemble des problèmes mis en exergue par la Figure 3. Elle cherche, dans le cadre de la gestion des processus, à réduire les écarts constatés dans les quatre premières vignettes !

### 1.2.2 Typologie adoptée

Pour faciliter la lecture du manuscrit, plusieurs éléments sont mis en place. D'un point de vue structuration, les parties ont été rédigées « indépendamment » les unes des autres. Ceci permet de les aborder dans n'importe quel ordre, à l'exception des parties « Préambule » et « Epilogue ». Aussi, chaque partie débute par un sommaire des chapitres abordés (Figure 4- a) et un résumé (Figure 4- b) permettant de rapidement saisir le contenu traité dans la partie.

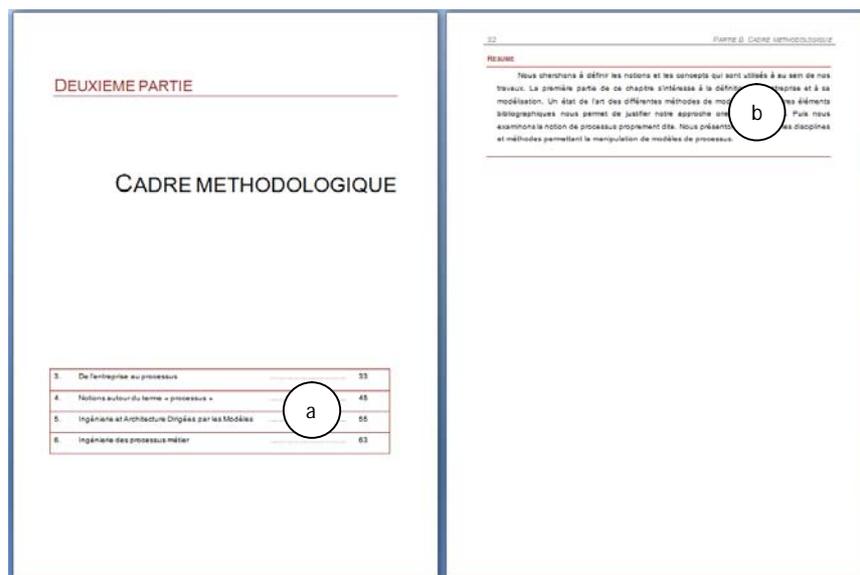


Figure 4. Présentation d'une partie

D'un point de vue contenu, chaque chapitre dispose d'éléments visuels afin d'en faciliter la lecture.

*Cette représentation* indique un terme en anglais. Nous pourrions simplement nous contenter d'utiliser le terme équivalent en français, cependant la majorité des sujets discutés dans ce manuscrit sont rencontrés le plus souvent dans la littérature sous leur forme anglo-saxonne.

*L'utilisation de cette forme* indique un terme technique, une formule mathématique.

Cette forme indique une définition présente dans le corps du manuscrit

Et un terme *indiqué de cette manière* possède une définition dans le Glossaire situé en Annexes.

De la même manière, un code couleur a été choisi et utilisé dans les différents schémas :

: représente un élément en rapport avec le domaine métier ;

: représente un élément en rapport avec le domaine SITI (Systèmes d'Information et Technologies d'Information) ;

: représente un élément en rapport avec l'environnement pivot.



*Px : Nom de la  
x-ième propriété*

Cette nomenclature indique les propriétés que la plateforme solution doit aborder ou posséder.

Le document est organisé de la manière suivante :

#### PREMIERE PARTIE : PREAMBULE

---

Nous présentons dans un premier chapitre les thèmes de recherches abordés dans le manuscrit ainsi que son organisation. Dans un deuxième temps cette partie présente le contexte et la problématique dans lesquels s'inscrivent nos travaux. Nous commençons par décrire l'importance prise par les systèmes et les technologies d'information et du besoin d'alignement entre le domaine métier d'une organisation et son domaine technologique lié aux systèmes d'information. Puis nous décrivons notre problématique d'alignement opérationnel entre les processus métier et les systèmes et technologies d'information d'une entreprise. Enfin, cette partie se conclut par les objectifs de recherche permettant la réalisation d'un tel alignement.

#### DEUXIEME PARTIE : CADRE METHODOLOGIQUE

---

Nous cherchons à définir les notions et concepts qui sont utilisés au sein de nos travaux. La première partie de ce chapitre s'intéresse à la définition de l'entreprise et à sa modélisation. Un état de l'art des différentes méthodes de modélisation nous permet de justifier notre approche orientée processus. Puis nous examinons la notion de processus proprement dite. Nous présentons également les disciplines et méthodes permettant la manipulation de modèles de processus.

#### TROISIEME PARTIE : DEFINITION DE L'APPROCHE

---

Nous mettons en avant les dysfonctionnements de l'ingénierie des processus métier lors de son application. Après un court exemple utilisant les standards les plus répandus du BPM, nous discutons des limites d'une démarche BPM. De ces limites, nous exposons les solutions qu'apporte notre approche pivot. Les chapitres suivants définissent le champ d'application de notre approche ainsi que ses caractéristiques. Nous insistons en particulier sur les liens existants entre modèles et métamodèles, ainsi que sur la formation et la constitution du métamodèle pivot.

#### QUATRIEME PARTIE : MISE EN ŒUVRE

---

Les différents éléments constituant la plateforme solution expérimentant notre approche sont développés. Puis un cas concret d'utilisation de notre approche sera présenté. Un processus réel et utilisé en industrie sera manipulé depuis sa modélisation à sa transformation en module fonctionnel pour un logiciel de type ERP. Nous présentons aussi nos travaux de recherche relatifs à la possible adéquation entre l'ingénierie des processus et l'ingénierie des procédés. Nous pensons qu'une telle adéquation est possible à travers l'utilisation de notre approche pivot, et nous cherchons à le démontrer.

## CINQUIEME PARTIE : EPILOGUE

---

Cette dernière partie contient une conclusion générale portant sur l'ensemble du manuscrit. Cette partie contient également les annexes et la bibliographie.

---

## Contexte et problématique

---

Ce chapitre expose le contexte scientifique dans lequel s'inscrivent nos travaux. En premier lieu, nous cherchons à caractériser et à définir la notion d'alignement, en particulier l'alignement entre processus d'entreprise et systèmes d'information. Notre objectif, à travers ce chapitre, est d'identifier l'importance graduelle qu'ont pris les systèmes d'information au sein d'une organisation.

Puis nous développons la problématique abordée et les questions qui y sont rattachées. Pour cela, nous nous intéressons à l'alignement entre modèles de processus métier et modèles d'implémentation destinés aux systèmes d'information. De fait, nous cherchons à définir précisément la notion d'alignement entre ces deux entités. Puis nous démontrons la difficulté à mettre en place un tel alignement et de l'écart entre le domaine « métier » et le domaine « technologies des systèmes d'information » qui en découle.

## 2.1 CONTEXTE

Afin de demeurer compétitives et d'être en mesure de répondre aux challenges du marché actuel, les organisations modernes se doivent d'être agiles, autrement dit, que ce soit d'un point de vue inter ou intra-organisationnel, les différents systèmes constituant l'organisation doivent pouvoir intervenir rapidement sur la chaîne de valeur. Une entreprise doit également être capable d'interagir de manière optimale avec l'ensemble des acteurs la constituant, qu'ils appartiennent au domaine « métier » de l'entreprise ou au domaine des Systèmes d'Information (SI). Dans la suite du manuscrit, nous constatons la forte dépendance existante entre l'entreprise et son système d'information. Puis nous montrons à travers nos travaux l'importance de l'alignement de l'entreprise avec ses systèmes d'information et exposons l'intérêt de percevoir l'ensemble des éléments composant une entreprise comme un tout cohérent.

### 2.1.1 *Naissance et importance des stratégies SITI et métier*

Depuis 1960, les Technologies de l'Information (TI) et les systèmes d'information se sont caractérisés par un développement rapide, et ont émergé dans le domaine métier d'une organisation jusqu'à en devenir des éléments essentiels. La majorité des organisations, que ce soit dans les secteurs de l'industrie, des affaires et du commerce, du gouvernement, ou encore de la santé sont fondamentalement tributaires de leurs systèmes et des technologies de l'information (dans la suite du manuscrit, nous utiliserons l'acronyme SITI pour les désigner). Pour que les organisations demeurent compétitives dans un environnement socio-économique dynamique, elles doivent déterminer et comprendre comment gérer efficacement et de façon stratégique le domaine SITI. Posséder un système d'information efficace et efficient supportant les stratégies métier et les processus qui y sont rattachés est rapidement devenu un facteur clé de succès (Henderson and Venkatraman 1999). Ceci permet un rendement organisationnel efficace qui apporte une contribution à la création de valeur, et ce d'une manière reconnue.

### 2.1.2 *Besoin d'un alignement entre stratégie métier et stratégie SITI*

C'est pourquoi le besoin d'aligner le SI à la stratégie d'entreprise est devenu en quelques années la priorité absolue des directeurs des systèmes d'information (Luftman and Maclean 2004). De cet alignement résulte un accroissement de la performance d'une organisation. A l'inverse, si cet alignement n'est pas réalisé correctement, les processus d'entreprises n'utiliseront pas les ressources technologiques mises en œuvre de manière optimale. En conséquence, une organisation doit être en mesure de définir des processus d'entreprise répondant aux exigences métier de ses décideurs, ainsi que de construire des SI performants. Ces systèmes sont jugés performants s'ils prennent correctement en charge les processus, permettant le bon fonctionnement et une augmentation de la performance de l'organisation.

Les SITI sont ainsi devenus une partie intégrante de la plupart des formes de structure public et socio-économique (entreprise commerciale ou individuelle, association, collectivités, administration,...)<sup>4</sup>. Les SI dépassent le traditionnel rôle de *back-office* qui leur était attribués et acquièrent une valeur stratégique à prendre en considération au sein de la stratégie de l'entreprise. Un alignement explicite entre domaine métier et SITI devient nécessaire afin de réagir efficacement et avec agilité face aux événements exogènes ou endogènes de l'entreprise. Ces événements peuvent être une modification des besoins clients ou des objectifs de l'organisation entraînant des changements au niveau des processus d'entreprise et donc des SI. Ainsi le type d'interactions existant entre métier et SITI ne se restreint pas au seul niveau stratégique, il couvre également le niveau opérationnel (entre processus d'entreprise et SI) comme nous allons le voir.

### 2.1.3 Alignement stratégique, alignement opérationnel

Un *alignement stratégique* améliore la performance organisationnelle à travers divers mécanismes comme les processus de pilotage, les ressources humaines et les capacités technologiques, où ces capacités peuvent être interprétées essentiellement comme la capacité à déployer des ressources (Amit and Schoemaker 1993). Ce concept d'alignement a été présenté par (Henderson & Venkatraman 1993) qui ont proposé le modèle d'alignement stratégique *Strategic Alignment Model, SAM* représenté Figure 5.

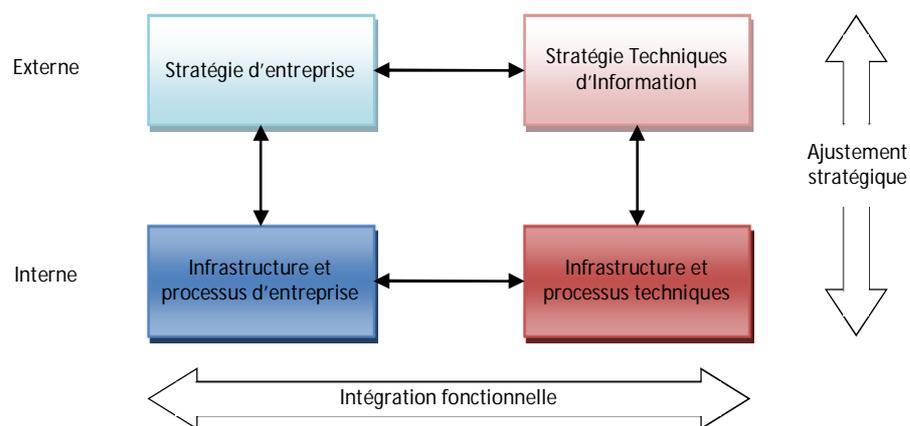


Figure 5. *Strategic Alignment Model (SAM)* selon (Henderson and Venkatraman 1993)

Le SAM fait une distinction entre un *domaine externe* et un *domaine interne*. Le domaine externe porte sur la formulation des stratégies reflétant l'environnement dans lequel s'inscrit l'entreprise. Le domaine interne se préoccupe de l'infrastructure SITI et des processus d'entreprise. *L'ajustement stratégique* décrit la relation verticale existant entre un domaine externe et le domaine interne qui lui est associé. *L'intégration fonctionnelle* représente la relation horizontale existant entre deux domaines. Cette dernière relation peut également être perçue comme un alignement opérationnel (Wagner et al. 2006).

<sup>4</sup> Nous utiliserons maintenant le terme « entreprise » pour désigner ces différents types de structures.

De nombreuses contributions soulèvent l'importance d'un alignement stratégique SITI-métier (Broadbent and Weill 1993), (Luftman and Brier 1999), (Avila Cifuentes 2009), (Cuenca et al. 2010). L'alignement stratégique SITI-métier peut être conceptualisé à l'aide du modèle SAM sous la forme de deux actions (ou « briques ») : un ajustement plus un alignement. Cet enchaînement est défini comme étant une séquence d'alignement.

A l'opposé, les implications directes pour l'entreprise quant à la relation entre les SI et les unités métier sont beaucoup moins représentées dans la littérature (Gordon and Gordon 2000). Ainsi, peu d'attention a été accordée aux relations pouvant exister entre les SI (infrastructures et processus techniques) et les unités métier (infrastructures et processus d'entreprise) qui utilisent ces systèmes d'information.

#### 2.1.4 *Emergence d'un nouveau domaine de l'ingénierie des SI*

L'alignement entre la stratégie d'entreprise et les choix de déploiement des SITI sont devenus un point important de préoccupation pour les entreprises (Silvius 2007), (Ward and Peppard 2002). Ceci est particulièrement le cas lorsque les SITI deviennent une partie essentielle de l'entreprise et sont utilisés pour exploiter des compétences métier spécifiques, mis à contribution lors de fusions entre entreprises, lors de restructuration... (Earl 1992), (Chan et al. 1997), (Luftman & Maclean 2004) insistent sur l'importance prise par l'alignement entre la stratégie de l'entreprise et les SITI, qui en 20 ans est devenu un enjeu majeur de gestion. Si l'intérêt de réaliser un tel alignement est reconnu, sa mise en œuvre reste trop souvent limitée. Il n'est pas rare que les acteurs de l'organisation ne savent pas en quoi consiste cet alignement, de même qu'il existe une absence de communication entre domaine métier et domaine SITI.

De ce fait, les entreprises et les experts SITI sont constamment à la recherche de meilleures pratiques de gestion les aidants à aligner leurs stratégies métier avec les SITI. Nous avons vu que de nombreux travaux de recherche s'intéressent et étudient cet alignement. Nous pouvons également signaler que plusieurs ateliers lui ont été consacrés (BPMDs, 2004<sup>5</sup>, REBNITA, 2005<sup>6</sup>, MOSIM, 2010<sup>7</sup>). A travers ces différents articles et/ou présentations, les auteurs mettent en avant des méthodes et des outils propres à l'alignement, par exemple pour construire un système et des processus d'entreprise alignés, ou pour évaluer l'évolution de la relation d'alignement au cours du temps (Avila Cifuentes 2009).

Tout ceci participe à l'émergence d'un domaine de l'ingénierie des SI : celui de l'ingénierie de l'alignement. (Etien 2006) observe que l'ingénierie de l'alignement peut être considérée comme un sous-domaine de l'ingénierie des systèmes d'information, au même titre que l'ingénierie des méthodes ou de l'ingénierie des exigences. L'ingénierie de l'alignement devrait ainsi être considérée comme une discipline à part entière, de part l'importance que revêt l'alignement pour les chercheurs, les entreprises et leurs acteurs.

---

<sup>5</sup> Fifth Workshop on Business Process Modeling, Development, and Support, CAiSE 2004

<sup>6</sup> 1st International Workshop on Requirements Engineering for Business Need and IT Alignment, 13th IEEE International Conference on Requirements Engineering, 2005.

<sup>7</sup> 8ème ENIM IFAC Conférence Internationale de Modélisation et Simulation, 2010

Ce manuscrit contribue à la littérature en apportant des solutions pour un meilleur alignement opérationnel.

## 2.2 PROBLEMATIQUE

Pouvoir saisir de nouvelles opportunités nécessite de la *flexibilité* au niveau du métier d'une organisation. Cette flexibilité se traduit par l'utilisation de processus d'entreprise supportés par les SI. Cette flexibilité de gestion permet d'obtenir l'*agilité* d'entreprise souhaitée. L'agilité des SI est présentée en détail par (Boucher 2007). Cependant nous observons une discontinuité franche entre domaine métier et domaine SITI. Ainsi, au lieu de permettre et d'accompagner le changement des processus, les SI peuvent brider ou ralentir la mise en place de telles évolutions. Notre présent travail est motivé par la réalisation d'un alignement entre domaine métier et domaine SITI selon une démarche orientée processus permettant de résoudre ces différents points. Néanmoins, ils subsistent plusieurs difficultés inhérentes à l'utilisation d'une telle approche.

### REMARQUE : AGILITE OU FLEXIBILITE

---

Les concepts d'agilité et de flexibilité peuvent paraître très proches en théorie. Cependant, ces deux termes présentent des différences notables. La flexibilité est semblable à la notion de réactivité industrielle, à savoir la capacité d'une entreprise à répondre rapidement aux besoins de ses clients par une attribution différente de ses ressources. L'agilité définit la capacité d'une entreprise à s'adapter rapidement à son environnement, permettant de procéder à des changements harmonieux et continus de l'organisation.

---

#### 2.2.1 *Alignement opérationnel, intégration fonctionnelle, cohérence, ...*

(Avison et al. 2004) estime que la notion d'alignement demeure mal définie, tant entre stratégies d'entreprise et stratégies des SITI qu'au niveau opérationnel, entre SI et processus d'entreprise. Cette absence formelle de définition nous amène à nous poser des questions sur ses caractéristiques et ses objectifs.

A travers la solution proposée, nous considérons que l'alignement opérationnel permet de réduire l'écart reconnu entre domaine métier et SITI. Dès lors, l'alignement est-il considéré comme un état à atteindre ou un processus à mettre en œuvre ? De même, est-il erroné d'associer la définition de l'alignement opérationnel à celle de l'intégration fonctionnelle ?

#### 2.2.2 *Cycles de vie d'applications et de technologies asynchrones*

Les applications et technologies des domaines métier et SITI évoluent différemment au cours du temps.

Les besoins métier d'une organisation s'illustrent par leur constante évolution. Leurs évolutions sont dues à un marché changeant, aux acquisitions et fusions entre entreprises, à l'application de nouvelles stratégies et aux priorités dictées par les investisseurs. Les nécessités métier liées au changement forment un processus totalement imprévisible. Les produits associés aux technologies de

l'information (par exemple les services Web ou J2EE) sont mis à jour une à deux fois par an. Nous pouvons observer des changements de tendances majeures tous les deux à cinq ans.

Ainsi, les cycles de vie sont différents pour les besoins métier, les SITI, les projets et les produits qui y sont associés. Cette désynchronisation crée une discontinuité entre le processus souhaité (*as-wish*, processus d'entreprise) et le processus implémenté par le système d'information (*as-is*, processus technique) les mettant en œuvre.

### 2.2.3 De la difficulté à maintenir l'alignement

Maintenir un *alignement opérationnel* revient à conserver une cohérence sémantique et structurelle entre modèles hétérogènes et d'abstraction/granularité différentes. Un non-alignement se crée entre ces modèles hétérogènes lors de leurs manipulations : il s'agit de l'écart métier-SITI tel que décrit dans la littérature. Pour se rendre compte de la difficulté à réaliser ce maintien, considérons le processus de transformation depuis un modèle *métier* vers un modèle *SITI*, issus de leurs domaines respectifs. Ce processus est représenté Figure 6 (Stein et al. 2008).

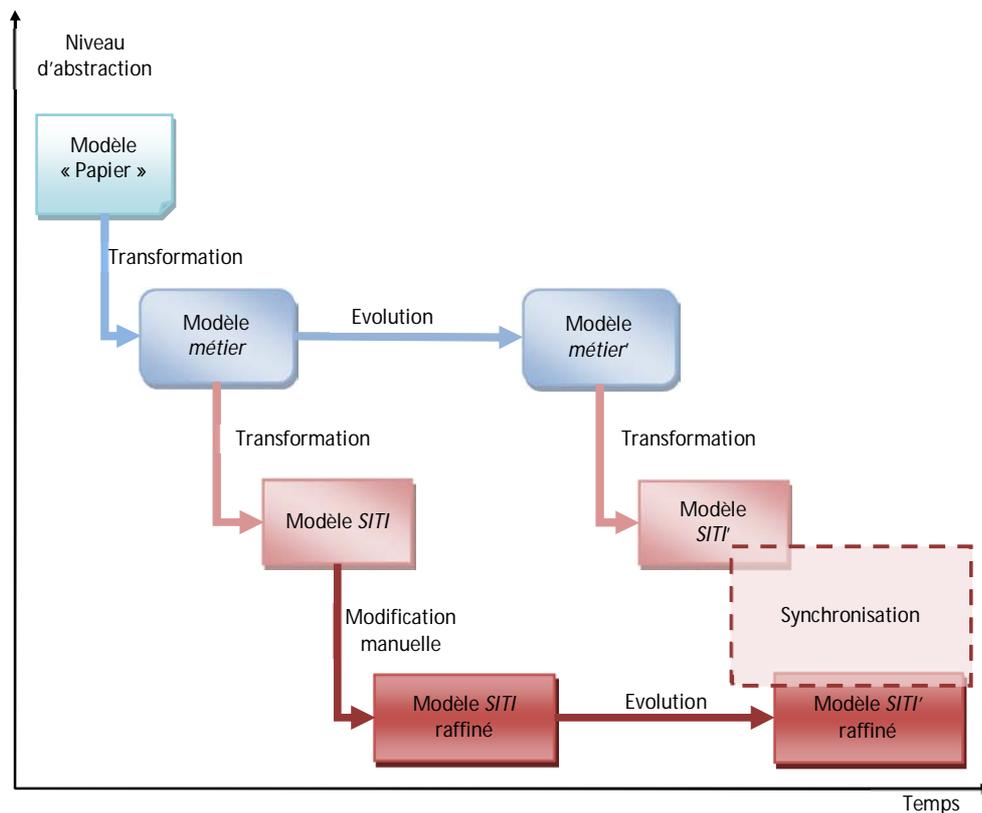


Figure 6. Processus de transformation d'un modèle métier vers un modèle TIC

Le domaine métier produit des modèles de processus non-formels et interprétables essentiellement par l'homme rendant la phase d'implémentation s'exécutant au niveau SITI plus complexe. Une première transformation depuis un modèle métier vers un modèle SITI est nécessaire. Il convient de garder à l'esprit que ces modèles sont amenés à être modifiés, les processus représentés et les technologies de l'information utilisées pouvant évoluer comme nous l'avons indiqué précédemment.

Des efforts de synchronisation, de mises à jour et de cohérence entre modèles s'avèrent alors nécessaires pour garantir leur alignement.

## 2.3 OBJECTIFS DE RECHERCHE

Afin d'améliorer les stratégies d'alignement actuelles, nos travaux suggèrent une approche générique pour la modélisation et l'implémentation des processus et de réaliser une plateforme supportant une telle approche.

### 2.3.1 Contributions

Nous proposons une méthode permettant le pilotage des processus métier, de leur modélisation à leur implémentation au sein d'un SI. A l'aide de concepts issus de l'ingénierie d'Entreprise et des Systèmes d'Informations dirigés par les modèles et des SITI, nous définissons une démarche générique assurant une cohérence intermodèle. Son rôle est de conserver et de fournir toutes les informations liées à la structure et à la sémantique des modèles.

En permettant la restitution intégrale d'un modèle transformé au sens de l'ingénierie inverse, notre plateforme permet une synchronisation entre modèle d'analyse et modèle d'implémentation issus respectivement du domaine métier et du domaine SITI. La cohérence intermodèle et la synchronisation obtenue sont les conditions nécessaires et suffisantes permettant un alignement opérationnel entre les domaines considérés.

Nous envisageons également l'adéquation possible entre l'ingénierie des procédés et l'ingénierie des processus d'entreprise à travers un point de vue multi-échelle et justifiée par notre cadre méthodologique.

### 2.3.2 Résultats principaux

Suite à nos travaux, nous proposons les principaux résultats suivants qui seront détaillés tout au long de ce manuscrit :

- Une analyse bibliographique du domaine ;
- Une approche *Business Process Management* -BPM reposant sur un élément-pivot,
- Une plateforme logicielle nommée Solution pour la Cohérence et l'Agilité des Processus (SCALP),
- Une étude de cas,
- Une ouverture vers le *Process Systems Engineering* - PSE.



## DEUXIEME PARTIE

---

# B. CADRE METHODOLOGIQUE

3.	De l'entreprise au processus	.....	33
4.	Notions autour du terme « processus »	.....	43
5.	Ingénierie et Architecture Dirigées par les Modèles	.....	53
6.	Ingénierie des processus métier	.....	61

## RESUME

---

Nous cherchons à définir les notions et les concepts qui sont utilisés au sein de nos travaux. La première partie de ce chapitre s'intéresse à la définition de l'entreprise et à sa modélisation. Un état de l'art des différentes méthodes de modélisation et autres éléments bibliographiques nous permettent de justifier notre approche orientée processus. Puis nous examinons la notion de processus proprement dite. Nous présentons également les disciplines et méthodes permettant la manipulation de modèles de processus.

---

---

## De l'entreprise au processus

---

Dans ce chapitre, nous décrivons le cheminement permettant de passer du niveau d'abstraction de l'entreprise à celui de processus. Nous définissons les différents concepts ponctuant cette approche. Les différents cadres de modélisation utilisés sont alors décrits avec notamment les vues et les caractéristiques qu'ils considèrent.

## 3.1 ENTREPRISE

### REMARQUE : ENTREPRISE ET SYNONYMES

---

Dans l'ensemble du manuscrit, le terme *entreprise* est considéré comme synonyme des termes *organisation*, *société* ou encore *compagnie*.

---

#### 3.1.1 Définition

Une entreprise est, selon (Izza 2006), le « siège d'activités les plus diverses visant à créer de la valeur et dans laquelle l'information est considérée comme une essence et une ressource vitale pour son fonctionnement ». Dans le reste du manuscrit, nous suggérons la définition suivante d'une entreprise :

Une *entreprise* est un système socio-technique motivé par un ou plusieurs buts et objectifs. Elle possède une complexité apparente à la fois structurelle et organisationnelle. Ce système est ouvert, car il possède des relations avec des entités extérieures et est dynamique, sa structure et son organisation n'étant pas figées dans le temps.

Les buts et objectifs sont atteints grâce à l'orchestration d'activités diverses réalisées à l'aide d'acteurs et des ressources de l'entreprise.

Une entreprise peut également adopter un comportement émergent, c'est-à-dire qu'il est a priori imprédictible. Nous la qualifierons comme étant un système sociotechnique complexe (Aloui 2007).

Dans le rapport de l'action spécifique PRODLOG (AS n°35 PRODLOG) du GDR MACS (LAAS-CNRS 2003), ces activités sont « mises en œuvre par des ressources sociotechniques dans le cadre d'une finalité identifiée. Ces activités peuvent s'exécuter aussi bien au sein d'un projet, d'une entreprise traditionnelle, ou d'un réseau de sociétés ». La réalisation de ces activités collectives dans un domaine particulier, car impliquant différents acteurs partageant un but commun est la définition même du métier d'une entreprise selon (Martin et al. 2004). Il convient de noter qu'une entreprise peut également être perçue comme une « collection de métier ». Nous proposons et retiendrons la définition suivante d'un métier d'entreprise :

Le *métier* d'une entreprise représente un ensemble d'activités d'un domaine donné nécessitant des compétences et savoir-faire des acteurs de l'entreprise.

A travers ces différentes définitions, nous voyons qu'une entreprise est un système complexe de systèmes. Afin d'analyser les propriétés d'une entreprise, nous devons définir ses sous-systèmes caractéristiques.

### 3.1.2 *Entreprise, un système de systèmes*

Les propriétés que nous désirons mettre en avant sont celles liées à la représentation et à la modélisation de l'entreprise en tant que système de systèmes. Nous ne nous intéressons pas aux aspects liés au marché économique, au développement de l'entreprise, aux considérations sociales et sociétales... Nous pouvons distinguer trois aspects caractérisant le « système » entreprise. L'aspect fonctionnel de l'entreprise représente l'ensemble des tâches existantes d'une entreprise (les applications d'entreprises). L'aspect comportemental définit les différents ordres disponibles d'exécution des tâches d'une entreprise, que nous nommons processus organisationnels ou processus métier. L'aspect dynamique du comportement prend en compte les ressources et le temps de l'entreprise.

Selon (Le Moigne 1984), nous avons trois sous-systèmes caractérisant le système sociotechnique complexe qu'est l'entreprise. Tout d'abord, le système opérant, qui réagit aux événements quotidiens, qui viennent de l'environnement, selon des règles définies. Il est chargé de transformer des ressources ou flux primaires (flux financiers, flux de personnel, flux de matière, flux d'information). Ensuite, le système de pilotage, qui permet d'engager le processus de décision tout en définissant au préalable les objectifs, les critères d'évaluation et les règles de gestion. Enfin, le *Système d'Information* (SI), qui relie les deux systèmes précédents tout en jouant un rôle de « coupleur » (Figure 7). Il correspond à la partie chargée de la collecte, du traitement, du stockage et la diffusion des informations. Il peut être perçu comme une représentation du système opérant et/ou du système de pilotage.

Dans les paragraphes suivants, nous nous attachons à définir la notion de ce système d'information, cœur du système de systèmes que représente l'entreprise.

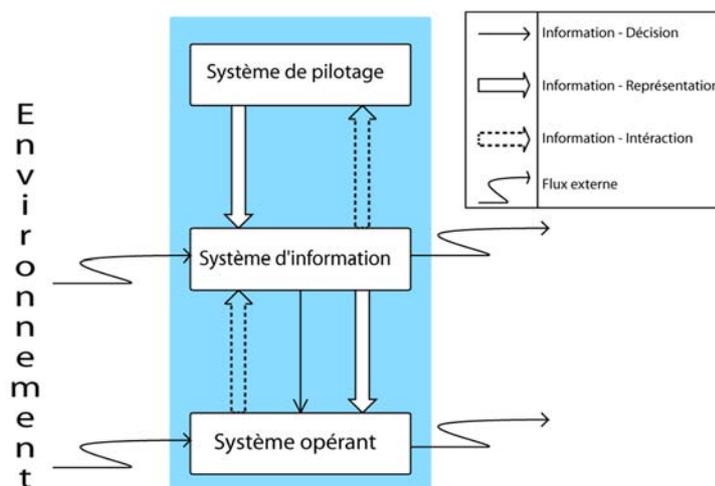


Figure 7. L'entreprise perçue comme un système de systèmes selon (Le Moigne 1984)

### 3.2 SYSTEME D'INFORMATION

Le Système d'Information (SI) est devenu le centre du fonctionnement de tout type d'entreprise. Son efficacité est un élément-clé de la performance de ces entreprises. Il est même considéré comme étant « l'objectif global recherché de la modélisation d'entreprise » (Ferchichi 2008).

Plusieurs définitions sur le système d'information existent. La définition d'(Alter 1999) définit un système d'information comme « un système de travail dont les fonctions internes sont limitées à traiter l'information, en exécutant six types d'opérations : saisir, transmettre, stocker, retrouver, manipuler, afficher l'information. Un système d'information produit de l'information, assiste ou automatise le travail exécuté par d'autres systèmes de travail. » Selon cette définition, nous pourrions presque borner l'utilisation d'un SI à celui d'un système informatique, ce qui reste très réducteur pour le SI ! Nous retrouvons cette notion d'un SI très informatisé, transformant des données en informations puis les redistribuant, n'ayant qu'une faible partie décisionnelle dans la définition de (Le Moigne 1984) vue dans le paragraphe précédent.

Au début des années 2000, la définition du SI se fait plus globale. (Reix 2004) ne fait pas de distinction entre les trois sous-systèmes d'une entreprise en considérant le SI (Figure 8) comme « un ensemble organisé de ressources : matériel, logiciel, personnel, données, procédures permettant d'acquérir, traiter, stocker, communiquer des informations (sous forme de données, textes, images, sons, etc.) dans l'organisation».

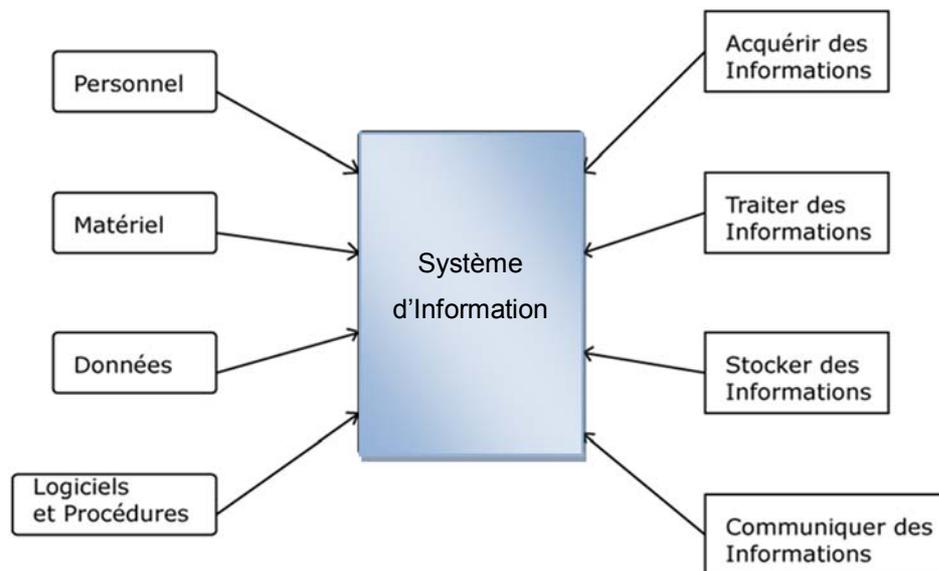


Figure 8. SI « boîte à outils »

Il est caractérisé par (Laleau 2002) :

- L'utilisation de plusieurs données,
- Des relations complexes entre les structures de données,
- Des opérations impliquant plusieurs structures de données, un important volume de données et préservant néanmoins l'intégrité des données modifiées,
- Des utilisateurs hétérogènes pouvant agir en concurrence,

- Une communication adaptée des requêtes des utilisateurs et une gestion des messages d'erreur en cas d'appel invalide des opérations.

Il faut attendre 2005 pour obtenir des définitions du SI où le souci d'organisation apparaît. Ainsi (Bloch & Krob 2005) considère non seulement le système informatique mais également l'organisation de l'entreprise comme éléments du système d'information (Figure 9). La notion de SI est en effet étroitement liée à celle d'organisation.

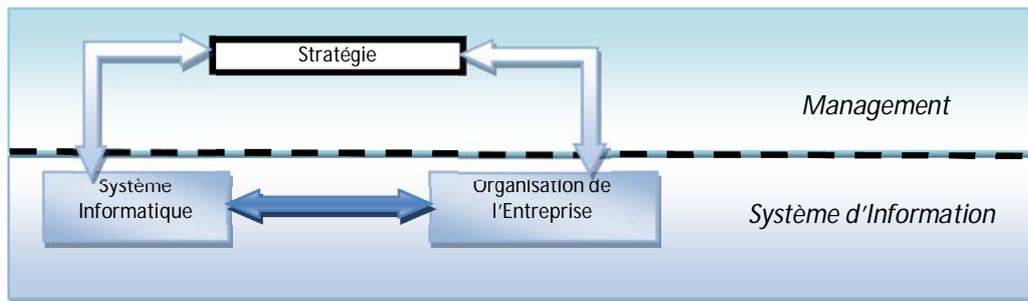


Figure 9. Vision systémique simplifiée d'un SI (Bloch and Krob 2005)

Nous remarquons aussi que (Morley, Hugues, Leblanc, & Hugues 2007) distingue clairement le SI du système informatique comme le montre la Figure 10. En considérant le SI sous la gouvernance du système de pilotage, nous retrouvons l'entreprise « système de systèmes » telle que définie par (Lemoigne 1984), où le SI supportant la gestion de l'entreprise est supporté par les ressources informatiques d'aide à la décision.

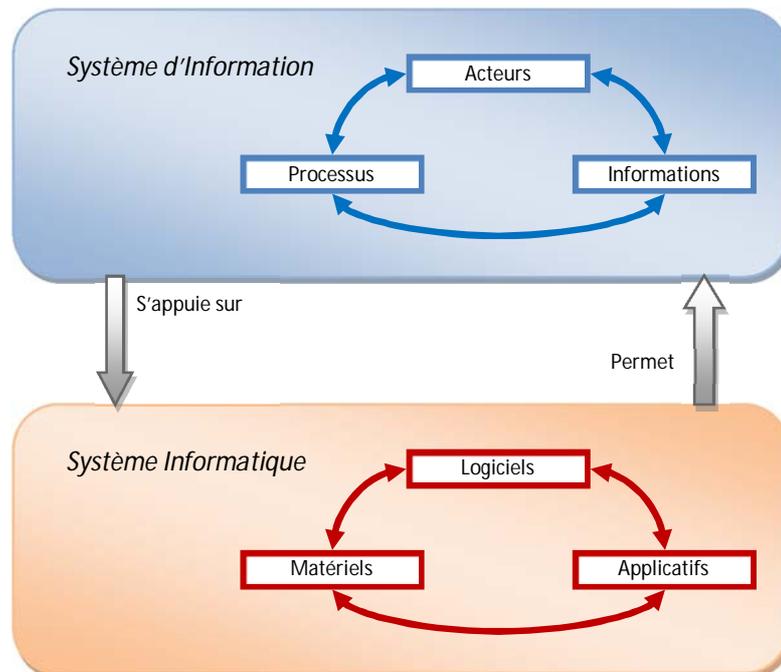


Figure 10. Système d'information et système informatique

Dans la suite du manuscrit, nous considérons le SI selon la définition de (Morley, Hugues, Leblanc, & Hugues 2007) :

Un *système d'information* est la partie réelle constituée d'informations organisées et d'acteurs qui agissent sur ces informations ou à partir de ces informations, selon des processus visant une finalité de gestion et utilisant des technologies de l'information.

Une fois ces différents systèmes reconnus, il faut modéliser ce système complexe qu'est l'entreprise. Cette représentation se fait à l'aide d'un ensemble de modèles. Nous présentons succinctement les méthodes supportant cette modélisation d'entreprise.

### 3.3 MODELISATION D'ENTREPRISE

La *modélisation d'entreprise* a pour objet la construction du modèle de tout ou partie de l'entreprise. L'entreprise est alors vue comme un système et sa modélisation doit en expliquer la structure, l'organisation et le fonctionnement (Pourcel and Gourc 2005). Le modèle réalisé doit fournir une représentation de l'architecture de l'entreprise facilitant sa compréhension. Nous considérons qu'il permet d'analyser le comportement et de situer le fonctionnement du système en cohérence avec la stratégie de l'entreprise. La modélisation d'entreprise est également perçue comme un « art d'externaliser le savoir-faire de l'entreprise (Touzi et al. 2008) », la rendant de ce fait comme un prérequis pour une démarche d'intégration d'entreprise. La conception d'un système sociotechnique complexe (numérique, analogique ou mixte) se caractérise par une forte augmentation de sa complexité alors que les cycles de développement sont de plus en plus réduits. Le recours à des formalismes et des outils devient plus que nécessaire. Les méthodes de modélisation diffèrent selon le point de vue qu'elles ciblent (opérant/information/décision) ou selon leurs objectifs (audit/ analyse/ conception).

Idéalement, une *Méthode de Modélisation d'Entreprise* (MME) doit pouvoir décrire les aspects comportementaux (notamment la gestion des événements), fonctionnels et dynamiques, décrits au début de la section 3.1.2, ainsi que la formalisation des savoirs-faire. Plusieurs définitions d'une MME existent, (Vernadat and Hamaidi 1998) considère les MME de la manière suivante :

Une MME fournit la description de l'organisation des processus d'un système soit dans le but de les simuler pour comparer divers scénarios, soit dans le but de les analyser et de les restructurer pour améliorer la performance système

Une MME doit également être en mesure de représenter la structure de pilotage ou des aspects liés à la gouvernance de l'entreprise. Toutes les méthodes de modélisation d'entreprise manipulent les quatre composants de base suivant :

- Objets (produit ou objet technique, information ou objet symbolique,...) ;
- Ressources (humaines, techniques, naturelles) ;
- Activités et processus ;
- Différents aspects dynamiques, économiques humains et sociaux.

Nous pouvons différencier ces méthodes selon le type de méthodes utilisées (Aloui 2007), (Darras 2004). Nous pouvons en distinguer deux en particulier.

Considérons en premier lieu les méthodologies se reposant sur une architecture de référence. Cette architecture est formée par les modèles d'entreprises et les méthodes de conception qu'ils soutiennent. Citons par exemple CIMOSA (AMICE 1993), ARIS (Scheer 2002), GERAM (IFAC-IFIP Task Force 1997).

Puis nous avons des méthodes appliquant elles-mêmes des méthodes dites opérationnelles. Nous pouvons citer SADT, méthode générale où un système complexe est perçu puis décomposé en plusieurs systèmes simples. La méthode GRAI (Roboam 1993) est une méthode orientée système de décision. Nous avons également des méthodes dédiées aux SI comme MERISE (Tardieu et al. 2002), orientée processus et SI comme OLYMPIOS (Théroutte et al. 2001). Nous avons enfin des méthodes opérationnelles reposant sur les langages de modélisation d'entreprise, comme UML (OMG 2007a) ou BPMN (OMG 2008).

Les approches utilisées permettent également de différencier les types de méthodes selon trois approches. (1) L'approche fonctionnelle est basée sur le principe de décomposition descendante ou approche *top-down*. Elle est modulaire et structurée. Les méthodes s'y rapportant représentent le plus souvent des diagrammes de flux de données centrés sur les fonctions du système. (2) L'approche systémique se base sur l'interaction des systèmes, en particulier l'analyse des flux, la séparation des données et traitements et considère les niveaux organisationnels et conceptuels. (3) L'approche orientée objet est plus orientée conception des systèmes d'information qu'analyse de ces derniers.

A ces trois approches, nous pouvons rajouter l'approche orientée processus permettant de représenter les modèles d'entreprise centrés autour de ses processus (Bouchiba and Cherkaoui 2007) (Tableau 1). Les processus définissent les différentes actions réalisées au sein d'une entreprise, depuis la prise de décision jusqu'à la gestion de ressources. Ils sont étroitement liés au SI de l'entreprise, dont le rôle devient alors critique lors de son adaptation à son environnement. (Sadiq et al. 2004) constate ainsi qu'une entreprise qui ne s'engage pas dans une approche orientée processus est décrite par de nombreux éditeurs comme étant une organisation qui sera dans l'impossibilité de répondre rapidement aux conditions changeantes du marché et de son environnement.

Tableau 1. Principales approches de modélisation d'entreprise

Approches	Structurée	Systémique	Orientée objet	Orientée Processus
Méthodes, langages et outils	SADT-IDEFO IDEF2, 3	CIMOSA, MERISE, GRAI, GIM ; PERA, GERAM ; UEMIL	UML	ARIS SCOR

Ces approches sont accompagnées de cadres de modélisation afin d'en faciliter la réalisation. Nous n'allons pas décrire de façon exhaustive ces différents cadres mais plutôt décrire brièvement leurs principaux concepts.

### 3.4 CADRE DE MODELISATION D'ENTREPRISE

Plusieurs définitions sur les cadres de modélisation, ou *frameworks*, existent (Martin, Robertson, & Springer 2004). Nous proposons la définition suivante :

Un *cadre de modélisation* d'entreprise propose une approche, incluant un ensemble de composants - modèles et définitions - formant un squelette adaptable et/ou modifiable au domaine d'application de l'entreprise et ce dans le but de développer et documenter les descriptions d'architecture.

Il peut exister une confusion entre l'architecture d'une entreprise et son cadre de modélisation. Le premier terme désigne le fonctionnement et l'organisation de l'entreprise alors que le second se concentre en particulier sur la définition et la mise en œuvre du SI et des processus métier de l'entreprise. Néanmoins ces deux termes demeurent indissociables l'un de l'autre. Comme nous l'avons présenté précédemment, afin de capitaliser les savoirs et d'augmenter les performances générales d'une entreprise, nous procédons à un alignement du SI sur la stratégie de l'entreprise, et ce, à l'aide de modèles. La modélisation peut être réalisée selon une mise en perspective du modèle par les utilisateurs, reposant essentiellement sur le cadre proposé par Zachman en 1987. Cette approche est la plus fréquemment utilisée dans des configurations commerciales et gouvernementales. Une seconde approche peut être l'utilisation de cycle de vie comme thème de modélisation, reposant sur les trois normes ISO suivantes : 15704 : 2000 ; 19439 : 2003 et 15288 : 2002. Une telle approche reste populaire en production et automatisation industrielles et en ingénierie des systèmes.

Un cadre de modélisation d'entreprise décrit le plus souvent les trois dimensions suivantes : le cycle de vie, la structure et le comportement du modèle et son degré de particularisation.

### 3.4.1 Cycle de vie

Le cycle de vie d'un modèle représente ses différents niveaux de dérivation. Il décrit le modèle depuis « l'énoncé des prescriptions pour arriver à un modèle traitable par informatique » (Pourcel & Gourc 2005). La dimension exprime les points de vue principaux du métier de l'entreprise (Darras 2004) à travers les trois étapes suivantes :

- Expression des besoins. Ce niveau prend en considération le cahier des charges : l'identification, la définition et la réalisation des spécifications et exigences du client.
- Spécification de conception. Il s'agit d'analyser et de spécifier des solutions en adéquation avec les besoins exprimés dans le cahier des charges. Cette étape n'est pas anodine, il faut effectivement définir ces besoins en fonction des possibilités du système d'information et du marché (où nous recherchons des choix techniques abordables en termes de coûts, performance et délais).
- Description de l'implantation. Ici, nous fournissons une description de l'implantation retenue. Cette implantation du système doit se faire en considérant les autres systèmes existant en prévision d'une intégration ou connexion entre eux.

A l'aide de ces différentes étapes, le cycle de vie, bien que non-temporel stricto sensu, tend à servir d'ordonnancement pour les opérations de l'entreprise (Martin, Robertson, & Springer 2004).

### 3.4.2 Structure et comportement

Cette dimension regroupe les différentes vues de modélisation de l'entreprise. La représentation d'une entreprise à travers un seul point de vue serait incompréhensible. La complexité d'une entreprise peut être représentée selon différents points de vue. Ces différentes déclinaisons permettent une meilleure compréhension de l'entreprise en définissant plusieurs catégories d'acteurs. Dans la suite du manuscrit, nous utiliserons la notion de « vues d'entreprise » afin d'améliorer la lecture. Selon (IEEE 2000) :

Une *vue* est une représentation de tout un système selon la perspective d'un ensemble d'intérêts liés.

Selon la norme (ISO TC 184/SC 5 2000), nous pouvons en dénombrer quatre. La *vue fonctionnelle* décrit les processus et leurs structures. Nous identifions les noms, buts et actions des processus. La *vue informationnelle* forme le data-flow. Cette vue indique quels sont les documents et données utilisés à chaque étape d'un processus. Elle décrit les objets du système, leurs relations et leurs différents états possibles. La *vue organisationnelle* définit les agents responsables et aptes à exécuter le processus. Enfin, la *vue des ressources*, ou vue opérationnelle, spécifie les outils ou les systèmes permettant la réalisation du processus en décrivant les moyens humains et matériels nécessaires, ainsi que le mode de gestion de ces ressources.

Chacune de ces vues manipulent des concepts différents et peuvent être exprimées à travers un formalisme différent, cependant il faut veiller à garder une cohérence globale. Elles ne possèdent pas non plus la même importance lors d'une démarche de modélisation d'entreprise. (Vernadat 1999) et (Touzi 2007) situent la vue fonctionnelle au centre des autres vues d'entreprise comme le montre la Figure 11.

(Jablonski 2009) distingue la *vue comportementale* de la vue fonctionnelle. La vue comportementale définit les dépendances entre processus et donc l'ordre dans lequel ces processus sont exécutés. Ainsi il détermine le *control-flow* entre processus.

### 3.4.3 Degré de particularisation

Cette dimension permet d'identifier l'ensemble des modèles possibles séparés selon leur niveau de généralité du modèle.

Cette dimension comporte ainsi trois niveaux :

- Niveau générique : ce modèle définit les primitives de base du langage de modélisation ;
- Niveau partiel : ce modèle décrit des structures prédéfinies et réutilisables pour un domaine d'application ;
- Niveau particulier : ce modèle décrit les spécificités du système étudié.

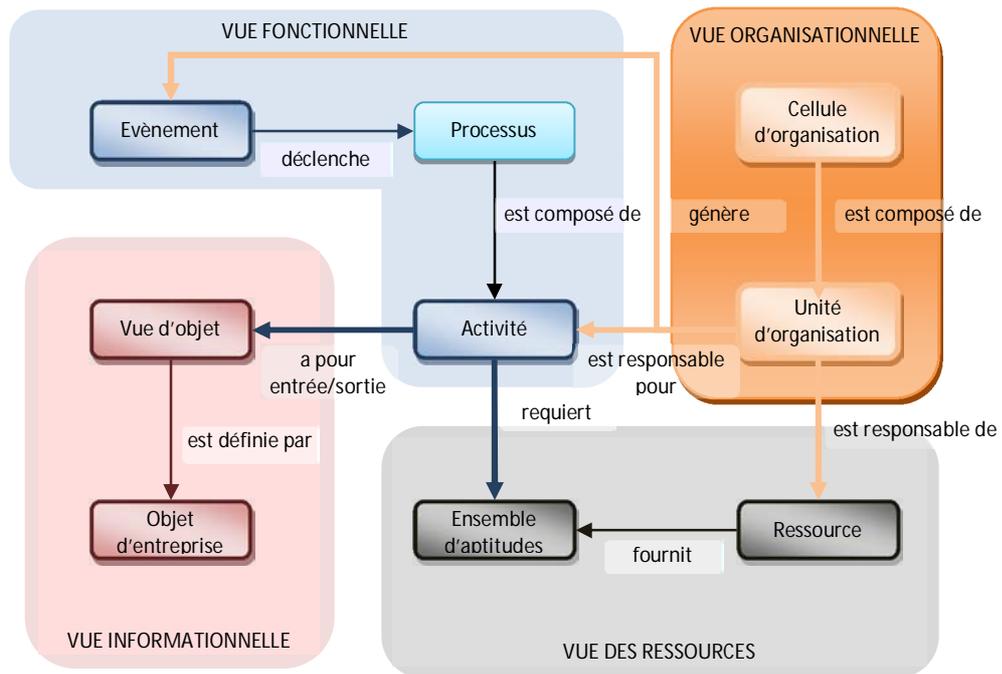


Figure 11. Vues d'entreprises (Vernadat 1999)

### 3.5 CONCLUSION

Afin d'en embrasser toute la complexité, l'entreprise est considérée comme un système de systèmes. Le système d'information se démarque du système de pilotage et du système opérant par son importance. Situé au cœur de l'entreprise, le SI permet de mettre en œuvre les différentes stratégies et objectifs de l'entreprise.

Une entreprise peut être modélisée par plusieurs approches et selon plusieurs vues. La vue fonctionnelle se distingue des autres vues en se positionnant au centre de ces dernières, comme l'indique la Figure 11. Nous pouvons également constater que l'élément processus est un élément central de cette vue fonctionnelle. Au cours de ce manuscrit, l'approche orientée processus peut s'imposer comme l'approche fondamentale garantissant l'agilité requise par l'entreprise. Le chapitre suivant discute des différentes notions attenantes au terme processus.

---

## Notions autour du terme « processus »

---

S'adapter à la globalisation et à une demande versatile, pouvoir améliorer constamment son efficacité et modifier rapidement sa chaîne de valeur sont les défis que cherchent à relever les entreprises depuis plusieurs années. Afin de rester compétitive, une entreprise doit être capable de décrire et de demeurer réactive face à un événement endogène ou exogène. Une telle flexibilité peut s'obtenir en décrivant en processus les fonctionnalités d'une entreprise.

Ce chapitre présente les différentes notions accompagnant le terme processus. Il décrit également le cycle de vie d'un processus et les différentes étapes qui le jalonnent.

## 4.1 PROCESSUS

Afin de réaliser une démarche qualité, l'approche processus est recommandée aux entreprises par la norme ISO 9000 : 2000<sup>8</sup>. Un processus est alors défini comme étant un « ensemble d'activités corrélées ou interactives qui transforme des éléments d'entrée en éléments de sortie ». La définition d'un processus d'entreprise selon (Morley, Hugues, Leblanc, & Hugues 2007) nous semble être la plus adéquate :

Un *processus d'entreprise* est un ensemble d'activités, entreprises dans un objectif déterminé. La responsabilité d'exécution de tout ou partie des activités par un acteur correspond à un rôle. Le déroulement du processus utilise des ressources et peut être conditionné par des événements d'origine interne ou externe. L'agencement des activités correspond à la structure du processus.

Un tel processus est représenté Figure 12.



Figure 12. Processus d'entreprise (Morley et al. 2007)

Généralement, nous adoptons la typologie de (Debauche & Megard 2004) pour différencier les processus d'entreprise selon trois catégories :

- Les *processus de pilotage* ou de management ont pour but d'organiser les objectifs stratégiques de l'entreprise.
- Les *processus opérationnels* ont pour fonction d'accomplir une mission dans un domaine donné et utilise plusieurs fonctions de l'entreprise.
- Les *processus de support* ou de soutien sont périphériques au métier de l'entreprise et ne participent qu'indirectement à l'accomplissement d'un objectif métier.

(SPINOV 2006), (ISO TC 184/SC 5 2000) indiquent qu'une quatrième catégorie de processus peut être considérée, les *processus de mesure* qui fournissent les métriques nécessaires à l'évaluation des processus et à leur amélioration continue. Cette typologie est représentée Figure 13.

<sup>8</sup> ISO : International Organization for Standardization, <http://www.iso.org>

Si (Debauche & Megard 2004) prend bien en compte ces fonctionnalités, il les inclut directement dans les processus de pilotage : « Ils [les processus de pilotage] observent aussi les processus opérationnels et de support pour analyser leur performance, ..., et pour éventuellement les faire évoluer».

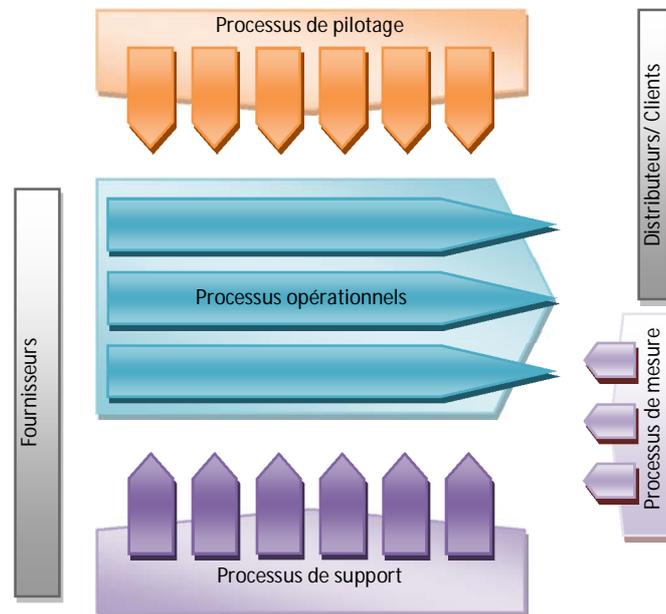


Figure 13. Typologie des processus

## 4.2 PROCESSUS MÉTIER

Les *processus métier* sont les processus représentatifs des activités de l'entreprise, de sa chaîne de valeur. Différentes définitions ont été énoncées au sujet des processus métier. Voici ce qui émerge de ces définitions.

Les processus métier ont pour fonction d'orchestrer les activités participant à l'accomplissement d'un but ou objectif de l'entreprise. Les activités d'un processus métier sont exécutées par des acteurs jouant des rôles particuliers, consommant et produisant des ressources. Les activités peuvent être déclenchées par des événements et peuvent à leur tour produire des événements. Les activités d'un processus peuvent être liées par des dépendances de ressource (dépendances de producteur-consommateur) ou des dépendances de commande (une activité déclenchant une autre). Les acteurs opèrent à l'intérieur des frontières des organisations qui exécutent des fonctions métier spécifiques. Les rôles peuvent soutenir des fonctions.

Le métamodèle proposé par (Morley, Hugues, Leblanc, & Hugues 2007) permet de voir la structuration de ces différents concepts entre eux (Figure 14).

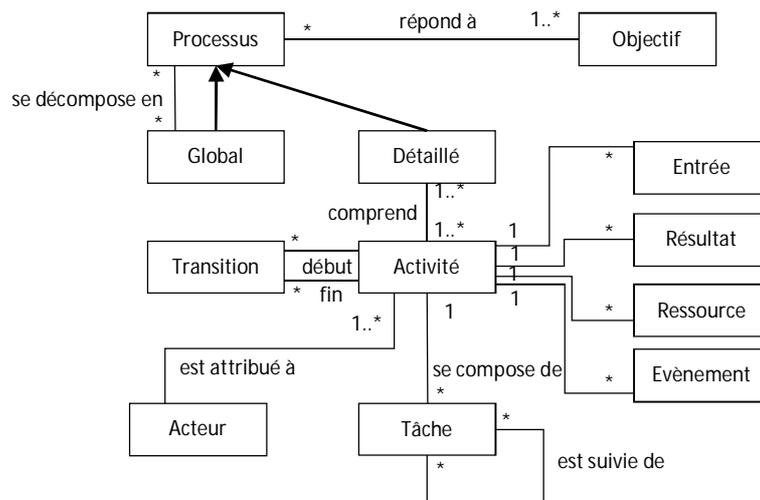


Figure 14. Métamodèle du processus métier selon (Morley, Hugues, Leblanc, & Hugues 2007)

Dans la suite du manuscrit, nous considérons la définition suivante du processus métier :

Un *processus métier* est une orchestration d'activités, incluant une interaction entre différents acteurs sous la forme d'échange d'informations, réalisant des objectifs métier.

L'exemple de la Figure 15 est constitué de trois processus métier : « Vente du produit p », « Gestion des stocks » et « Production du produit p ».

Par nature, un processus d'entreprise possède une dimension horizontale. Cette dimension est multi-domaine. Par exemple, la Figure 15 peut être perçue comme la représentation du processus « Gestion du produit p ». Ce processus est multi-domaine, il dépend des départements « Ventes », « Stocks » et « Production » d'une entreprise. Un processus possède également une dimension verticale, variant selon le degré de spécification recherché par le modélisateur. Ainsi, la Figure 15 peut représenter la « gestion du produit p » (niveau stratégique), contenant le sous-processus « Vente du produit p », lui-même contenant l'activité « Réserver p » (niveau opérationnel).

A travers cet exemple, nous constatons que le processus d'entreprise intègre la chaîne de valeurs de manière transversale.

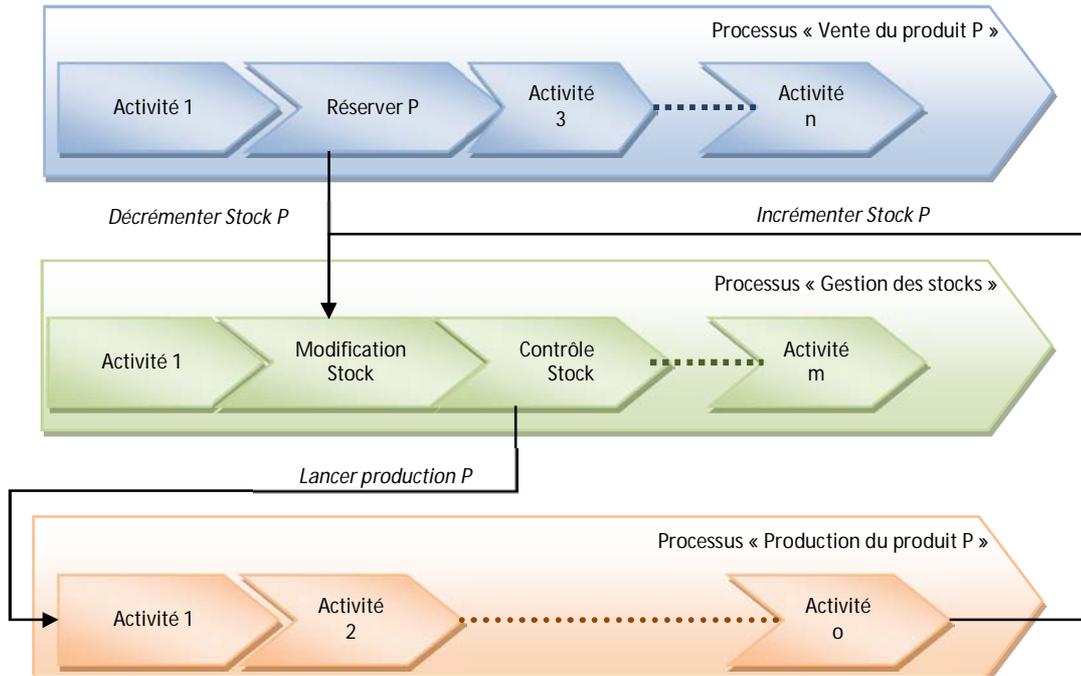


Figure 15. Exemple de processus métier

4.2.1 Processus métier et SI

Les processus métier permettent d’organiser et de répartir les tâches entre les différents acteurs d’une entreprise. Ceci contribue à la réalisation des objectifs stratégiques de celle-ci. Ce processus métier obéit également à des règles et procédures représentant le point de vue organisationnel de celui-ci. Un processus métier est mis en œuvre à l’aide du SI de l’entreprise. Ainsi, un processus métier peut être décomposé en un ou plusieurs processus SI selon (Morley, Hugues, Leblanc, & Hugues 2007).

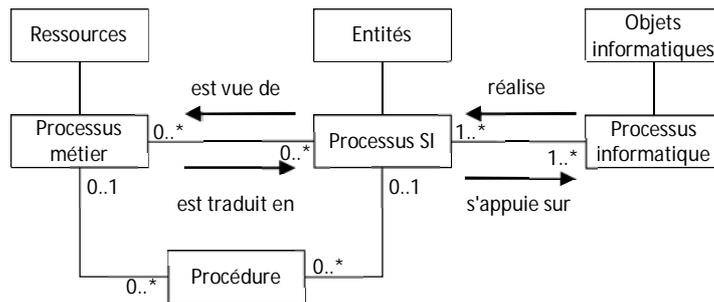


Figure 16. Processus métier, processus SI, processus informatique

Ces processus SI représentant tout ou partie du processus métier participent aux mêmes objectifs mais se focalisent sur la mise à disposition et le traitement de l’information. Ces processus SI sont mis en œuvre par des processus informatiques, soit un ensemble d’activités logicielles, exécutées par des machines et dialoguant éventuellement avec des humains.

L’articulation de ces différents concepts est représentée Figure 16.

#### 4.2.2 Processus métier, processus fonctionnel, procédure

Après avoir exposé la typologie des processus métier Figure 13, nous définissons ses différents niveaux d'abstraction Figure 17 :

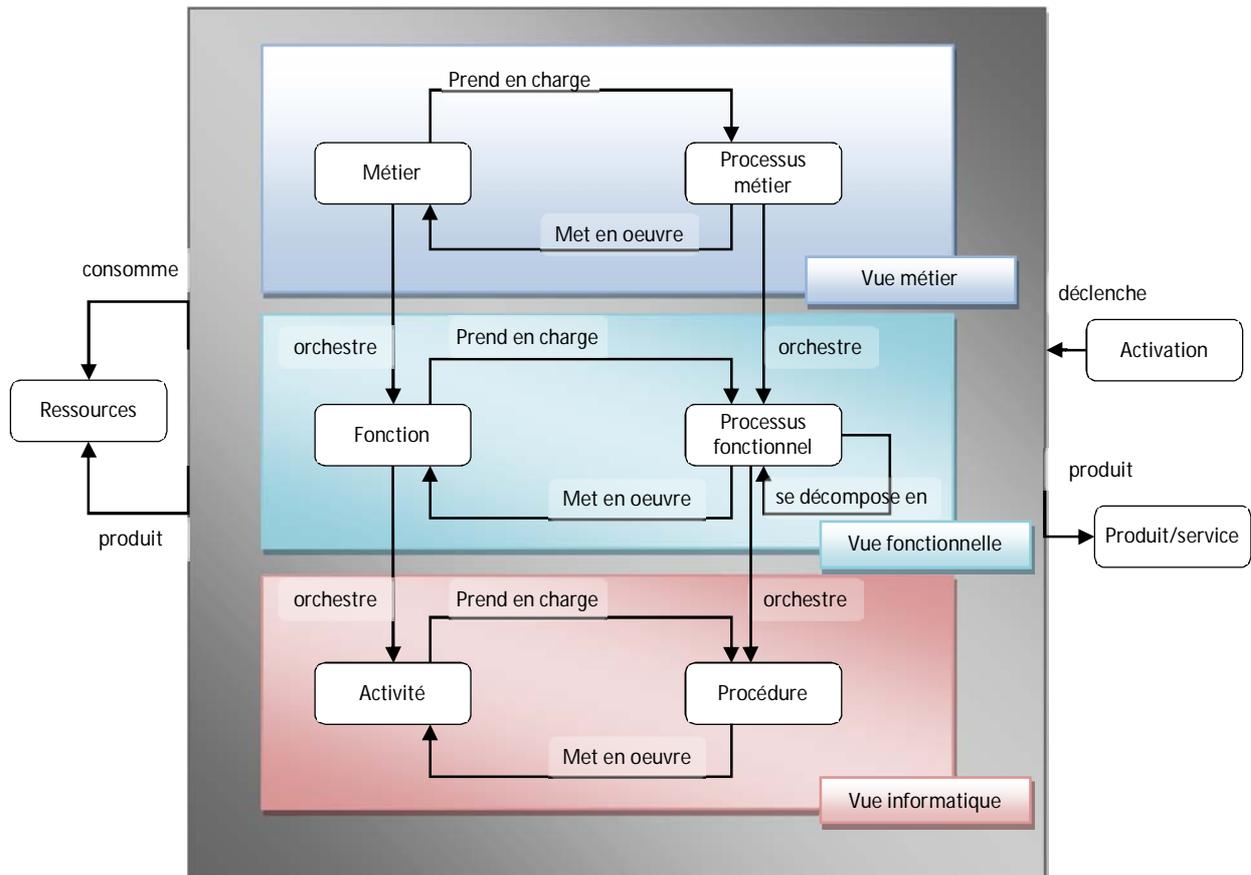


Figure 17. Niveaux d'abstraction du processus métier

Selon (Debauche & Megard 2004), il faut resituer le processus métier selon les trois niveaux d'une entreprise :

- Le niveau métier : où se trouvent les processus stratégiques de pilotage ainsi que les processus opérationnels ;
- Le niveau fonctionnel : où se situent les processus opérationnels et de support. Nous y trouvons, dans une moindre mesure, les processus informatiques d'intégration interentreprise et interapplication ;
- Le niveau informatique : dans lequel se trouvent les procédures, des activités limitées dans l'espace et le temps.

Nous remarquons que (Debauche & Megard 2004) réalise une séparation franche entre processus fonctionnels et processus métier. Or, nous considérons qu'un processus opérationnel peut intervenir à un niveau métier ou à un niveau fonctionnel.

---

**REMARQUE : PROCESSUS, PROCESSUS METIER, PROCESSUS D'ENTREPRISE**

---

Dans la suite du manuscrit, nous utiliserons les termes processus métier et processus pour désigner l'ensemble des processus d'entreprise : processus de pilotage, processus fonctionnel, processus de support, processus de mesure.

---

### 4.3 CYCLE DE VIE ET ACTEURS

Durant son cycle de vie, le processus métier évolue, et est manipulé par différents acteurs. Dans cette section, nous décrivons de manière générique les principales étapes jalonnant le cycle de vie d'un processus. Nous verrons plus tard comment ces différentes étapes seront interprétées par les systèmes de gestion type workflow et BPM section 6.1.

#### 4.3.1 Cycle de vie

Le cycle de vie d'un processus métier est décrit Figure 18. Ce cycle hérite directement des concepts appartenant aux cycles de vie workflow, eux-mêmes héritant des concepts issus de l'automatique.

Les différentes étapes de ce cycle de vie sont réalisées dans deux domaines bien distincts : le domaine métier et le domaine SITI. Les étapes appartenant au domaine métier sont relatives à l'analyse, la modélisation et l'optimisation du processus. L'objectif recherché par les étapes s'effectuant dans le domaine SITI est essentiellement un objectif d'automatisation du processus : son implémentation, son exécution et le prélèvement de données relatives à son exécution. L'ingénierie des processus métier reprend l'ensemble de ces étapes qu'il regroupe selon trois grandes phases : une phase d'analyse, d'exécution et de surveillance. Nous présentons plus en détail les méthodes et outils liés à cette discipline chapitre 6.

#### 4.3.2 Acteurs

Jusqu'à récemment, les premiers cycles de vie ne distinguaient pas le domaine métier du domaine SITI. Il n'y avait pas de séparation franche entre l'implémentation technique et la représentation conceptuelle d'un processus, notamment au sein de systèmes de gestion de type workflow (Van der Aalst 2004), (Heilmann, 1997). Au début des années 2000, l'arrivée de l'ingénierie des processus métier a modifié la perception des acteurs intervenant sur les processus.

Les deux acteurs nécessaires à l'accomplissement d'un cycle d'ingénierie de processus peuvent facilement être identifiés, l'analyste métier et l'expert SITI. Un *analyste métier* recherche de nouvelles façons d'améliorer l'efficacité « métier » de son entreprise. Cette amélioration peut être réalisée en modifiant la manière de travailler collectivement, en changeant d'outils ou de processus (Various IIBA and Brennan 2008). Dans ce manuscrit, nous supposons que l'analyste métier produit systématiquement un modèle juste selon les critères définis par son domaine. L'*expert SITI* prend en charge ces besoins métier et les transforme selon des considérations techniques. A travers l'approche que nous proposons dans ce manuscrit, nous mettons en avant l'importance de ces acteurs et la

contribution qu'ils peuvent apporter afin de diminuer l'écart entre domaine métier et domaine SITI et ainsi permettre un alignement opérationnel tel que défini dans la section 2.2.

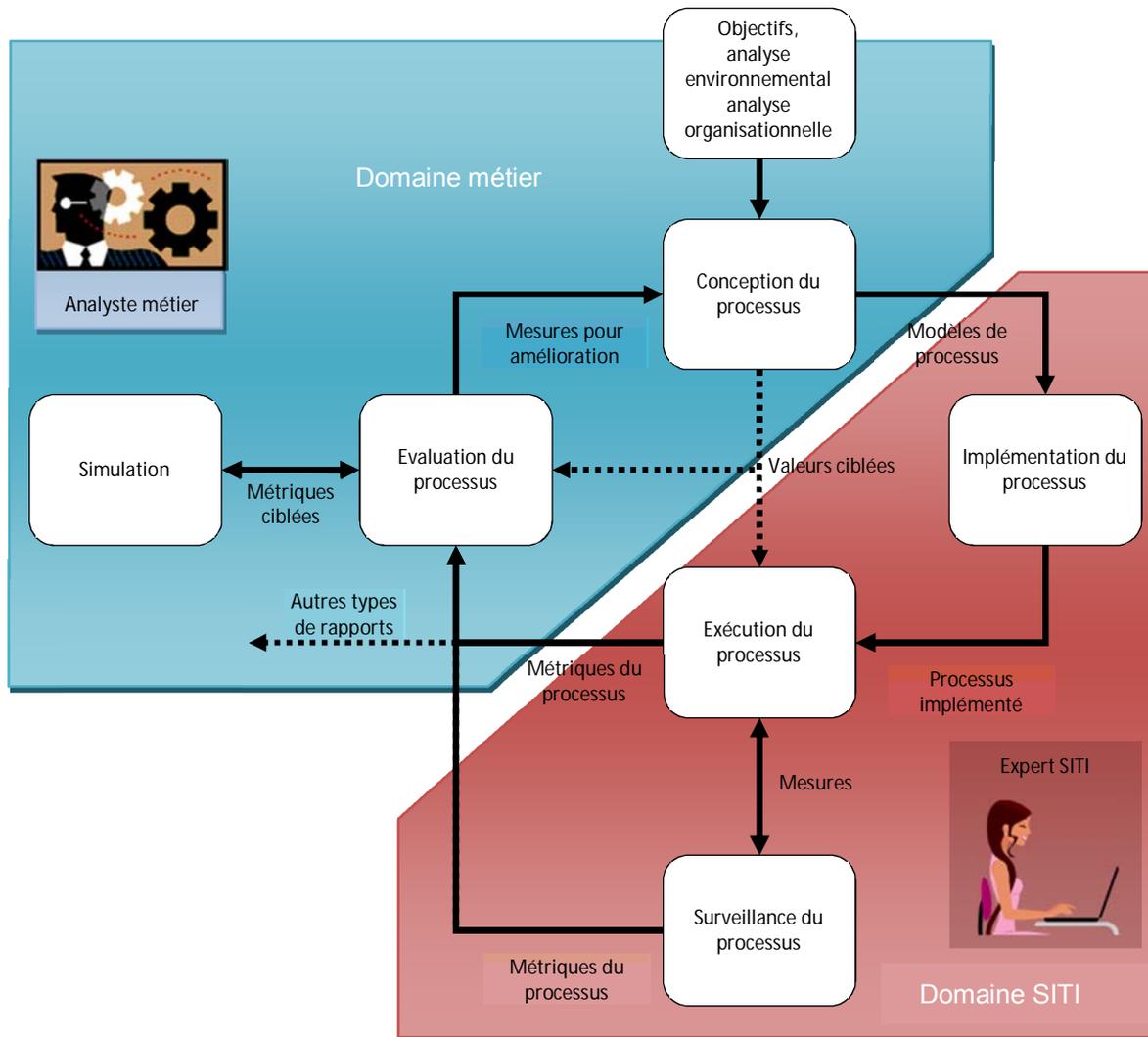


Figure 18. Cycle de vie du processus métier, inspiré de (Zur Muehlen 2004)



P1 : Multi-domaine

Un processus métier doit rester compréhensible et manipulable par des acteurs de domaines différents.

#### 4.4 MODELISATION PAR LES PROCESSUS : IMPORTANCE DE LA VUE FONCTIONNELLE

Si les modèles de processus intègrent généralement plusieurs vues, la perspective « fonction » de l'entreprise demeure l'élément essentiel de l'analyse dans la plupart des méthodologies, en se rapprochant le plus de la définition du processus de gestion décrit par (Scheer, 2002). Le processus est alors décrit selon les fonctions à exécuter et le séquençement entre ces fonctions. Ce séquençement représente le flux des fonctions dirigé par un flux de contrôle décrivant les événements influençant le déclenchement et l'enchaînement logique des fonctions. Cette logique est définie au préalable selon les objectifs métier de l'organisation et l'expertise professionnelle des acteurs de l'entreprise. Néanmoins, ces pratiques professionnelles restent limitées par les contraintes technologiques associées au métier.

## 4.5 CONCLUSION

Ce chapitre montre comment la relation entre l'entreprise et son informatique devient importante de par ses processus étroitement liés à son système d'information.

Selon de nombreux auteurs, ce n'est pas le concept de processus qui est nouveau mais la convergence de techniques informatiques qui permettent de gérer aujourd'hui le cycle de vie complet des processus métier de l'entreprise : identification, modélisation, déploiement, exécution, opération, analyse et optimisation des processus métier. Ainsi, gérer une entreprise à travers ses processus n'est pas une révolution en soi. Une approche orientée processus repose en grande partie sur l'expérience mature de l'industrie dans le domaine de la planification et de la gestion des processus de fabrication.

Une telle approche demeure efficace lorsque le processus modélisé possède une structure formellement définie pouvant être décrite selon une séquence événementielle de processus/activités. Cependant, les processus d'entreprise rencontrés ne remplissent pas forcément ces critères. Il existe ainsi des processus d'ordre conceptuel possédant une description informelle. Dans ce cas, les fonctions le constituant ne sont réellement définies qu'au moment du déploiement du processus. Nous parlerons de processus peu structuré dont la modélisation reste partielle d'un point de vue des SITI. Nous verrons dans le chapitre 6, que l'ingénierie des processus métier, ou BPM, s'évertue à réduire l'écart existant entre domaines métier et SITI. Pour cela, il faut être en mesure de représenter et diriger les processus, et ce tout au long de leur cycle de vie. Nous présentons dans le chapitre suivant les principaux paradigmes et outils portant sur l'utilisation et la manipulation des modèles de processus.



---

## Ingénierie et Architecture Dirigées par les Modèles

---

Les disciplines et méthodes pour le développement et la maintenance des systèmes à prépondérance logicielle reposent sur l'utilisation de modèles abstraits à partir desquels sont générés d'autres modèles ainsi que du code informatique. Afin d'y parvenir, l'Object Management Group - OMG a proposé en 2000 une nouvelle approche : l'Architecture Dirigée par les Modèles. Cette approche a « évolué » en une approche intégrative plus générale, l'Ingénierie Dirigée par les Modèles. Dans ce chapitre, nous décrivons les concepts régissant l'Ingénierie Dirigée par les modèles. Nous décrivons également l'approche reconnue comme une spécialisation de l'IDM, l'Architecture Dirigée par les Modèles.

## 5.1 INGENIERIE DIRIGEE PAR LES MODELES

L'*Ingénierie Dirigée par les Modèles* (IDM) - ou *Model-Driven Engineering, MDE* - est une « approche intégrative générale » (Favre et al. 2006), (Perez et al. 2006), (Combemale 2009) mettant à disposition des outils, concepts et langages pour créer et transformer des modèles. Evolution résultant de l'approche MDA, initiative de l'OMG comme expliquée dans le paragraphe 5.2, l'IDM permet d'intégrer différents espaces techniques, qu'ils s'agissent de technologies orientées objet (reposant UML) ou de documents structurés (reposant XML).

La majorité des méthodes de modélisation définit des modèles contemplatifs, essentiellement utilisés dans des buts de communication et de compréhension entre agents/ acteurs humains. Pour que ces modèles deviennent interprétables et exécutables par une machine, il faut les transformer. Pour y parvenir, il est nécessaire de formaliser les modèles, les transformations qu'ils subissent, les langages de représentation et les métamodèles qui y sont associés. L'idée principale proposée par l'IDM est de pouvoir utiliser autant de langages de modélisation différents, nommés *Domain-Specific Modeling Languages – DSML*, que le nécessitent les aspects technologiques utilisés. L'IDM propose également l'utilisation systématique des métamodèles, des modèles et des processus de tissage de conception suffisamment précis et formels, pour être interprétés ou transformés par des machines.

### 5.1.1 Système réel, modèle, métamodèle

Les liens existants entre système réel, modèle et métamodèle sont représentés dans la Figure 19. Dans le niveau  $M_0$  se situe le système réel. Au niveau  $M_1$ , un modèle représente une simplification de ce système. Ce modèle doit être conforme à un langage d'expression, défini par son métamodèle au niveau  $M_2$  (Bézivin 2004), (Favre, Estublier, & Blay-Fornarino 2006).

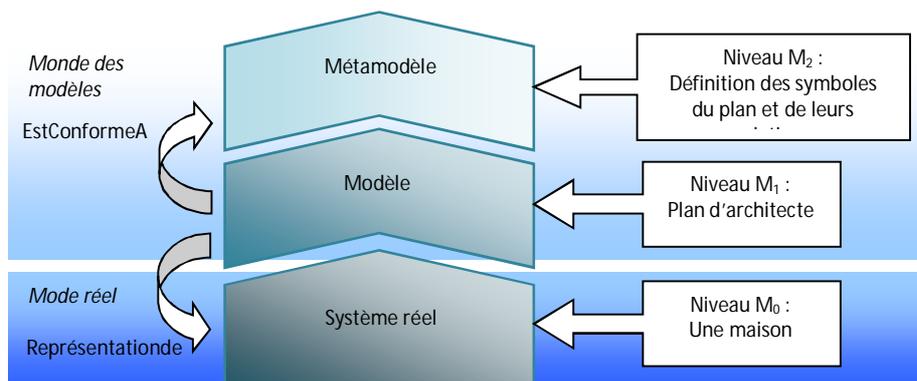


Figure 19. Système réel, modèle, métamodèle

La Figure 19 montre également une représentation simplifiée d'une maison à travers les différents niveaux d'abstraction. Ici, le système réel est la maison (niveau  $M_0$ ) qui est représentée par un plan d'architecte (niveau  $M_1$ ). Ce plan utilise une légende (des symboles) et des conventions de représentation (des contraintes) qui sont définis par le métamodèle (niveau  $M_2$ ).

### 5.1.2 Processus de tissage de conception

Le processus de tissage de conception représente l'ensemble de transformations partiellement ordonné permettant l'obtention d'un artefact exécutable. Chacune de ces transformations prend des modèles en entrée et produit des modèles en sortie. L'utilisation d'un tel processus augmente l'agilité de la conception de produit. En effet, lorsqu'un nouveau produit doit être dérivé, qu'il s'agisse d'une évolution ou variante d'un produit existant, nous nous contentons de réutiliser la majeure partie du processus de conception, à quelques détails près.

## 5.2 ARCHITECTURE DIRIGÉE PAR LES MODELES

Spécialisation de l'IDM présentée par l'OMG en 2000, l'*Architecture Dirigée par les Modèles* ou *Model Driven Architecture* – MDA, s'appuyait initialement sur le standard UML pour décrire séparément les parties d'un système logiciel indépendamment des plates-formes spécifiques les mettant en œuvre. En quelques années, ce projet s'est étoffé, ce qui permet la parution de la spécification version 1.0.1 de MDA en 2003.

MDA reprend la *démarche en Y - 2 Tracks Unified Process*, 2TUP. 2TUP est un processus de développement logiciel pour la description de l'architecture logicielle, en particulier à la modélisation des systèmes d'information (Roques and Vallée 2004). L'axiome fondateur de cette approche est de diviser sa démarche en deux branches : fonctionnelle (approche par les fonctionnalités) et technique (étude de la mise en œuvre).

MDA (Kadima 2005) suggère une nouvelle approche mettant en avant l'utilisation systématique de modèles comme support à la conception et au développement de différents types de systèmes. Pour définir une séparation probante entre le domaine métier d'une entreprise, des logiciels et de plates-formes technologiques des systèmes d'information, et donc séparer les contraintes fonctionnelles des contraintes techniques, MDA utilise trois types de modèles, chacun d'entre eux situés à niveaux d'abstraction très différents. Ces types de modèles sont *les Computation Independent Models - CIM*, *Platform Independent Models - PIM* et *Platform Specific Models - PSM* (Figure 20).

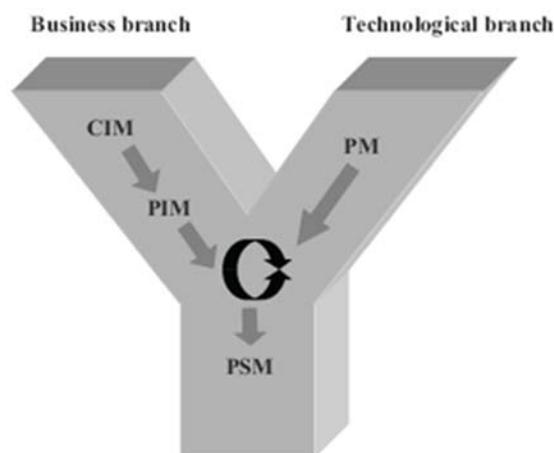


Figure 20. Les modèles MDA selon l'approche 2TUP

A titre d'exemple au sein des travaux Easy-Dim, (Touzi 2007) exploite cette démarche afin de réaliser un Système d'Information Collaboratif – SIC.

### 5.2.1 Rôles des CIM, PIM et PSM

Les CIM désignent les modèles les plus abstraits. D'après (OMG 2003), ces modèles sont associés aux exigences d'un système et/ou à un domaine métier. Plus précisément le CIM décrit usuellement l'environnement, les processus métier et objets ainsi que les exigences spécifiques du système. Il permet de définir les règles et le vocabulaire métier et représente l'aspect organisationnel du système.

Les PIM décrivent tout ou partie des fonctionnalités du système modélisé sans se soucier des détails techniques (Panetto 2006). Le PIM est donc un modèle conceptuel indépendant de toutes considérations liées à la plateforme cible, à son langage ou à la technologie utilisée. Le PIM capture l'aspect logique du processus métier, en respectant les règles haut niveau décrites par le CIM.

Ces PIM sont « techniquement enrichis » afin d'obtenir par génération un PSM. Le PSM est associé à une plateforme spécifique basée sur une technologie bien définie, plateforme décrite par un *Platform Model* - PM. Concrètement, le PSM peut être lié à un système, un langage, ou une technologie particulière, contrairement au PIM. Cette étape est indispensable à la génération du code et donc à l'implémentation du processus cible selon l'approche MDA.

Ces différents modèles et les rôles qui leur sont attribués sont résumés Figure 21.

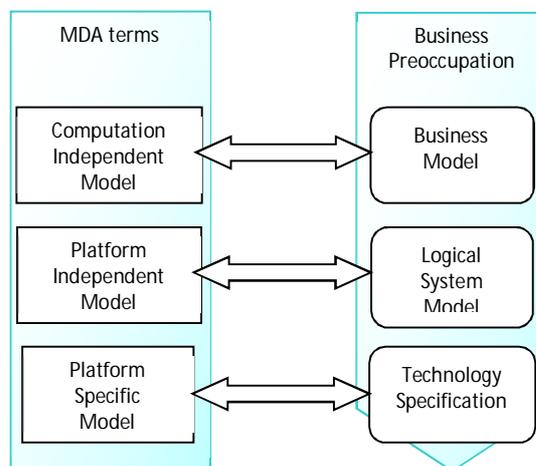


Figure 21. MDA adapté de Model-driven.org

Les différentes étapes lors de la transformation d'un processus modélisé à un processus implémenté à travers MDA sont représentées Figure 22.

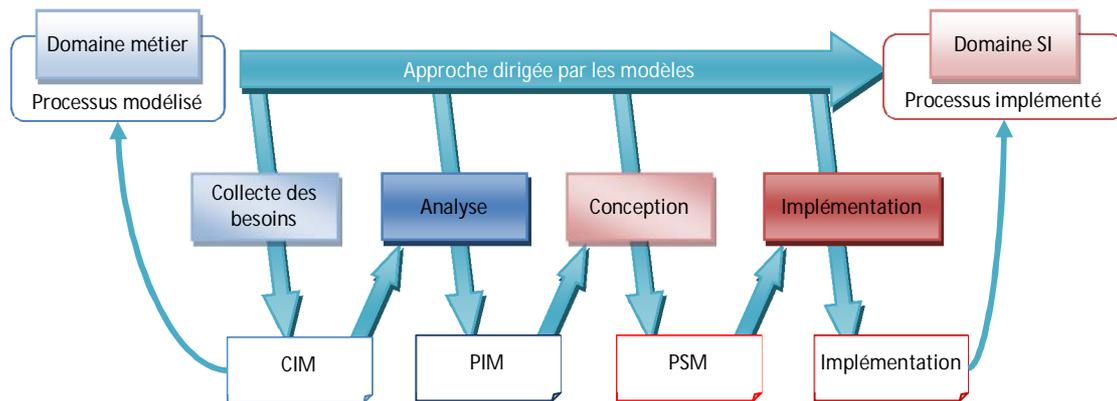


Figure 22. Domaine métier, domaine SI et MDA

Il convient de noter que MDA n'est pas un standard mais un paradigme de développement se reposant sur d'autres standards de l'OMG comme MOF, UML, XMI que nous allons décrire brièvement dans le paragraphe suivant.

### 5.2.2 MOF, UML, XMI

Pour favoriser l'interopérabilité entre ces modèles, l'OMG promeut l'utilisation du métalangage orienté objet le Meta Object Facility - MOF (OMG 2003). Le MOF fournit un socle sur lequel les langages de modélisation peuvent se baser à travers une nouvelle couche d'abstraction : le méta-métamodèle. Il fournit des concepts de métamodélisation standardisés et définit quatre couches d'abstraction (M0-M3) (Figure 23). Ce métalangage supporte la définition de métamodèles en tant qu'ensembles de structures orientées objet (les classes, les packages et les opérations). Il fournit également des constructions spécifiques aux modèles comme par exemple la contenance d'une classe dans une autre, ou encore l'association.

Son utilisation a été popularisée par l'Essential MOF - EMOF, issu du framework de modélisation Eclipse, Eclipse Modeling Framework - EMF. Le MOF préconise la création d'un langage spécifique pour un problème de domaine spécifique et ceux à l'aide de métamodèles.

Parmi les langages de modélisation compatibles avec le MOF, il en est un à mettre en évidence, l'Unified Modelling Language - UML. UML couvre la majorité des facettes connues du développement de logiciels orientés-objets et orientés composants. La méta-modélisation avec UML se réalise à l'aide de profils, substituts de métamodèles. Un profil redéfinit les concepts UML afin de s'adapter à des besoins spécifiques : des stéréotypes et des valeurs tagguées peuvent être appliqués aux différents éléments d'un modèle, comme par exemple le profil SysML pour l'ingénierie système (Fontan 2008).

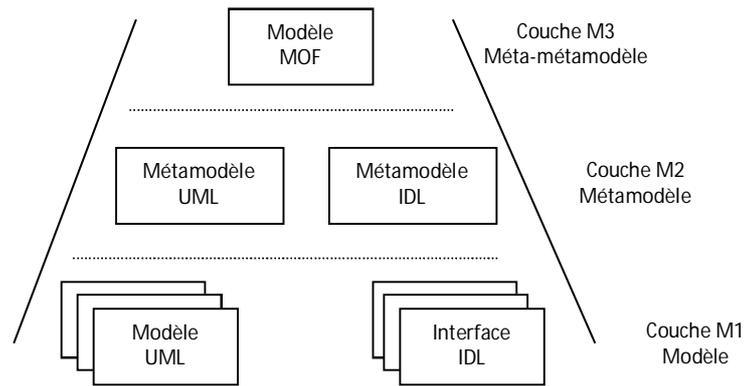


Figure 23. Couches de méta-modélisation selon MOF

De manière à permettre l'échange de modèles MOF et UML entre différents éditeurs logiciels, le langage standardisé XML Metadata Interchange - XMI a été créé (OMG 2007b).

### 5.2.3 Limite du MDA

La principale limite du MDA dans notre contexte est qu'il n'est pas assez investi dans le domaine métier. Le MDA a été créé dans un but de génération de logiciels et non dans un but de gestion de processus métier. Nous constatons que MDA ne décrit pas formellement comment les modèles métier définis au niveau CIM doivent être manipulés et associés aux modèles logiques PIM. Dans la littérature, de nombreux travaux portant sur MDA sont axés sur les modèles logiciels, PIM et PSM, (Steel and Jézéquel 2004), (Bézivin and Blanc 2002) sans prendre véritablement en compte le domaine métier duquel sont issus les modèles CIM. Ainsi, dans certaines utilisations du MDA, l'écart entre le domaine métier et SITI n'est pas envisagé et encore moins résolu, comme le montre la Figure 24. Pourtant nous avons présenté l'importance des aspects logiciels, du domaine SITI et la stratégie qui l'accompagne au sein d'une entreprise. Il faudrait privilégier le domaine métier comme étant le premier aspect à prendre en compte et à gérer dans un processus de développement logiciel type MDA.

Dans le cadre de la modélisation d'entreprise, l'ingénierie des processus métier est devenue un point important. Le but de l'ingénierie des processus métier est de contribuer à diminuer ou au mieux résoudre l'écart entre le domaine métier et le domaine SITI, en particulier lors de l'étape de développement logiciel. Un développement logiciel adapté aux processus métier rendrait l'application développée plus efficiente et plus réactive face aux changements d'ordre métier. En d'autres termes, réduire cet écart permettrait d'augmenter l'alignement opérationnel entre domaine métier et domaine SITI.

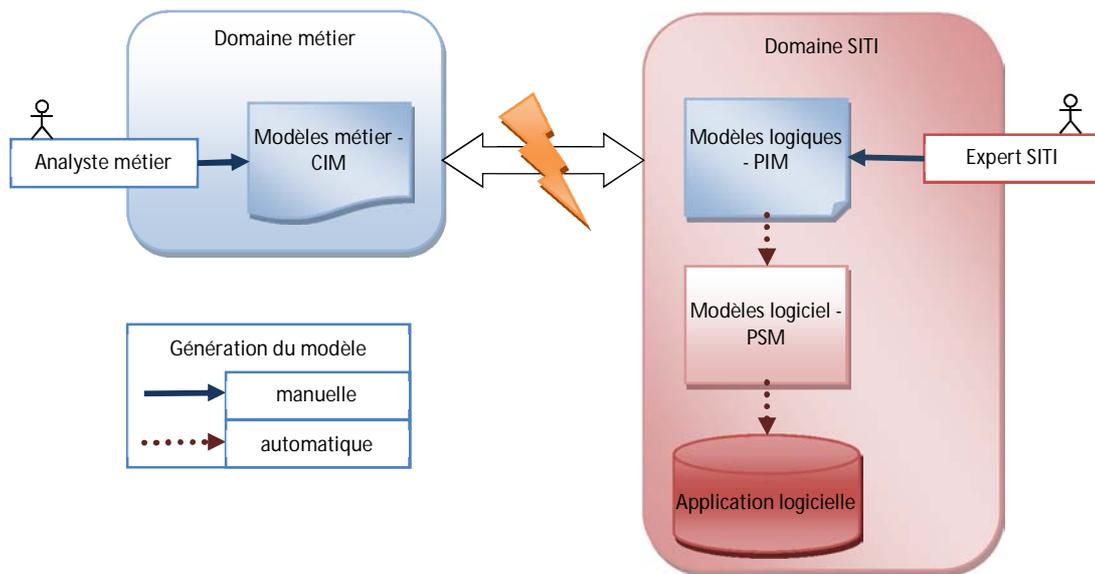


Figure 24. MDA, fracture entre domaine métier et domaine SITI

### 5.3 CONCLUSION

De nos jours, les projets menés au sein d'une entreprise se distinguent par l'accroissement de leur complexité et d'une quantité d'information toujours plus importante. Ces informations sont apportées par plusieurs acteurs, exerçant des métiers et possédant des rôles différents. Afin d'être le plus efficace possible, l'entreprise doit faire preuve d'une gestion moderne et pouvoir gérer conjointement ses différentes compétences.

Les notions issues de l'IDM et de l'utilisation du MDA permettent d'organiser cette gestion multi-domaine et de modèles d'abstractions différentes, de façon fluide et en favorisant au maximum les formalismes de chacun. Néanmoins le MDA reste limité par rapport au domaine métier. Le chapitre suivant aborde la nécessité d'évoluer d'une approche MDA, surtout utilisée pour le développement logiciel, vers une approche orientée métier, l'ingénierie des processus d'entreprise ou Business Process Management.



---

## Ingénierie des processus métier

---

Nous exposons comment *l'ingénierie des processus métier*, *Business Process Management*, met l'accent sur une « architecture orientée-métier ». L'intérêt d'une telle architecture est de fournir une réponse adaptée aux différents besoins des utilisateurs, en abordant la question de l'amélioration continue des processus priorisant le point de vue « métier » au point de vue technique. En effet, BPM préconise une vision transverse des processus métier, « de bout en bout », et non une vision morcelée se concentrant indépendamment sur chaque fonction de l'entreprise.

BPM fournit ainsi une véritable architecture reposant sur un ensemble d'outils et prenant en charge le cycle de vie d'un processus métier. Il permet de définir rapidement et en souplesse des processus depuis leur analyse à leur implémentation, de déterminer leurs objectifs, et de les superviser.

## 6.1 DE LA GESTION DU WORKFLOW A LA GESTION DES PROCESSUS METIER

### REMARQUE

L'ingénierie des processus d'entreprise – ou métier – est souvent représenté par l'acronyme BPM signifiant *Business Process Management*. Pour ne pas déroger à cette règle et alléger le texte, l'acronyme BPM sera utilisé en lieu et en place du terme « ingénierie des processus d'entreprise ».

Le Business Process Management est le plus souvent considéré et interprété comme une nouvelle étape, une évolution naturelle de la vague workflow des années 90. Ainsi, la terminologie employée pour définir le BPM est celle appartenant au *workflow*. La Workflow Management Coalition (WfMC) définit le terme *workflow* de la manière suivante (WfMC 1999):

*The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules.*

Le système prenant en charge ce workflow, le *WorkFlow Management System* - **WFMS** est défini comme étant :

*A system that defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of Information-Technology tools and applications.*

Ces deux définitions mettent l'accent sur l'utilisation de logiciels permettant de faciliter l'exécution des processus métier. Au cours de la dernière décennie, de nombreux chercheurs et praticiens commencèrent à réaliser que cette considération traditionnelle des processus devenait trop restrictive. A la suite de ceci, de nouveaux termes sont apparus comme la gestion des processus métier, ou Business Process Management, incluant dans la plupart des cas, la gestion de workflow. La Figure 25 proposée par (Van der Aalst 2004) montre en effet que le cycle de vie d'un processus selon BPM englobe les mêmes phases que celui de la gestion du workflow :

- Représentation du processus, *Process Design*,
- Configuration du système, *System Configuration*,
- Exécution du processus, *Process Enactment*,

Une des premières différences avec le BPM est que ce dernier rajoute une phase de Diagnostic, *Diagnosis*, à son cycle. La section suivante décrit les fonctionnalités de ces différentes phases au sein d'un cycle d'ingénierie de processus.

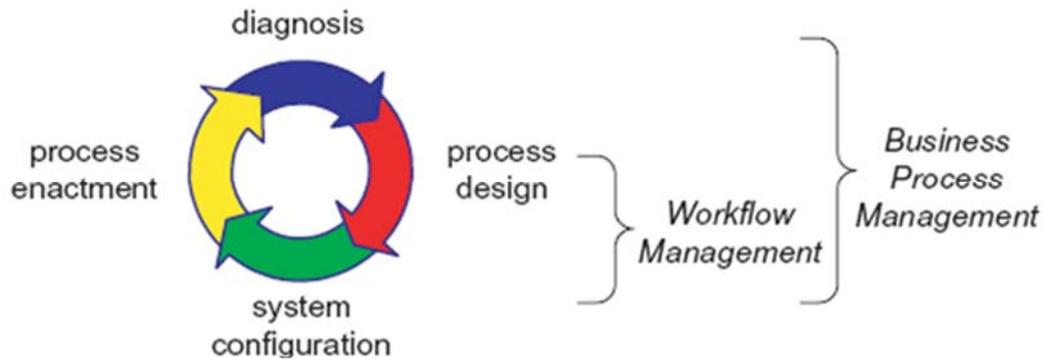


Figure 25. Gestion Workflow et BPM

Le BPM offre aux organisations la liberté de changer très rapidement leurs systèmes et leurs processus, sans avoir besoin de redévelopper complètement leurs applications (Crusson, 2003). L'objectif du BPM est de garantir à une entreprise que ses processus métier sont adaptés de manière continue à un environnement en constante évolution (Fingar and Bellini 2004). Le BPM permet aux entreprises de gérer les processus métier depuis un niveau stratégique jusqu'à un niveau opérationnel (Pyke 2006). Perçu tel un concept permettant l'amélioration de la modélisation et de l'intégration des processus métier mais également l'augmentation de la productivité et la diminution de coût, le BPM s'est imposé comme une discipline à part entière. Cette discipline décrit l'ensemble des activités d'analyse, de modélisation, de conception, de développement, d'exécution, de suivi et optimisation des processus métier. Un consensus des différentes définitions, comme celle de (Malone et al. 2003) ou (Smith et al. 2002), donnerait la définition suivante :

Le *BPM* correspond à l'ingénierie des processus de l'organisation, ou processus métier, à l'aide des technologies de l'information. Il a pour vocation de modéliser, déployer, exécuter et optimiser de manière continue les différents types de processus et ainsi d'améliorer l'agilité d'une organisation.

#### REMARQUE : M POUR MANAGEMENT OU POUR MODELLING ?

La modélisation des processus métier ou *Business Process Modelling – BPM*, désigne la représentation courante (*as-is*) et proposée (*to-be*) des processus. Par défaut, dans ce manuscrit, l'acronyme BPM désigne « BPMmanagement », l'ingénierie des processus métier.

Dans les sections suivantes, nous décrivons les différentes étapes constituant le cycle de vie d'un processus métier selon le BPM. Puis nous fournissons une analogie avec un cadre MDA, où nous observons que BPM permet d'analyser, de modéliser et gérer les processus depuis le niveau CIM vers le niveau PSM, notamment des éléments, comme les acteurs, les ressources, les tâches et l'information.

Les modèles obtenus fournissent généralement une meilleure connaissance du système et une simplification de la vision du problème pour les experts techniques. Puis nous concluons sur les améliorations qu'apporte l'utilisation de BPM mais également sur les limites qui la contraignent.

## 6.2 LE CYCLE DE VIE DU PROCESSUS SELON L'APPROCHE BPM

Le BPM possède une nature multidisciplinaire revendiquant une mise en œuvre de la « théorie à la pratique » à travers plusieurs vues, définitions et perspectives. La présentation du cycle de vie d'un processus métier selon BPM permet d'assimiler la plupart de ces points. Ce cycle accompagne le processus depuis sa conception à sa réingénierie, évoluant en permanence selon les buts métier qui le définissent ou les modifications de son environnement qui l'impactent. Ce cycle d'ingénierie des processus comporte trois phases principales comme le montre la Figure 26: la phase *Business Process Analysis*, la phase *Business Process Implementation* et la phase *Business Activity Monitoring*.

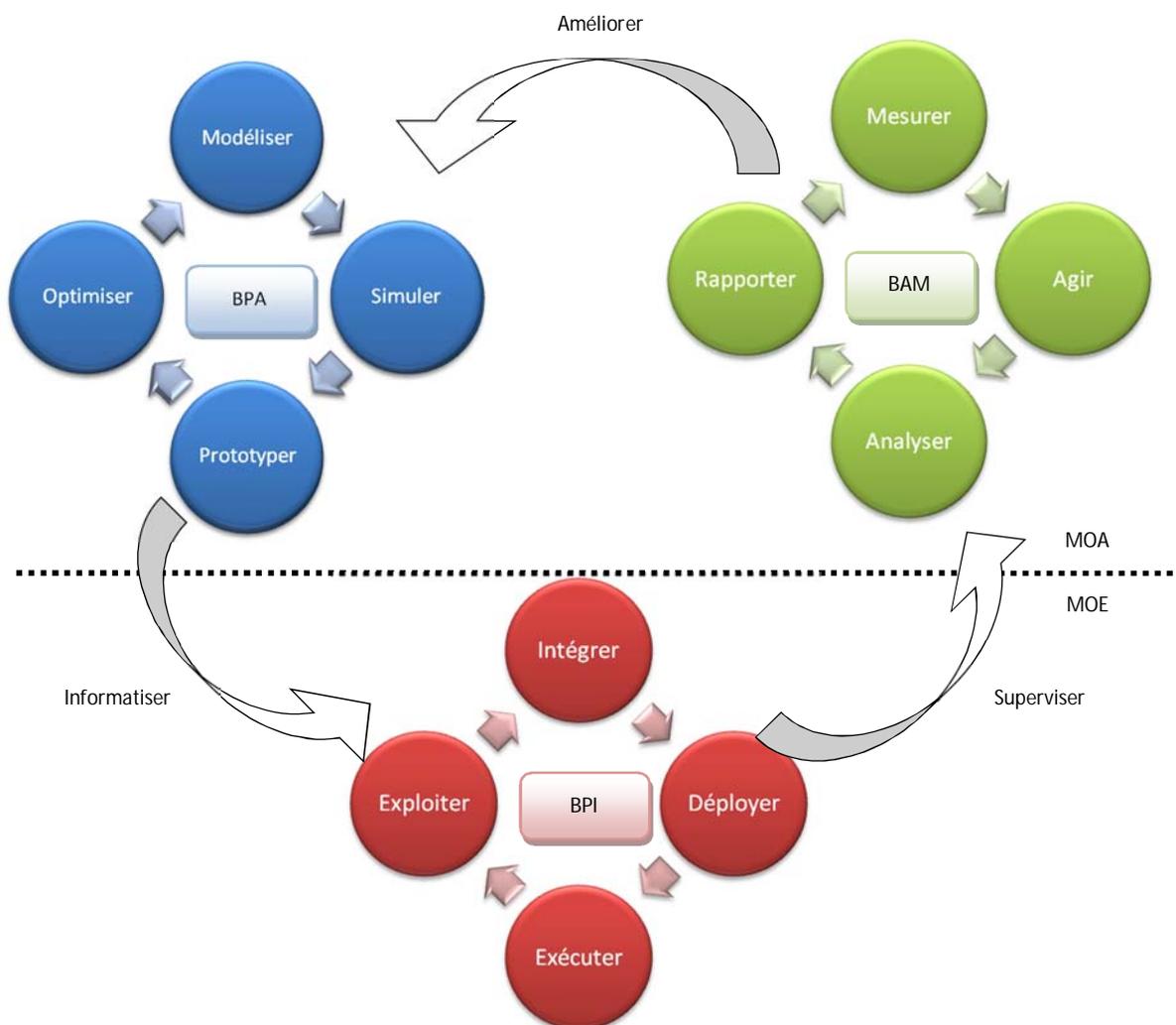


Figure 26. Cycle d'ingénierie continue des processus, BPM (Debauche and Megard 2004)

---

**REMARQUE : GESTION OU INGENIERIE ?**


---

Les lecteurs anglophones auront constaté qu'ici le terme de *management* n'est pas traduit par le terme gestion, mais par ingénierie au sens « génie » du terme, désignant l'art de l'ingénieur. En effet, nous pensons que le terme gestion de processus est trop restrictif pour décrire l'ensemble des possibilités offertes par une approche BPM. Le BPM ne permet pas seulement de diriger les processus. Il permet de considérer un processus sous tous ses aspects, et regroupe l'ensemble des fonctions de l'ingénierie, de sa conception, sa réalisation, son contrôle à sa réingénierie. Egalement, cette discipline comprend des langages, méthodes, outils et standards.

---

**6.2.1 Business Process Analysis, BPA**

La finalité du BPM est de fournir à la direction d'une entreprise une maîtrise accrue dans l'exercice de son métier. Pour ce faire, les analystes métier représentant la maîtrise d'ouvrage doivent identifier et concevoir les processus métier dont ils sont propriétaires. Ainsi, tout effort BPM commence par une phase de conception de processus. Les processus existants (*as-is*) doivent être identifiés et les processus cibles (*to-be*) doivent être définis. La phase de conception est étroitement associée à la phase de modélisation des processus. Durant cette phase, les modèles de processus sont définis à l'aide d'un langage de modélisation, la plupart du temps graphique et respectant une grammaire structurée. Les modèles de processus sont construits selon différents points de vue (flux de contrôle et de données, aspects organisationnels et socio-techniques). Ces actions correspondent à celles réalisées dans la phase « Process Design » d'un WFMS. Comme le cycle BPM est un cycle de vie continu, les processus modélisés sont amenés à être modifiés et optimisés, s'adaptant à un environnement fluctuant. Ces différentes analyses sont similaires à celles constituant la phase « Diagnosis ». Une des caractéristiques la plus importante de la démarche BPM est la possibilité pour un analyste métier de définir les processus métier sans aucune compétence technique. Les analystes métier conçoivent les processus par l'intermédiaire d'une interface facile d'utilisation qui apporte une grande assistance dans la définition de processus métier complexes. Néanmoins la seule façon d'obtenir un processus compréhensible par le SI est d'ajouter des connaissances techniques liées au domaine SITI de l'entreprise.



*P2 : Indépendance  
des environnements*

Un modèle source doit être réalisé indépendamment des considérations de l'environnement cible.

**6.2.2 Business Process Implementation, BPI**
**REMARQUE : « I »**


---

Dans la littérature, le *I* de l'acronyme BPI représente le terme *Integration*. Cependant, compte tenu des tâches que cette phase doit accomplir, nous jugeons le terme *Implementation* plus adéquat.

---

Suite à la phase BPA, une entreprise obtient une description informelle ou semi-formelle de ses processus métier. Il revient aux experts SITI représentant la maîtrise d'œuvre de rendre les processus élaborés précédemment exploitables par un SI. Pour cela, les experts SITI ajoutent des informations techniques : le format des messages échangés, les protocoles de transport utilisés, les transformations de données effectuées, les applications impliquées dans le processus, l'intégration des utilisateurs comme participants du processus, etc.



*P3 : Prise en considération  
des modifications apportées*

Un modèle d'implémentation est souvent modifié avant d'être exécuté. Notre approche doit pouvoir en tenir compte.

Ceci fait, le modèle BPI peut être implémenté sur un large choix d'applications logicielles : les moteurs d'exécution de processus. Nous pouvons les regrouper en trois catégories selon (Debauche & Megard 2004) :

- Développement « maison » ou « sur mesure » : l'application métier est directement générée par la définition du processus métier. Il s'agit d'un développement spécifique et d'une transformation unilatérale. Une fois le code généré puis modifié, nous ne pouvons pas redéfinir le modèle conceptuel au sens rétro-ingénierie.
- Mise en place d'un ou plusieurs progiciels applicatifs : les experts SITI configurent les progiciels afin d'obtenir les processus voulus par les analystes métier. Cette personnalisation reste toutefois limitée par les fonctionnalités fournies par les différents progiciels et demeure incomplète. L'entreprise s'adapte alors le plus souvent au progiciel au détriment de ses caractéristiques métier. Une question se pose alors, l'entreprise doit-elle s'adapter au progiciel et à ses processus « embarqués » afin d'assurer son alignement ? De plus, l'utilisation de plusieurs progiciels (*Enterprise Resource Planning*) afin de fournir une couverture fonctionnelle la plus complète possible au métier de l'entreprise risque de créer des îlots fonctionnels au sein du SI, inaptes à communiquer entre eux.
- Mise en œuvre d'un système spécifique de gestion des processus – *Business Process Management System* (BPMS) : est une plateforme logicielle de production pour modéliser, exécuter et superviser les processus de bout en bout de l'organisation Cette solution permet de considérer les modèles des processus issus du BPA et de générer les applications métier adéquates. Le BPMS est décrit plus en détail en fin de ce chapitre.

Le choix et la paramétrisation de l'application logiciel correspond à la phase « System Configuration » d'un WFMS. L'exécution du processus résultant correspond au « Process Enactment ».

### 6.2.3 *Business Activity Monitoring, BAM*

Un des buts principaux de BPM est de procurer un contrôle et une amélioration constants des processus métier. Les processus peuvent effectivement être suivis en temps réel. Leur exécution contrôlée par le moteur de processus paramétré lors du BPI génère une source considérable de mesures concernant par exemple les délais, les ressources, les coûts,.... Le BAM fournit un ensemble de concepts, pratiques et technologies qui assurent le pilotage de l'entreprise par un ensemble d'indicateurs d'analyse de performance, *Key Performance Indicators* (KPI), métriques calculées sur la base des mesures

obtenues précédemment. Les analystes métier ont besoin de ces indicateurs complets sur les différentes instances des processus qu'ils suivent (en cours ou terminés). Ces KPIs permettent de comparer le déroulement des activités basées sur les processus par rapport aux résultats attendus. Durant cette étape d'analyse, nous identifions les problèmes que nous désirons résoudre. Nous proposons également les améliorations vis-à-vis de l'efficacité des processus. Nous retrouvons ici le lien avec le BPA et la phase « Diagnostic » d'un WFMS. Le BAM regroupe des méthodes qui modifient considérablement les approches actuelles d'aide à la décision en ceci qu'elles travaillent directement sur des données réelles du système d'information (par opposition aux entrepôts de données) et ceci selon un traitement en temps réel.

### 6.3 DU MDA AU BPM

Nous avons décrits dans les sections précédentes deux approches permettant la modélisation des processus, le MDA et le BPM. Nous allons maintenant décrire et identifier les liens existants entre leurs différents éléments.

Le CIM a été décrit comme étant un modèle métier définissant les règles et le vocabulaire métier utilisés par le processus. Nous pouvons ainsi associer un CIM avec le modèle métier obtenu durant la phase BPA. Mais, à ce stade, le modèle métier demeure incomplet sans une description du flux de contrôle de ses processus. Cette information est contenue dans les PIM. En effet, les PIM contiennent les informations relatives aux fonctionnalités du processus modélisé ainsi que leur orchestration. L'association d'un CIM et d'un PIM permet d'obtenir un modèle de type BPA.

Le PSM est une étape critique de génération de code. Le processus modélisé est rendu compréhensible et implémentable sur l'application cible. Il s'agit typiquement des objectifs fixés par l'étape BPI. En conséquence, un PSM peut être identifié avec le modèle d'implémentation BPI.

Ces différentes associations sont représentées Figure 27.

Néanmoins, l'analogie s'arrête là (Smith 2003). BPM et MDA n'ont pas été conçus pour réaliser les mêmes objectifs. MDA a été conçu pour permettre la conception et la génération d'unités logicielles assistées par ordinateur. BPM permet l'ingénierie des processus.

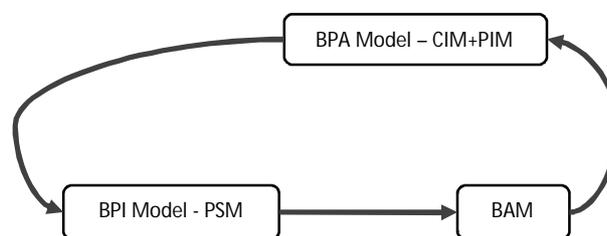


Figure 27. MDA et BPM

## 6.4 SUITES BPM

Des outils spécifiques pour mettre en œuvre un cycle BPM sont les suites BPM ou *Business Process Management Suite* – BPMS (le « s » est parfois considéré comme représentant le terme system). L'arrivée des BPMS marque une évolution dans la structuration des applications du SI de l'entreprise.

### 6.4.1 Une évolution nécessaire

Les architectures deux tiers de type client-serveur, trois tiers puis n-tiers de type web services ont permis la séparation de la couche présentation (client léger, lourd ou riche), de la couche application (règles et traitements) et de la couche données (SGBDR, ERP, *legacy softwares*, EAI ...) (Figure 28(a)). Néanmoins, les processus et la logique métier qui leur est associée, restent « noyés » dans la logique applicative. Il en est d'ailleurs de même avec les règles métier. L'extraction des processus et des décisions dissimulées dans les applications reste délicate. Ainsi les BPMS s'imposent comme une nouvelle couche du système d'information et visent à en extraire les processus (voire également les décisions ou règles métier) selon les mêmes raisons qui ont motivé dans les années 90 les informaticiens à extraire des applications les données dans les SGBD (Figure 28(b)). L'extraction des processus métier des applications du système d'information, notamment des ERP doit permettre l'orchestration des activités d'un processus interne à l'entreprise.

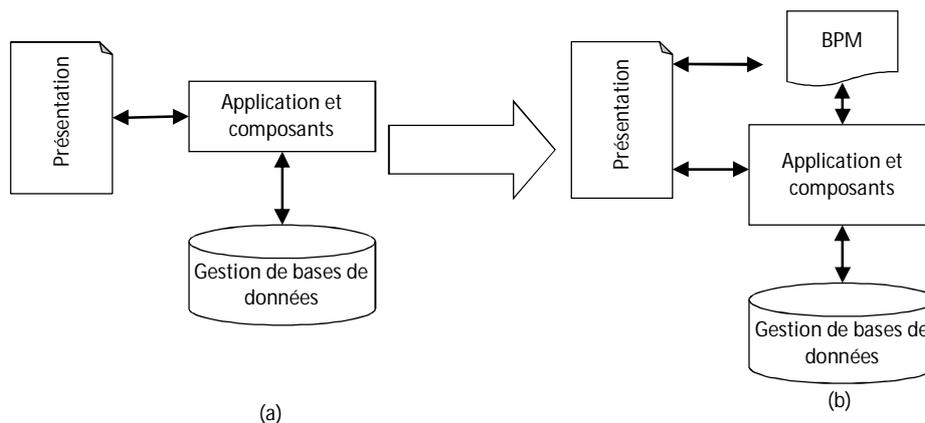


Figure 28. D'une architecture n-tiers à une architecture n-tiers dirigée par les processus

### 6.4.2 Définition

Une suite BPM est une plateforme intégrée d'un éditeur unique rassemblant tous les composants nécessaires au développement et à l'exécution d'une solution BPM : modélisation et analyse, orchestration automatisée, tâches humaines, intégration des applications, règles métier, et supervision. Pour certaines catégories métier, un BPMS peut également se voir attribuer des composants comme la gestion de projet, la gestion de cas, les outils de collaboration,...

En peu d'années, de nombreux éditeurs d'applications se sont intéressés au BPMS et ont contribué au développement d'outils s'y approchant. Au départ, les solutions BPM étaient vendues sous forme de modules se complétant les unes aux autres (un module servant à l'analyse des processus, un autre à leur monitoring...). En 2003, l'entreprise de conseil américaine Gartner identifie cette nouvelle

génération d'outils BPM comme étant des produits BPM « pure-play », coordonnant les interactions entre utilisateurs, systèmes et données. Ces outils sont rapidement devenus concurrentiels et équivalents à des outils de gestion du workflow. Ainsi des éditeurs workflows et d'Intégration d'Applications d'Entreprise ( ou *Enterprise Application Integration* – EAI) sont entrés dans le marché des BPMS. Enfin des éditeurs d'applications plus larges, notamment d'architecture et de modélisation d'entreprise ont rejoint cette mouvance.

La Figure 29 reprend l'évolution des BPMS et des principaux éditeurs y participant, en s'inspirant de l'étude menée par (Silver 2008).

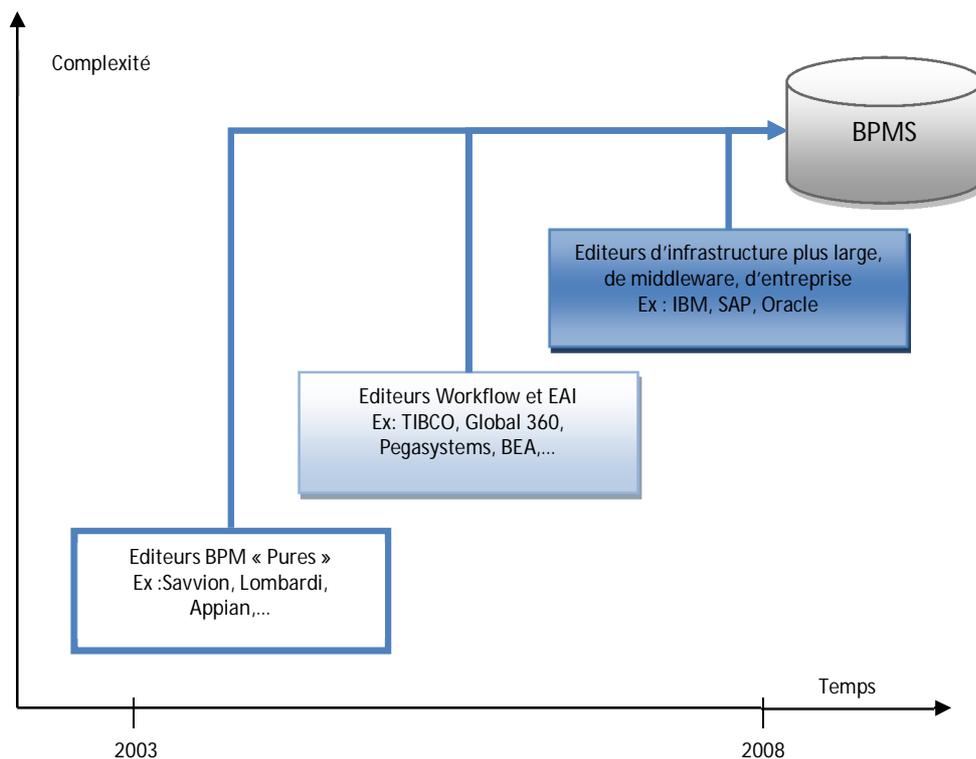


Figure 29. Evolution du marché BPMS

**REMARQUE : APPROCHE BPM MAIS CYCLE DE VIE DIFFERENT**

Aujourd'hui des éditeurs spécialisés, Software AG – IDS Scheer, IBM Telelogic, MEGA, W4, Intalio, Bizagi, ... proposent des plateformes logicielles intégrant un ensemble d'outils supportant une démarche BPM. Par exemple l'éditeur ARIS définit son approche BPM par « ARIS Methodology » qui comprend quatre phases (« BP Strategy, BP Design, BP Implementation, BP Controlling »).

Côté logiciel libre, nous notons l'arrivée en 2009 d'une jeune pousse française, BonitaSoft, qui délivre le BPMS, Bonita (Garcès et al. 2009), en partenariat avec l'INRIA.

### 6.4.3 Structure

S'il existe, à ce jour, de nombreux éditeurs de BPMS de maturité différente, la structure de cette plateforme intégrée décrite Figure 30 reste identique. Le modèle de processus est déployé sur le moteur de processus, qui assigne les tâches aux utilisateurs (cadre « Tâches humaines »), exécute les règles métier (cadre « Règles métier »), intègre les autres plateformes d'exécution (cadre « Intégration »). Le moteur de processus collecte également les données d'exécution et constitue les métriques permettant la surveillance des processus, dans les tableaux de bord BAM. L'ensemble de ces informations peut être réinjecté dans le modèle, permettant l'amélioration itérative du processus.

Il convient de rappeler qu'un BPMS n'est pas un empilement de composants interchangeables, mais essentiellement une plate-forme intégrée éditée par un seul fournisseur.

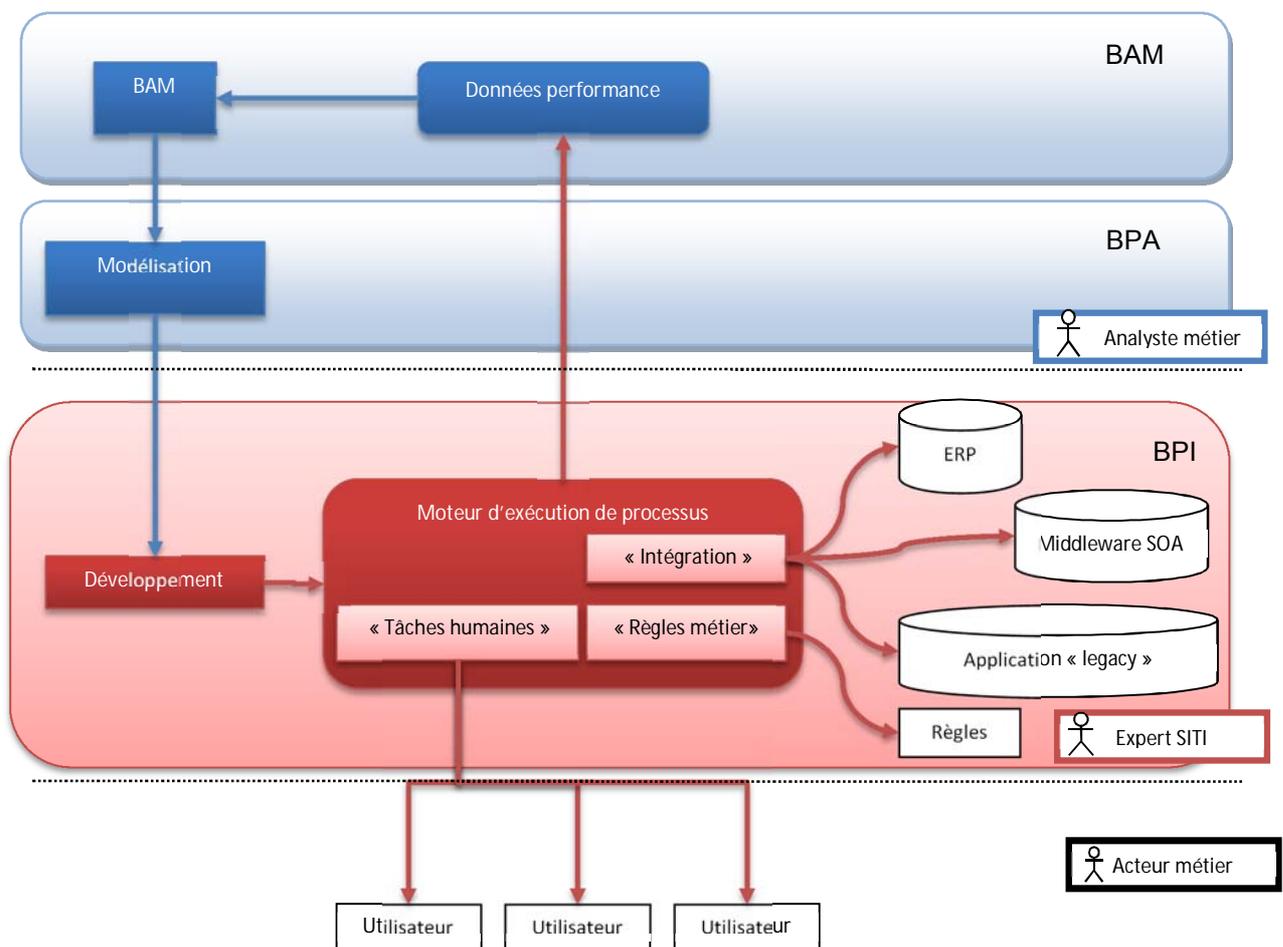


Figure 30. Structure BPMS

### 6.4.4 Classification

Selon (Silver 2008), nous pouvons grossièrement distinguer deux types de processus : les processus *integration-centric*, basés sur l'intégration et les processus *human-centric*, basés sur l'humain. Nous pouvons ainsi classer un BPMS selon les caractéristiques des processus qu'il implémente. Néanmoins, il

revient de considérer que tout (ou presque tout) processus contient à la fois des tâches humaines et de l'intégration d'application.

#### *BPMS pour processus human-centric : BPMS-HC*

Les processus « human-centric » sont des processus mettant l'accent sur le travail effectué par les acteurs métier. Ainsi le rôle d'un BPMS « human-centric » – BPMS-HC est d'affecter et d'accompagner les activités des employés d'une entreprise et d'en mesurer la performance. Cette classification peut elle-même être classée en deux sous catégories : BPMS-HC orienté workflow de production et BPMS-HC orienté gestion de cas.

La catégorie workflow de production désigne les processus métier connus et dont le flux de contrôle est constitué d'activités bien définies et basées sur des règles métier. Il s'agit la plupart du temps de processus structurés comme par exemple des ordres de paiements, ou un centre d'appels. Plusieurs instances d'un processus peuvent être créées puis exécutées. Le but d'un BPMS orienté workflow de production est d'optimiser leur temps de cycle, leur nombre d'exécution par jour et leur coût.

Au contraire, la catégorie gestion de cas regroupe les processus nécessitant un travail collaboratif résolu en temps réel et d'une manière plus ad hoc. La gestion de cas regroupe les processus non-structurés comme les projets, dont le but est clairement fixé mais pas les différentes étapes pour l'atteindre. L'objectif d'un tel BPMS est d'optimiser la prise de décision ainsi que la conformité avec les lois et règlements tout en autorisant une certaine flexibilité parmi les étapes à prendre.

#### *BPMS pour processus integration-centric : BPMS-IC*

Un BPMS « integration-centric » est orienté optimisation de l'intégration métier. Ainsi l'implémentation métier est plus technique et l'orientation de développement plus traditionnelle. Ce type de BPMS favorise l'agilité et l'interopérabilité entre les différentes applications de l'entreprise : ERP, chaîne logistique, CRM ... Il est plus optimisé à détecter et réparer les erreurs en temps réel que de mesurer les performances des tâches humaines.

### *6.4.5 Conclusion sur les BPMS*

Les BPMS sont devenus les outils nécessaires à une mise en œuvre complète de l'approche BPM. Pour rendre une suite BPM opérable, il faut systématiquement paramétrer les différentes applications la composant. Afin de rendre la communication possible entre ces différents composants logiciels d'un BPMS, la solution est le plus souvent mono-éditeur.

L'entreprise et donc la nature de ses modèles étant en perpétuelle évolution, elle doit pouvoir s'adapter à différents éditeurs ou à de nouvelles technologies. A l'inverse d'une approche intégrée de modélisation basée sur un atelier logiciel unique, se traduisant par une dépendance à l'éditeur, l'entreprise doit conserver la possibilité de modifier l'outil de modélisation voire la plateforme d'intégration.



#### *P4 : Evolution des outils*

La solution proposée doit permettre une évolution des outils de modélisation et des plateformes d'exécution.

Il faut également signaler que les BPMS ne renferment pas systématiquement une formalisation des niveaux  $M_1$  et  $M_2$  via une approche IDM formalisée et outillée.

## 6.5 LES LIMITES DU BPM

Le BPM propose une démarche empirique permettant la capitalisation des processus métier. Son cycle d'ingénierie inclut des étapes qui peuvent dépendre du domaine métier de l'entreprise ou de son domaine SITI. Ceci devrait contribuer à réduire l'écart existant entre ces domaines. Cependant, une relation systématique entre le BPM et le SI de l'entreprise reste difficilement réalisable. L'utilisation d'un BPMS ne permet qu'une relation unidirectionnelle, depuis le BPM vers le SI. Lors de la phase BPI, pour rendre un processus vraiment exécutable, les experts SITI doivent documenter un nombre conséquent d'informations techniques. Il existe un grand nombre de standards en ce qui concerne la modélisation du processus, la notation graphique, l'exécution des processus et les langages associés. Néanmoins, la phase d'implémentation reste péniblement automatisable et il paraît illusoire de croire pouvoir fournir directement un modèle BPA prêt à être exécuté. Pour y remédier, les éditeurs BPMS proposent des architectures souvent opaques et peu modifiables (via des connecteurs spécifiques, comme par exemple pour ceux fournis pour l'ERP SAP), à l'opposé de la notion de couplage faible recherché et de l'agilité qui lui est associée.

L'objectif principal du BPM est d'améliorer la communication entre acteurs métier et experts SITI pour rendre les processus de l'entreprise exécutables et contrôlables. Cependant, toutes modifications apportées au modèle lors de la phase BPI n'est pas systématiquement signalées, ou documentées aux analystes métier. Il faudrait à cet effet définir des règles de collaboration entre maîtrise d'ouvrage et maîtrise d'œuvre, ce qui est d'autant plus vrai que le BPM perturbe les frontières habituelles.

## 6.6 CONCLUSION

La démarche BPM est une démarche d'ingénierie dirigée par les processus permettant une gestion des processus métier de bout-en-bout. Il permet un début de conciliation entre domaine métier et domaine SITI. Il accompagne également le processus métier dans sa démarche d'amélioration continue. Le BAM récolte en effet un ensemble de données depuis le BPI afin de constituer des métriques permettant l'évaluation du processus implémenté. Ces données ensuite utilisées dans le BPA permettent de modifier et d'optimiser le processus.

Une démarche BPM réduit les difficultés rencontrées lors du passage entre la modélisation et l'exécution des processus. En effet, nous avons pu constater que les acteurs métier sont réticents à s'appropriier les formalismes proposés par les experts SITI et autres informaticiens pour spécifier simplement les processus à automatiser. Les formalismes de type UML ou bien les démarches d'urbanisation se sont avérés très éloignés des manières de penser des analystes du métier. Une démarche BPM permet ainsi de séparer la vision métier (et le formalisme l'accompagnant) de la vision SITI (et le formalisme associé).

Des suites BPM furent développées facilitant la transformation des modèles BPA vers des modèles BPI ainsi que leur supervision. En effet, les outils de modélisation seuls utilisés par les acteurs métier ne sont pas forcément en mesure d'exécuter les processus modélisés. Rares sont les logiciels d'analyse de

processus utilisés lors de l'étape BPA qui vont jusqu'à l'informatisation des processus ou leur exécution. Rien ne garantit que le modèle de processus soit informatisable lors de sa génération d'un outil de modélisation non-intégré à un BPMS complet. Ainsi au sein de l'entreprise, les modèles de processus restent la plupart du temps purement contemplatifs, déconnectés du SI, utilisés seulement comme un référentiel de connaissances (sous MS Visio ou autre outil de bureautique par exemple). L'alignement métier-SITI n'est ni validé, ni vérifié.

Nous avons néanmoins vu les limites du BPM. Il reste difficile de maintenir un lien entre BPA et BPI permettant d'assurer une cohérence et une synchronisation intermodèle. L'utilisation d'un BPMS s'inscrit dans une architecture fermée et est à ce jour restrictif. Elle va à l'encontre de notre recherche d'agilité au niveau fonctionnel d'une entreprise. La partie suivante mettra en exergue ces difficultés dont la résolution caractérisera notre approche solution.



## TROISIEME PARTIE

---

# C. DEFINITION DE L'APPROCHE

7.	De la nécessité de l'approche	.....	77
8.	Caractérisation	.....	89
9.	Conception	.....	97

## RESUME

---

Nous mettons en avant les dysfonctionnements de l'ingénierie des processus métier lors de son application. Après un court exemple utilisant les standards les plus répandus du BPM, nous discutons des limites d'une démarche BPM. De ces limites, nous exposons les solutions qu'apporte notre approche pivot. Les chapitres suivants définissent le champ d'application de notre approche ainsi que ses caractéristiques. Nous insistons en particulier sur les liens existants entre modèles et métamodèles, ainsi que sur la formation et la constitution du métamodèle pivot, cœur de l'approche proposée.

---

---

## De la nécessité de l'approche

---

Dans ce chapitre, nous cherchons à définir les verrous empêchant l'alignement opérationnel, responsables de la création de l'écart métier-SITI. Nous présentons les concepts et éléments permettant de rendre cet alignement réalisable et nécessaire à l'approche proposée.

## 7.1 VERS UN ALIGNEMENT OPERATIONNEL

Afin d'illustrer les différents propos de ce chapitre, nous considérons la transformation au niveau fonctionnel depuis un modèle BPMN (Business Process Modelling Notation) vers un modèle BPEL (Business Process Execution Language) (Ouyang et al. 2006), (Hettel 2010). BPMN (OMG 2006) est un langage de notation graphique représentant les processus métier. Manipulé par les analystes métier d'une entreprise, le BPMN fournit entre autres :

- Des activités, représentées par des rectangles arrondis ;
- Des portes logiques permettant de faire converger ou diverger le flux de contrôle. Ces portes sont représentées par des losanges ;
- Des évènements identifiant les interactions avec le « monde extérieur », représentés par des cercles ;
- Un séquençage indiquant l'ordre d'exécution, défini par des flèches reliant les éléments évoqués précédemment.

A l'aide de ces différents éléments, une séquence d'activités d'un processus peut être détaillée afin de réaliser un objectif métier (comme produire ou vendre des biens). L'utilisation de portes logiques permet de représenter la prise de décision ou une exécution parallèle. La Figure 31(a) décrit un exemple de modèle BPMN.

BPEL d'un autre côté est un langage d'exécution orienté blocs ou *block-oriented* (OASIS 2007). Utilisé par les experts SITI, il ne possède pas de notation graphique et utilise à la place une sérialisation XML. La Figure 31(b) représente le résultat d'une transformation depuis le modèle BPMN précédemment cité.

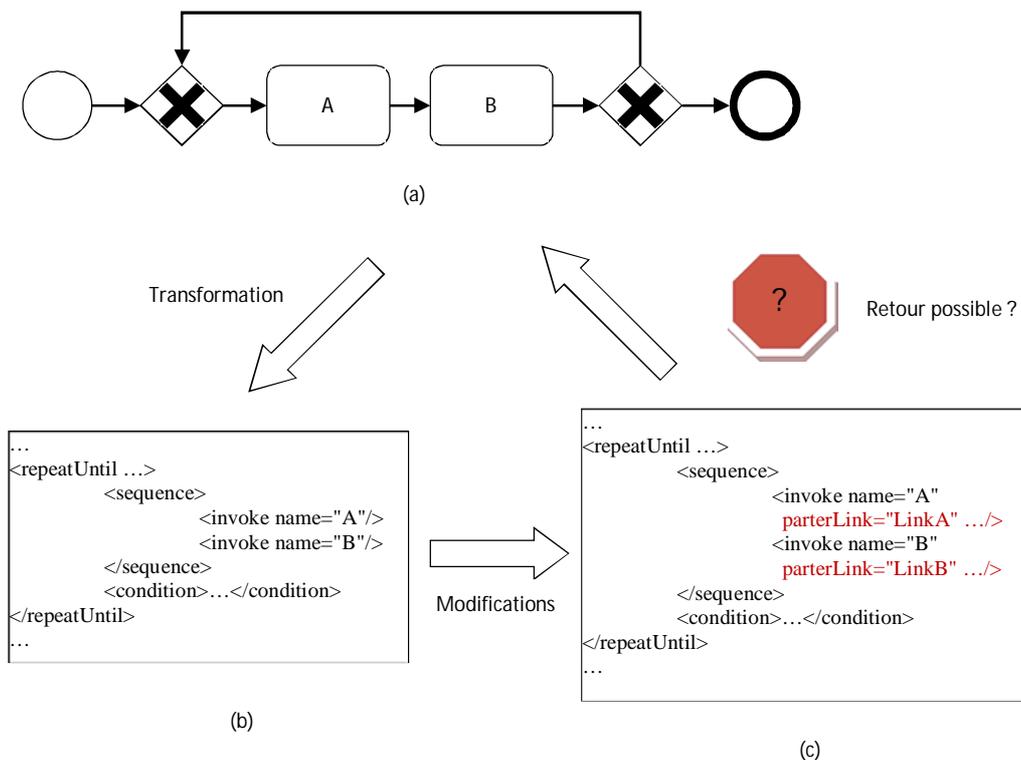


Figure 31. Transformation d'un modèle BPMN à un modèle BPEL

Plusieurs changements interviennent lors de cette transformation. D'une part, il faut préciser que lors de cette transformation, les aspects graphiques du modèle BPMN sont perdus. D'autre part, le modèle BPEL obtenu peut nécessiter des ajustements de code. Par exemple, les variables contenues au sein des activités A et B peuvent nécessiter d'être déclarées comme assignées à des paramètres de services-web. Ces informations nécessaires à l'exécution du processus modélisé sont ajoutées et représentées Figure 31(c), ici les « partnerLinks ».

Suite à cette transformation et à ces modifications, plusieurs interrogations peuvent nous interpeller :

- Les modèles (a) et (c) sont-ils toujours synchronisés ?
- Les modèles (a) et (c) sont-ils sémantiquement équivalents ?
- Les modèles (a) et (c) sont-ils cohérents entre eux ?

Les sous-sections suivantes tentent de répondre à ces différentes questions.

### 7.1.1 Synchronisation

*Les modèles (a) et (c) sont-ils toujours synchronisés ?*

Nous définissons la *synchronisation* entre modèles de la manière suivante :

Deux modèles, i et j, sont dits synchronisés si et seulement si des changements significatifs effectués sur le modèle i peuvent être répercutés sur le modèle j. Est considéré comme significatif tout changement modifiant la structure et/ou le comportement d'un modèle.

Dans l'exemple proposé, les modèles ne sont plus synchronisés. Les précisions apportées à la Figure 31(c) modifient son comportement (ici, la manière dont les différentes activités du processus modélisé sont exécutées). Cependant, ces modifications ne peuvent pas être retranscrites dans un modèle BPMN et donc dans le modèle (a).



*P5 : Synchronisation*

La solution envisagée doit assurer la propagation des modifications d'un modèle à un autre afin de maintenir ces modèles synchronisés.

### 7.1.2 Equivalence sémantique

*Les modèles (a) et (c) sont-ils sémantiquement équivalents ?*

(Izza 2006) définit l'équivalence ainsi : deux concepts sont équivalents s'ils ne représentent qu'un seul et même concept. Il demeure difficile de comparer l'*équivalence sémantique* de modèles

syntactiquement différents et d'abstraction différente. Cependant, en s'appuyant sur (Izza 2006), nous pouvons proposer la définition suivante :

Considérons les modèles  $i$  et  $j$  et l'ensemble respectif des éléments constituant ces modèles,  $E_i$  et  $E_j$ .

Les modèles  $i$  et  $j$  sont sémantiquement équivalents si:

- Pour chaque élément appartenant à  $E_i$ , un élément (ou un ensemble d'éléments) appartenant à  $E_j$  peut lui être associé ;
- Chacun de ces éléments associés suit la même orchestration.

Dans cet exemple simple, les modèles restent sémantiquement équivalents : le déroulement du processus reste identique, qu'il s'agisse du déroulement des activités dans le modèle (a) ou dans le modèle (c). Cependant, les modèles n'étant plus synchronisés, si une modification importante devait intervenir sur le modèle (c), celle-ci ne serait pas répercutée sur le modèle (a), le rendant obsolète et l'équivalence entre modèle ne serait plus maintenue.



*P6 : Equivalence  
sémantique*

Les relations sémantiques entre les éléments de modèles hétérogènes doivent être déterminées à l'aide la plateforme solution.

### 7.1.3 Cohérence intermodèle

*Les modèles (a) et (c) représentent-ils le même processus ?*

Pour les analystes métier de l'entreprise, le processus représenté par le modèle BPEL peut paraître différent de celui défini selon le langage BPMN. Il peut en effet être compliqué à un analyste métier de « retrouver » un modèle BPMN à la lecture des lignes de code de ce même modèle modifié et sérialisé en XML selon un standard mettant en avant l'implémentation des processus. Cependant, les modèles BPMN ou BPEL peuvent exprimer la même chose. Ainsi, le terme « Représentation » semble ici inadéquat. De ce fait, nous le remplaçons par celui de « *Cohérence intermodèle* », terme présenté dans (Ulmer et al. 2010b). Nous supposons qu'un tel lien existe entre deux modèles si :

Il existe une cohérence dite « intermodèle » entre deux modèles si nous observons une équivalence sémantique et s'ils sont synchronisés entre eux.

Nous considérons cette cohérence intermodèle comme une condition nécessaire et suffisante d'un alignement opérationnel tel que défini section 2.2.3. Obtenir un alignement fonctionnel n'est pas une solution triviale en soi, cependant cet alignement est recherché au sein d'un cycle BPM. Les sections suivantes décrivent pourquoi l'alignement métier-SITI est si important et quelles sont les différentes difficultés rencontrées.



*P7 : Cohérence  
intermodèle*

La solution proposée doit garantir l'obtention d'une cohérence intermodèle.

## 7.2 HETEROGENEITE ET DIFFERENTES ABSTRACTIONS

Un processus métier est essentiellement considéré selon deux visions bien distinctes : une vision métier et une vision SITI. Le processus modélisé doit être compréhensible à la fois par les acteurs métier et les acteurs SITI. En conséquence un processus métier est supporté par différents modèles au cours de son cycle de vie. Nous pouvons essentiellement distinguer deux types de modèles : le *modèle d'analyse* et le *modèle d'implémentation* (Figure 32).

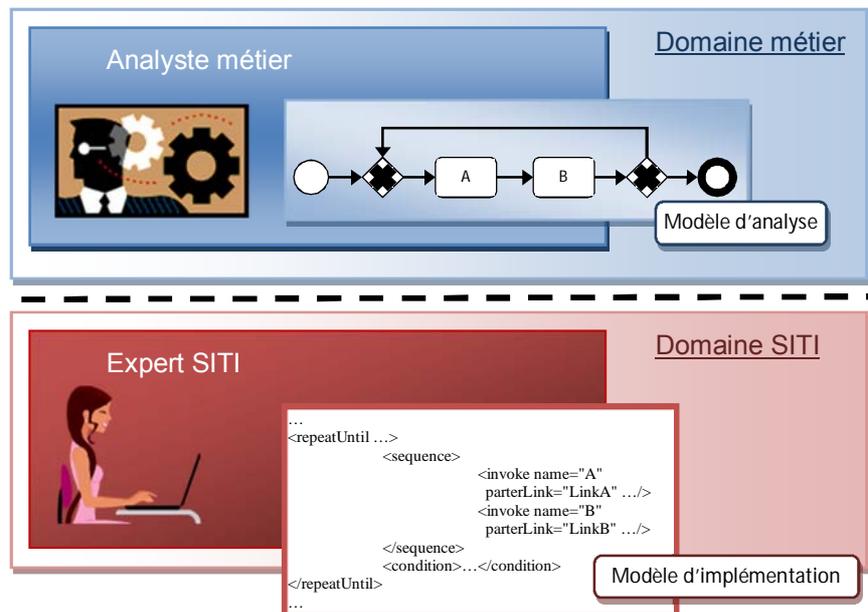


Figure 32. Domaines, acteurs, modèles

### 7.2.1 Modèle d'analyse

A travers la vision métier, nous considérons le processus suivant le point de vue du « monde réel » et selon un environnement donné. Dans le sens où nous désirons représenter un ensemble d'activités réalisées au sein d'une entreprise, le processus est représenté à l'aide de termes métier exprimés généralement dans un langage naturel. La plupart du temps, il s'agit d'une représentation graphique facilitant la communication entre acteurs au sein d'un projet. Le modèle de processus résultant respecte des règles métier<sup>9</sup> conventionnelles, mais pas forcément formelles.

### 7.2.2 Modèle d'implémentation

A l'inverse, une vision SITI permet d'adopter un point de vue technique. Le processus modélisé est décrit à l'aide de notions et de termes propres aux SITI. Le modèle résultant est défini avec précision

<sup>9</sup> Une définition des règles métier est donnée section 7.3.1

selon un contexte donné et respecte normalement les exigences spécifiées entre les utilisateurs finaux et les développeurs. Ce qui permet à ce modèle d'être correctement implémenté sur le moteur cible d'exécution de processus.

### 7.2.3 Deux domaines, deux visions : prémisses de l'écart métier-SITI

Ainsi un modèle conceptuel utilisant une charte graphique, comme le propose le standard BPMN, n'est pas utilisé par la même communauté qu'un modèle d'implémentation *bloc-oriented* reposant sur un standard comme BPEL. Chacun de ces modèles propose ainsi une vision bien distincte.

Nous avons décrit le modèle d'analyse comme un modèle conceptuel permettant de représenter tout ou partie d'un processus de manière compréhensible pour un analyste métier. Les critères mis en avant dans ce type de représentation sont l'aspect graphique et la représentation des flux. Nous ne chercherons pas systématiquement à formaliser les différents éléments du modèle selon une logique « technique » avérée, comme cela est le cas avec un modèle d'implémentation.

Ainsi, la représentation d'un processus selon des critères métier peut être imprécise, ambiguë et sujette à interprétation d'un point de vue informatique. Il s'agit typiquement d'un modèle métier non-formel utilisé à des fins de documentations ou de communications entre acteurs. Si par définition un modèle n'est bien qu'une abstraction d'un système plus complexe, lorsqu'un modèle métier n'est interprétable que par l'humain, alors il s'agit d'un modèle « contemplatif » (Ulmer et al. 2009).

La phase d'analyse produit des modèles de processus métier non formels et interprétables essentiellement par l'homme rendant la phase d'implémentation plus complexe. De ce fait, lorsque le SI est modifié ou évolue, les modèles d'analyse ne sont plus forcément mis à jour. Les modèles d'analyse ne sont donc plus en synchronisation avec les modèles d'implémentation et le système les exécutant. Les experts SITI peuvent considérer que le modèle d'implémentation obtenu suite à une transformation depuis un modèle d'analyse présente des irrégularités et/ou nécessite des modifications. La logique du processus peut ainsi être directement modifiée au niveau de la phase d'implémentation et non au niveau de l'analyse : la sémantique exprimée diffère selon les modèles. Et comme la transformation entre phase d'analyse et d'implémentation est le plus souvent unilatérale, le modèle d'analyse n'est jamais modifié en conséquence, au sens de la rétro-ingénierie.

Si l'utilisation de deux formalismes améliore la description du processus ainsi que sa compréhension envers les acteurs de l'entreprise le manipulant, il ne contribue nullement à améliorer le dialogue entre différents acteurs. L'absence de synchronisation et la non-équivalence sémantique entre modèles ne permettent pas d'obtenir une cohérence intermodèle. Il s'agit typiquement du « Business-IT gap » rencontré dans la littérature, de l'écart existant entre domaine métier et domaine SITI : le non-alignement opérationnel.

## 7.3 CONCEPTS ET APPROCHES POUR UNE GESTION AGILE

A l'aide de l'approche proposée, nous essayons de réduire cet écart métier-SITI. L'approche s'appuie sur une plateforme permettant la transformation des modèles d'analyse vers les modèles d'implémentation et garantissant la cohérence intermodèle. Aussi, l'approche propose une gestion plus

agile des processus métier. Ce résultat est notamment obtenu grâce à l'utilisation de règles métier, la définition d'un couplage faible entre modèles et l'utilisation systématique de métamodèles afin, entre autres, de faciliter et formaliser les transformations entre modèles.

### 7.3.1 Utilisation de règles métier

Utiliser une approche par règle métier permet de s'affranchir des problèmes dus à l'héritage d'application ou *legacy software*, dont le code devient au fil du temps difficile à maintenir et à faire évoluer. Or, nous cherchons à obtenir des systèmes capables de s'adapter rapidement aux changements de l'environnement. Le comportement d'un système d'information doit être modifiable par l'analyste métier sans pour autant qu'il ait à se soucier des contraintes techniques. Selon (Ross 2003) l'approche par règles métier fournit une zone de collaboration entre analystes métier et experts SITI. Les règles métier séparent la logique métier (le comportement) de la logique système d'une application (l'exécution).

Selon (Hay and Healy 1997), nous définissons les règles métier de la manière suivante :

Une *règle métier* est une formulation qui définit ou contraint certains aspects d'un métier. Elle permet de structurer, de contrôler ou d'influencer le comportement du métier.

Ainsi, avec une approche par règles métier, chaque domaine métier d'une application est contrôlé et géré par l'expert métier du domaine (par exemple le banquier pour une application bancaire).

(von Halle 2002) met en avant que si les règles métier se situent au cœur de toute application, cela les rend plus difficiles à recenser et à structurer en une approche de gestion efficace. L'utilisation d'un système de gestion de règles métier ou *Business Rules Management System – BRMS*, facilite le recensement et la mise en œuvre de ces règles.



P8 : *Comportement*

Le modèle d'un processus métier possède des informations sur le comportement des éléments structurant ce processus et leur agencement. Notre solution doit pouvoir représenter distinctement ces deux aspects.

### 7.3.2 Facilité l'évolution des plateformes

Nous avons déclaré précédemment que la principale limite rencontrée au sein d'un cycle d'ingénierie des processus réside dans la « discontinuité » entre le domaine métier et le domaine SITI. Majoritairement, la solution logicielle choisie pour supporter ces transformations est un BPMS (intégré ou non au SI, type ERP). Il convient de garder à l'esprit que les modèles sont amenés à être modifiés, les processus représentés et les technologies de l'information utilisées pouvant évoluer. Des efforts de synchronisation, de mises à jour et de cohérence entre modèles s'avèrent alors nécessaires. L'entreprise et donc la nature de ses modèles étant en perpétuelle évolution, elles doivent pouvoir s'adapter à différents éditeurs ou à de nouvelles technologies. Cependant l'approche intégrée de modélisation basée

sur un atelier logiciel unique comme le BPMS, se traduit par une dépendance forte à l'éditeur. L'entreprise ne possède plus la possibilité de modifier l'outil de modélisation voire la plateforme d'intégration, indépendamment l'un de l'autre (Figure 33).

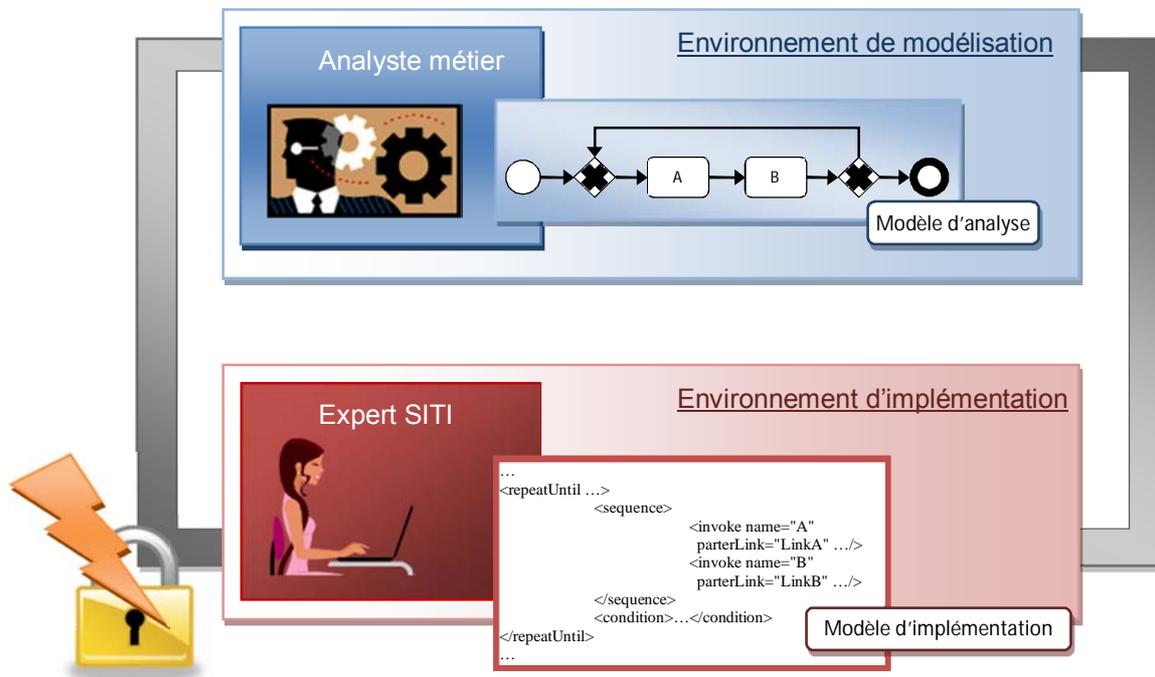


Figure 33. Eviter le couplage fort du BPMS

De nombreuses solutions type ARIS, IBM-Telelogic issues de « l'architecture d'entreprise » utilisent des transformations pour passer d'un modèle d'analyse ou modèle BPA (Business Process Analysis) à un modèle d'implémentation ou modèle BPI (Business Process Implementation). Pour améliorer leur utilisation et leur support, les différents éléments composant un BPMS (et les métamodèles sous-jacents lorsqu'ils existent) sont manipulés de façon opaque et propriétaire. Il en résulte une transformation unilatérale où le modèle de processus métier ne peut ni être transformé sous un autre langage d'exécution ni être modifié sous un autre éditeur. Il est également difficile de garantir la conformité entre modèles et métamodèles.

Par exemple, ARIS contient plusieurs techniques différentes de modélisation de processus métier. Chaque aspect du processus modélisé est décrit par un métamodèle. Néanmoins, il n'y a pas de métamodèle global assurant la cohérence entre ces métamodèles (Leist and Zellner 2006). Il existe néanmoins une DTD - *Document Type Definition* - du modèle global d'ARIS utilisable pour manipuler les exportations XML de ARIS, tous modèles confondus. Cependant, si une DTD définit bien une grammaire pour une classe de documents, elle ne permet pas d'exprimer clairement le vocabulaire et les règles utilisés au sein d'un modèle. Il peut ainsi se révéler insuffisant pour décrire l'ensemble des informations contenues dans un modèle.

Ceci conduit à la volonté d'obtenir un « *couplage faible* » entre le modèle issu du BPA et le modèle issu du BPI. Ici, la notion de couplage faible désigne le fait que bien qu'il existe une interaction forte, ces modèles demeurent « autonomes » et les environnements associés restent modifiables.



P9 : Couplage faible

Corollaire des propriétés P2 et P4, notre plateforme solution doit supporter ce couplage faible entre les outils des différents environnements.

### 7.3.3 Transformation entre modèles : la nécessité du métamodèle

Selon l'environnement dans lequel un modèle est défini, il obéit à un langage d'expression et des règles structurales. Ce langage et ces règles peuvent être formalisés sous la forme d'un métamodèle. Ainsi, les environnements de modélisation et d'implémentation peuvent avoir chacun un métamodèle. Nous désignons le métamodèle issu de l'environnement de modélisation comme étant le métamodèle BPA. Nous procédons de façon identique avec le métamodèle issu de l'environnement d'implémentation, désigné comme étant le métamodèle BPI. Ces différents métamodèles détaillent chacun des aspects différents d'un même processus ; considérer leurs relations peut s'avérer nécessaire et permet d'avoir une vue plus complète du modèle de processus (Saidani et Nurcan, 2008).

La mise en correspondance de ces métamodèles se réalise selon un métier particulier. Ceci implique une prise en compte structurelle et sémantique des processus modélisés. Cependant lors d'une approche BPM, les métamodèles BPA et BPI ne sont pas toujours explicites et formalisés. Les règles de transformation entre modèles BPA et BPI sont alors peu flexibles car définies au cas par cas.

Un des aspects important de l'Ingénierie Dirigée par les Modèles (IDM) est la transformation de modèles. Nous considérons que, de manière générale, la transformation de modèles utilise un ensemble d'applications prenant en entrée des modèles accompagnés de leur métamodèle et produisant en sortie d'autres modèles. La Figure 34 représente cette transformation de modèles selon l'IDM.

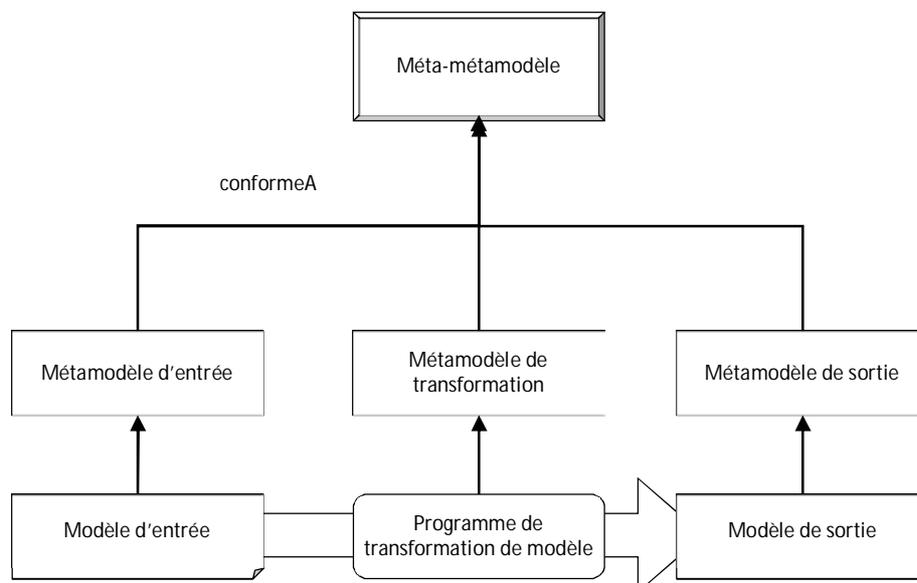


Figure 34. Transformation de modèles selon IDM

Un large panel d'outils a été développé afin de définir ces transformations. Ce panel peut être divisé selon quatre catégories (Faucher et al. 2008) :

- Outils de transformations génériques comportant, entre autres, des outils appartenant à l'espace technique XML comme XSLT ou « XML Query ». Les langages utilisés par ces outils ne permettent pas de travailler la sémantique des modèles.
- Langages de script intégrés à des ateliers de génie logiciel comme par exemple le langage J de l'atelier « Objecteering ». La limite principale de ces langages est qu'ils sont essentiellement propriétaires, pensés comme des langages utilitaires et non des langages de transformation.
- Outils dédiés spécifiquement à la transformation de modèles et normalement conçus pour être intégrables dans des environnements de développement normalisés, comme le langage *Atlas Transformation Language - ATL* (Bézivin & Blanc 2002) par exemple;
- Outils de métamodélisation où la transformation de modèles correspond à l'exécution d'un métaprogramme. Un métaprogramme est en mesure de manipuler les modèles et métamodèles d'entrées et de sorties grâce à la réflexivité du langage. La transformation est alors implémentée en ajoutant aux parties structurelles des comportements grâce à un langage d'action. Le langage Kermeta (Muller et al. 2005) en est un exemple.

Les deux dernières catégories citées permettent de disposer pleinement de langages de transformations à l'aide d'une syntaxe orientée modèle et non objet.

## 7.4 CONCLUSION

Les transformations des modèles selon un cycle d'ingénierie des processus restent compliquées à mettre en œuvre malgré l'aide d'outils comme les BPMS. Ces derniers mettent les dirigeants d'une entreprise en face d'un choix tout aussi compliqué : agilité ou alignement opérationnel ?

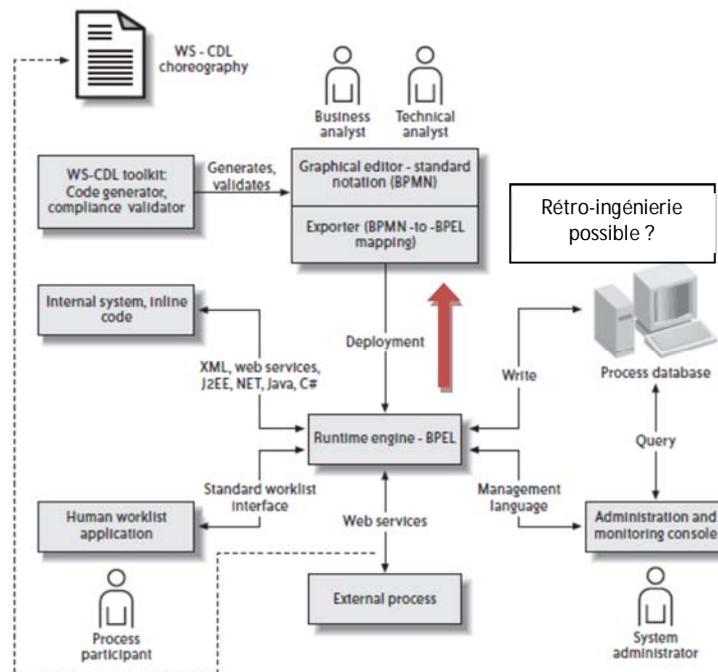


Figure 35. Une « bonne architecture BPM » (Havey and Havey 2005)

En effet, l'utilisation d'un BPMS a pour conséquence de faciliter les transformations depuis la phase d'analyse, BPA, vers la phase d'implémentation, BPI. Un alignement temporaire entre modèles d'analyse et modèles d'implémentation peut être obtenu. Afin d'implémenter correctement les processus, leurs modèles sont modifiés lors de la phase BPI. Ces modifications ne sont pas répercutées sur les modèles d'analyse associés, la synchronisation n'est pas maintenue. Il n'y a plus de cohérence intermodèle et donc plus d'alignement opérationnel comme nous l'avons défini. La transformation reste unilatérale. Toutes modifications apportées au processus à un niveau BPA entraîne une nouvelle génération complète d'un modèle. Les différentes expertises apportées au modèle lors de la phase BPI sont à réitérer. En considérant la « bonne architecture » BPM proposée par (Havey & Havey 2005) (Figure 35), nous remarquons qu'il n'y a pas de retour d'information vers les modèles BPA modélisés en amont. Si ces modèles ne sont pas enrichis et mis à jour, alors ils risquent de devenir à terme des modèles « contemplatifs », des fichiers documentaires difficilement exploitables et désynchronisés des modèles BPI et donc des processus métier exécutés.



*P10 : Transformation*

La transformation entre modèles doit être bilatérale et formalisée à l'aide d'outils de métamodélisation.

Afin de pallier ces déficits, nous proposons une approche se basant sur l'utilisation d'un modèle intermédiaire, un modèle pivot. Cette approche assurerait un couplage faible entre BPA et BPI et la cohérence entre les différents modèles et métamodèles.

Dans les chapitres suivants, nous présentons cette approche. Nous démontrons comment l'utilisation d'un élément pivot au sein d'un cycle permet d'obtenir un alignement entre modèles et environnements hétérogènes. Nous montrons comment les différents concepts et approches liés à l'agilité des processus y sont employés. Notre approche est ensuite mise en œuvre par une plateforme décrite partie D, nommée Solution pour un ALignement et une Cohérence entre Processus, SCALP. L'approche et la plateforme associée sont en mesure de garantir un meilleur alignement opérationnel entre domaine métier et SIT1 via une cohérence intermodèle telle que défini paragraphe 7.1.3 et une gestion plus agile des processus.



---

## Caractérisation

---

Après avoir exposé les difficultés à réaliser un cycle d'ingénierie des processus et du non-alignement qui se crée entre domaine métier et domaine SITI à un niveau opérationnel, nous définissons notre approche et expliquons comment elle parvient à améliorer cet alignement, tout en permettant une gestion agile des processus.

## 8.1 VERS UNE APPROCHE CENTREE PIVOT

Le rôle du pivot est d'assurer une équivalence sémantique entre un modèle analyse et le modèle d'implémentation lui correspondant.

A travers nos différents travaux (Ulmer et al. 2009), (Ulmer et al. 2010a), nous proposons une méthode centrée pivot afin d'améliorer l'alignement des différents modèles issus des phases d'analyse (BPA) et d'implémentation (BPI). L'alignement parfait tel que défini par (Etien 2006) peut être obtenu par l'utilisation d'un pivot. Le concept de pivot a déjà été utilisé selon divers niveaux d'abstraction.

Pour l'implémentation des transformations de ses modèles, le projet FAROS<sup>10</sup> (Blay-Fornarino et al. 2008) utilise cette notion de pivot. Il s'agit ici de transposer des éléments métier, des contrats, au sein de plateformes techniques. L'utilisation d'un modèle et d'un métamodèle pivot permet de formaliser ces transformations depuis la maîtrise d'ouvrage vers la maîtrise d'œuvre et de les rendre reproductibles.

Nous pouvons également citer les travaux de (Thevenet 2010) et de la méthode INSTAL (« INstal Strategic ALignment »). Cette méthode s'intéresse aux intentions d'alignement partagées par les deux niveaux à aligner (stratégique et opérationnel) et qui représente ici l'alignement stratégique.

A un niveau d'abstraction inférieur, nous pouvons considérer, par exemple, les systèmes de gestion de base de données (SGBD) notamment pour les bases de données relationnelles et la méthode MERISE (Baptiste 2009). L'utilisation de modèles différents dans les bases de données composantes pose des problèmes d'hétérogénéité syntaxique. Une solution possible est de traduire tous ces schémas en un modèle commun, le modèle pivot, nommé le modèle logique de données.

Le concept de pivot n'est pas non plus inconnu au sein du domaine métier du *Process Systems Engineering* (PSE). Le rapide développement d'outils d'ingénierie des procédés assistée par ordinateur a entraîné le développement du standard CAPE-OPEN (Belaud 2002). CAPE-OPEN a pour rôle d'accroître l'interopérabilité entre outils logiciels PSE. Tel un pivot de bas niveau, CAPE-OPEN permet, via son middleware, à des logiciels hétérogènes d'éditeurs différents d'interopérer, indépendamment de leurs langages de programmation et de leur système opérant.

A travers l'approche proposée dans ce manuscrit, nous considérons que le système pivot doit être en mesure de faciliter la transformation entre des modèles de perspectives et d'abstractions différentes. Il doit pouvoir ajouter les informations nécessaires, préserver l'intégrité de ces informations et leur cohérence. Il doit permettre une indépendance entre l'environnement de modélisation et l'environnement d'implémentation afin d'assurer un couplage faible entre modèles issus de ces environnements. Ce format intermédiaire devient alors nécessaire pour stocker et échanger les informations des modèles entre les environnements de modélisation et d'implémentation. Notre approche se veut holistique. Il faut identifier tous les aspects décrivant un système afin de mieux le définir, ce qui dans notre cas revient à définir les activités à exécuter, les documents nécessaires à leur exécution, les agents impliqués et les ressources utilisées.

---

<sup>10</sup> Projet ANR RNTL FAROS (composition de contrats pour la Fiabilité d'Architectures Orientées Services) : <http://www.lifl.fr/faros>

Cependant, bien que nous désirions adopter une approche holistique et générique, cette globalité et cette généricité sont dépendantes d'un domaine métier particulier. Dans les sections suivantes, nous établissons la couverture de notre approche et les acteurs nécessaires à sa réalisation. Ce chapitre se conclut par de brefs exemples d'approches utilisant cette notion de pivot.

## 8.2 VUES

Les perspectives, ou vues, utilisées par les processus manipulés, définissent la vision globale de notre approche. Un processus peut être perçu selon les différentes vues d'une entreprise. Chacune de ces vues permet de décrire un aspect important des modèles de processus :

- La vue fonctionnelle décrit les processus et leurs structures. Chacun des processus peut être décomposé en sous-processus ;
- La vue informationnelle décrit les données et les documents qu'un processus utilise ou produit. Ainsi les données entrées et sorties d'un processus sont décrites. Toutes ces entrées et sorties construisent ensemble le *data flow* du modèle de processus ;
- La vue organisationnelle définit les acteurs et les rôles qui sont responsables de l'exécution d'un processus donné ;
- La vue des ressources (ou vue opérationnel) définit les outils et les systèmes qui supportent l'exécution du processus
- La vue comportementale décrit l'ordre dans lesquels les processus sont exécutés, le *control-flow*.

Vu dans la section 3.4.2, la vue fonctionnelle est au cœur des autres vues de l'entreprise et est connectée à toutes ces autres vues par l'intermédiaire de l'élément Activité. Nos modèles d'analyses reposeront essentiellement sur une vue fonctionnelle (Figure 36). Néanmoins les concepts inhérents aux autres vues peuvent intervenir de manière ponctuelle, en fonction des standards et langages de modélisations utilisés.

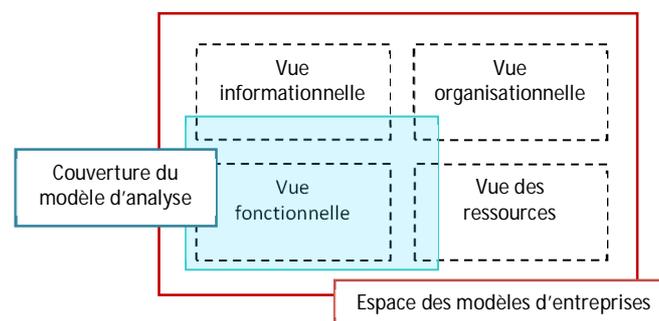


Figure 36. Modèles d'analyse et espaces des modèles d'entreprises (d'après (Touzi 2007))

Considérons par exemple BPMN, un langage standard de modélisation de processus d'entreprise, il est centré sur la vue fonctionnelle, mais il permet de représenter les documents émis ou reçus (vue informationnelle) et de définir des actions intervenant au sein des réseaux de circulation des produits/information (vue des ressources). Notre approche se focalise essentiellement sur la vue

fonctionnelle et est centrée en particulier sur la notion d'activité/processus, telle que perçue par (Vernadat 1999).

### 8.3 GENERICITE

La *généricité* prônée par notre approche n'est pas absolue. Une généricité absolue signifierait que notre pivot est capable de s'adapter à tous types de modèles, issus de langages standards ou spécifiques. Cet objectif peu réalisable rendrait notre métamodèle pivot complexe, difficile à mettre en œuvre et à maintenir. Il faut chercher un compromis entre généricité absolue et simplicité/agilité. La généricité, telle que considérée dans ce manuscrit, se veut générique selon un contexte donné et un domaine métier spécifique. Cette généricité est alors considérée comme étant « relative ». Ainsi notre pivot est relatif à un domaine métier (services bancaires, procédés physico-chimiques...), à un contexte d'étude et au niveau d'abstraction recherché. Néanmoins l'approche orientée pivot proposée ici demeure elle-même générique et indépendante du métier et des technologies employées.

### 8.4 ACTEURS

Nous pouvons facilement identifier les deux acteurs nécessaires à l'accomplissement d'un cycle BPM, l'analyste métier et l'expert SITI. Un analyste métier recherche de nouvelles façons d'améliorer l'efficacité « métier » de son entreprise (Various IIBA & Brennan 2008). Cette amélioration peut être réalisée en modifiant la manière de travailler collectivement, en changeant d'outils ou de processus. L'expert SITI prend en charge ces besoins métier et les transforme selon des considérations techniques. A l'issue des problèmes exposés dans le chapitre 7, nous avons montré qu'il est nécessaire d'adopter une approche holistique et donc d'avoir une vue multi-perspective complète et globale de la méthodologie BPM. Ainsi un troisième acteur se révèle nécessaire, que nous nommons *architecte des processus métier*. En conciliant les deux niveaux d'abstractions propres à l'analyste métier et à l'expert SITI, l'acteur « pivot » permet de « fluidifier » les échanges et d'obtenir une meilleure cohérence entre modèles. Il joue le rôle d'interface. Il possède les connaissances nécessaires métiers et techniques pour définir un « style architectural ». Ces différents acteurs sont représentés sur la Figure 37.

### 8.5 DONNEES

Concrètement le rôle de l'architecte des processus est de déterminer quelles sont les données provenant d'un modèle d'analyse et utilisées par un modèle d'implémentation associé (a), comme par exemple les données relatives au *control-flow*.

L'architecte des processus doit garantir la préservation de l'intégrité des informations depuis le modèle d'analyse vers le modèle d'implémentation résultant (b). Il peut s'agir d'informations graphiques, des annotations non-spécifiques ou non nécessaires pour le modèle d'implémentation.

Il doit également être capable d'apporter des informations nécessaires à la bonne formation du modèle d'implémentation (c), par exemple des détails sur les rôles et les méthodes. Ces données ont à ce stade une valeur par défaut. Elles sont amenées à être modifiées en conséquence par l'expert SITI. De ce fait, les données ainsi modifiées deviendront, lors de la transformation retour, soit des données

utilisables par l'analyste métier (a), soit des données non-utilisables (b), représentées par des flèches en pointillées sur la Figure 37.

L'opération inverse (transformation d'un modèle d'implémentation vers un modèle d'analyse) s'effectue de la même manière et nécessite la même participation de la part de l'architecte des processus. En effet, l'architecte des processus est responsable de la stratégie technique de l'organisation et il doit conserver une vision globale et complète de la méthodologie BPM employée.

La Figure 38 montre en détail comment l'étape pivot fournit les données additionnelles contenant des informations de type (c), à travers un cycle BPM. La figure met en avant les deux cas de transformations rencontrés dans notre approche orientée pivot :

- Première transformation : le modèle pivot permet de stocker des données qui sont propres au modèle d'entrée et non utilisables par le modèle de sortie. Il fournit également des données utilisables par le modèle de sortie et possédant des valeurs par défaut. Ces données sont ensuite manipulées par l'acteur adéquat.
- Seconde transformation : le modèle pivot est en mesure de stocker des données qui sont propres au modèle d'entrée et de restituer les données stockées lors de la première transformation. Le modèle de sortie obtenu possède ainsi l'ensemble de données nécessaires.

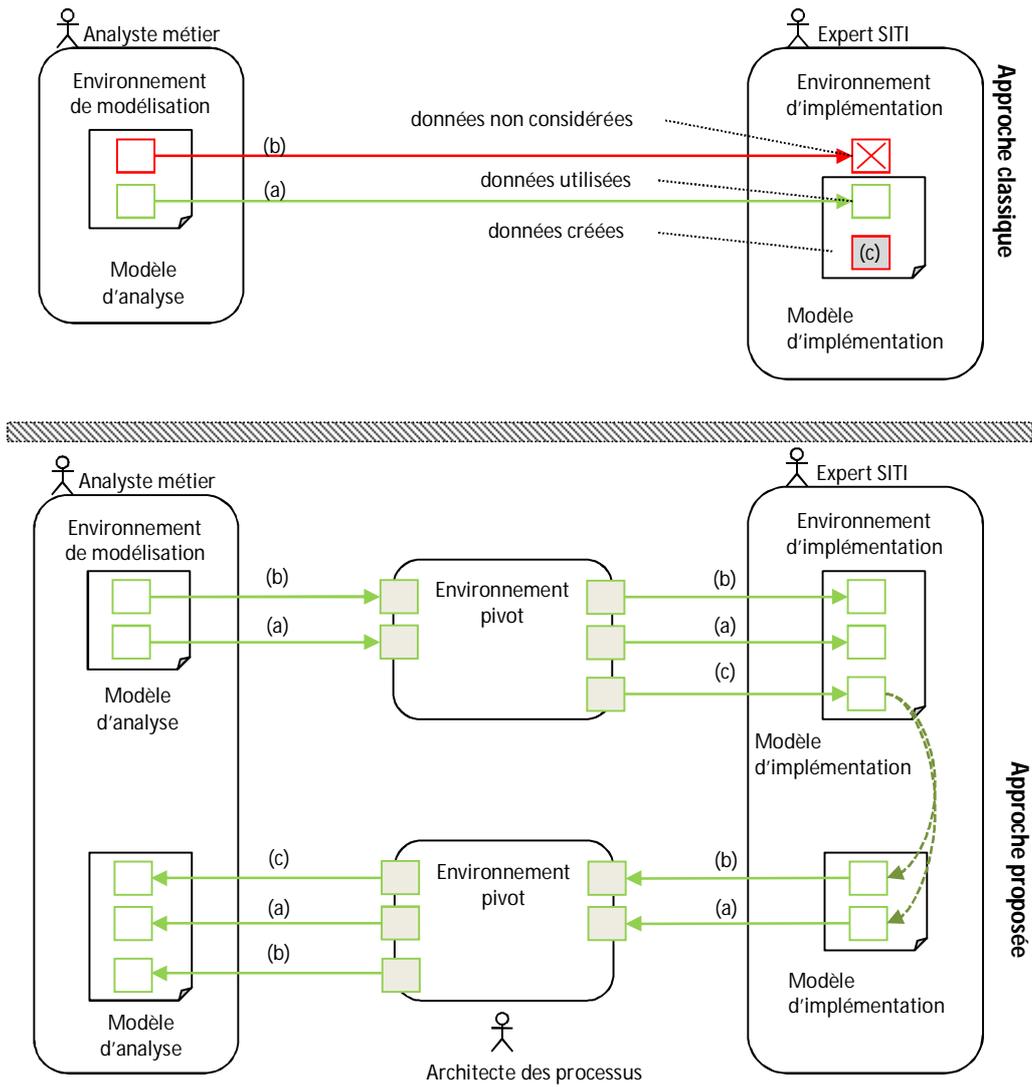


Figure 37. Rôles et données

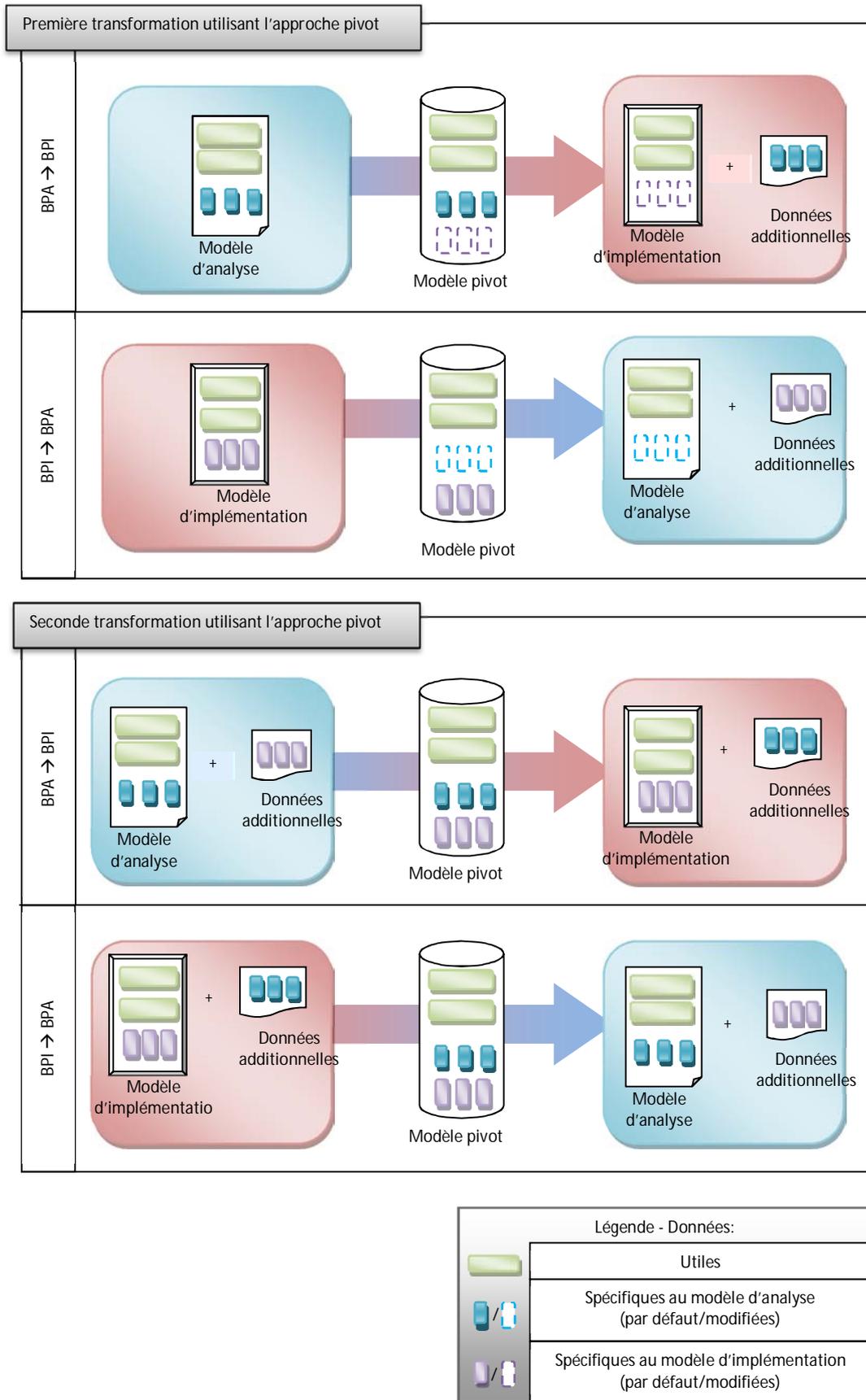


Figure 38. De l'analyse à l'implémentation à l'aide d'un pivot

## 8.6 CONCLUSION

Essentiellement basée sur la vue fonctionnelle, l'approche proposée se veut générique en restant indépendante des langages et des environnements. Nous allons voir qu'elle peut reposer sur l'utilisation de langages spécifiques ou standardisés, langages auxquels nous appliquons des règles de conformités et d'équivalence sémantique afin d'assurer une cohérence « intermodèle » ainsi qu'une transformation bidirectionnelle entre BPA et BPI. L'objectif est d'obtenir des modèles sémantiquement forts, indépendant des environnements de modélisation et d'intégration.

---

## Conception

---

Ici, nous mettons en exergue les efforts de formalisation qu'accompagne l'approche orientée pivot que nous proposons. Nous définissons formellement les liens existant entre modèles d'analyse et d'implémentation, leurs métamodèles respectifs et le métamodèle pivot. Les éléments constituant ce dernier sont présentés et justifiés.

## 9.1 CONFORMITE ENTRE MODELE ET METAMODELE

Afin de pouvoir expliquer les relations entre modèles et métamodèles, nous utilisons des concepts propres à l'IDM. Nous reprenons ici les exemples proposés par (Favre, Estublier, & Blay-Fornarino 2006).

### 9.1.1 Modèle et « ReprésentationDe ( $\mu$ ) »

Considérons un modèle comme étant une représentation simplifiée créée à partir d'un système et selon un objectif bien établi (Bézivin & Blanc 2002), (OMG 2007a). La relation existante entre le modèle et le système étudié peut être représentée par la relation représentationDe, symbolisée par  $\mu$  Figure 39.

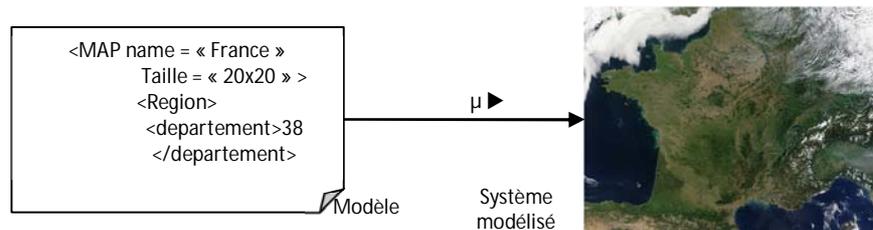


Figure 39. Relation ReprésentationDe

Il faut noter que si la relation  $\mu$  est présente deux fois, le modèle d'un modèle n'est pas systématiquement un métamodèle.

### 9.1.2 Métamodèle et « EstConformeA ( ) »

Un métamodèle est défini comme étant un modèle de spécification d'un ensemble de modèles, menant à l'identification d'une relation *ConformeA*. Par exemple, la Figure 40 représente une carte schématisant la France, en particulier ses démarcations avec les pays limitrophes, entre régions, et ses routes. Ces trois éléments sont définis à l'aide d'une légende. Ainsi la carte représentée est conforme à la légende, qui peut ainsi être considéré comme le métamodèle (ici extrêmement simpliste) du modèle carte.



Figure 40. Relation EstConformeA

Afin de décrire les relations existantes entre modèles et métamodèles, nous utilisons les notations suivantes (Tableau 2. Notations).

Lors d'une approche BPM, les métamodèles BPA et BPI ne sont pas toujours explicites et formalisés rendant notamment les règles de transformation entre modèles BPA et BPI peu flexibles. Afin

d'y remédier, nous définissons systématiquement et formellement nos métamodèles d'analyse et d'implémentation, soit respectivement  $MM_{BPA}$  et  $MM_{BPI}$ .

Tableau 2. Notations

Symbole	Description
$MM_i$	Métamodèle i
$m_i$	Modèle i
$\equiv$	Relation « EstEquivalentA »
$L(MM_i)$	Ensemble des modèles $m_i$ conformes au métamodèle $MM_i$
$\mu$	Relation « ReprésentationDe », ici considérée dans sa version raffinée « Spécifie »
$\chi$	Relation « EstConformeA ».
$\mapsto$	Relation « APourImage »

#### REMARQUE : LANGAGE DE MODELISATION ET LANGAGE DE PROGRAMMATION

La distinction entre ces langages s'effectue entre autres au niveau de leur abstraction. Un langage de modélisation a un niveau d'abstraction plus élevé que celui d'un langage de programmation. Dans la suite du manuscrit, ces langages seront traités indifféremment de leur type (modélisation ou programmation).

Chaque métamodèle est défini selon deux éléments. Tout d'abord, nous considérons que les métamodèles sont constitués d'un standard, ou langage, de représentation noté  $MM_{Rep}$ . Cette notation peut être textuelle, graphique ou mixte. Pour l'environnement de modélisation,  $MM_{Rep}$  est à fortiori graphique pour rester le plus compréhensible possible vis-à-vis des utilisateurs métier.

Le deuxième élément définissant le métamodèle représente sa sémantique. Elle définit la manière dont les concepts de  $MM_{Rep}$  doivent être interprétés. La sémantique peut être exprimée sous plusieurs formes comme par exemple du texte en langage naturel, des définitions mathématiques, des spécifications dans un langage informatique. Elle peut également être classée selon différents types comme dénotationnelle, impérative, etc. Nous considérons ce deuxième élément comme étant un ensemble de règles métier, définissant un référentiel métier que nous notons  $RefMet$ . Ce référentiel peut être obtenu selon des contraintes et/ou règles métier comme OCL (Object Constraint Language) pour UML et SBVR (Semantic Business Vocabulary Rules) pour le BPMN.

Si nous reprenons les relations déterminées dans la section précédente, le  $MM_{BPA}$  est spécifié de la manière suivante ( $\mu$ ) :

$$MM_{Rep} \mu MM_{BPA} \quad (1)$$

$$RefMet \mu MM_{BPA} \quad (2)$$

De la même manière, le modèle BPA (noté  $m_{BPA}$ ) se doit d'être conforme à ( $\chi$ )  $MM_{Rep}$  et  $RefMet$ . En effet, le modèle BPA doit respecter à la fois le langage de modélisation utilisé ainsi que les règles métier. Nous déduisons alors la relation suivante :

$$m_A \chi MM_{BPA} \quad (3)$$

Cette relation est indiquée en pointillés (Figure 41).

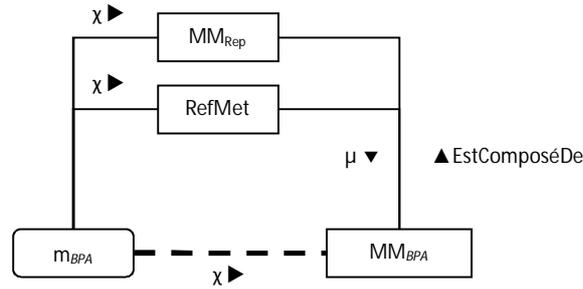


Figure 41. Relation entre  $m_{BPA}$  et  $MM_{BPA}$

Nous pouvons expliciter par analogie les différents liens existant entre le modèle BPI noté  $m_{BPI}$ , et  $MM_{BPI}$  via leur propre  $MM_{Rep}$  et  $RefMet$  (Figure 42).

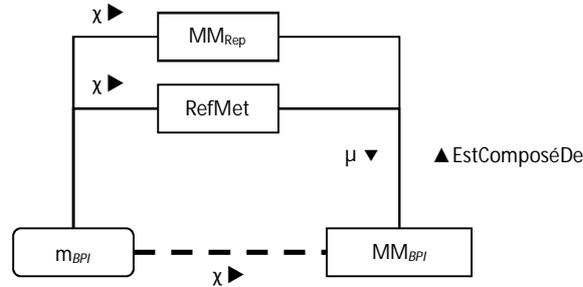


Figure 42. Relation entre  $m_{BPI}$  et  $MM_{BPI}$

Dans la section suivante, nous cherchons à définir explicitement la notion de pivot et les relations pouvant exister entre les modèles et leurs métamodèles et les éléments dérivant de cette notion de pivot.

## 9.2 DE LA TRANSFORMATION DE MODELES BIDIRECTIONNELLES A LA NOTION DE PIVOT

La transformation bidirectionnelle entre modèles permet de justifier mathématiquement la consistance entre eux. Dès lors, ces deux modèles seraient considérés comme étant équivalents et établissant une cohérence intermodèle.

### 9.2.1 Relation bidirectionnelle

Considérons  $MM_A$  et  $MM_B$ , métamodèles respectifs des modèles  $m_A$  et du modèle  $m_B$ . Une relation d'équivalence peut exister entre métamodèles si :

- Avec  $m_A \chi MM_A$  et  $m_B \chi MM_B$

$$L(MM_A) = L(MM_B) \quad (4)$$

Autrement dit, l'espace des modèles conforme à  $MM_A$  est identique à celui de  $MM_B$ .

La transformation bidirectionnelle  $f$  permettant d'obtenir cette équivalence est d'ordre bijectif :

$$- \quad f : MM_A \mapsto MM_B \text{ et } f^{-1} : MM_B \mapsto MM_A$$

Nous obtenons :

$$\forall m_A \chi MM_A, \exists! m_B \chi MM_B, f(m_A) = m_B, f^{-1}(m_B) = m_A \quad (5)$$

Ainsi pour chaque modèle  $m_A$  conforme à son métamodèle  $MM_A$ , il existe un unique modèle  $m_B$  conforme à  $MM_B$  tel que  $f$  transforme  $m_A$  en  $m_B$  et inversement.

Néanmoins cette transformation est trop restrictive et devient impossible à déterminer si les cardinalités entre modèles sont différentes (Stevens 2008). Hors, nous avons vu que nous considérons des modèles qui peuvent être hétérogènes et de niveaux d'abstraction différents. Il peut avoir des informations dans chaque modèle qui ne sont pas contenues dans les autres, en particulier entre modèles appartenant à des domaines différents, le domaine métier et le domaine des systèmes et des technologies de l'information.

Une approche possible est de prendre un de ces modèles et le transformer de manière à ce qu'il puisse contenir l'ensemble des éléments, concepts et informations des modèles et al. (2009). Soient  $\tau_A$  et  $\tau_B$  deux transformations et  $MM_{Int}$  un métamodèle intermédiaire :

$$- \quad \tau_A : MM_A \mapsto MM_{Int}$$

$$\tau_A(m_A) = m'_A \quad (6)$$

$$- \quad \tau_B : MM_B \mapsto MM_{Int}$$

$$\tau_B(m_B) = m'_B \quad (7)$$

Les modèles  $m_A$  et  $m_B$  sont considérés comme équivalents si et seulement si :

$$m'_A \equiv m'_B \quad (8)$$

Donc, en considérant que  $MM_A \subseteq MM_{Int}$  et  $MM_B \subseteq MM_{Int}$ , nous obtenons la relation suivante :

$$m_A \mapsto^{\tau_A} m'_A \equiv m'_B \leftarrow^{\tau_B} m_B \quad (9)$$

La transformation bijective peut donc être rendue surjective, sans pour autant modifier l'apparence des modèles perçue par l'utilisateur,  $m_A$  et  $m_B$  restant inchangés. Notre approche pivot intervient au niveau de cette nouvelle équivalence.

Notre approche définit ces transformations  $\tau_A$  et  $\tau_B$  comme étant des fonctions de conformité constructive, le modèle construit suite à ces transformations étant le modèle pivot.

### 9.2.2 Fonctions de conformité constructive

Ces règles de transformations permettent de passer d'un modèle  $m_i$  ( $i$  pour BPA ou BPI) à un modèle pivot (noté  $m_{pivot}$ ). Son rôle est d'assurer une équivalence sémantique entre un modèle  $m_{BPA}$

avec un modèle  $m_{BPI}$ . Il doit également permettre une indépendance entre modèle cible et modèle de départ (modifier  $m_{BPA}$  sans se soucier du domaine SITI et de l'environnement d'implémentation associé ou modifier  $m_{BPI}$  sans se soucier du domaine métier et de son environnement d'analyse) et ainsi d'obtenir un couplage faible entre  $m_{BPA}$  et  $m_{BPI}$  tel que présenté section 6.4.5. Ce format intermédiaire devient alors nécessaire pour stocker et échanger les informations des modèles entre les environnements de modélisation et d'intégration.

Afin de garantir la cohérence des modèles lors de ces transformations, nous établissons des fonctions de conformité constructive (Favre, Estublier, & Blay-Fornarino 2006)  $f_{c_{BPA}}$  et  $f_{c_{BPI}}$  définies respectivement de  $MM_{BPA}$  et de  $MM_{BPI}$  vers  $MM_{pivot}$ .

Par définition, une fonction de conformité constructive définit les deux fonctionnalités suivantes :

- « fonction de conformité » : définit un ensemble d'opérations, *conforme*, tel que :

$$\forall m_i \in L(MM_i) \text{ si } conforme(m_i) \text{ est vraie} \quad (10)$$

- « fonction de conformité constructive » : définit une relation d'équivalence entre modèles selon leurs métamodèles :

Soit  $m_{BPA} \in L(MM_{BPA})$  et  $m_{BPI} \in L(MM_{BPI})$ , alors :

$$m_{BPA} \stackrel{s}{=} m_{BPI} \Leftrightarrow f_{c_{BPA}}(m_{BPA}) \equiv f_{c_{BPI}}(m_{BPI}) \quad (11)$$

Le  $s$  signifie la relation est considérée d'un point de vue sémantique. Cette typologie est détaillée section Figure 43. Ces fonctions de conformité constructives nous permettent d'obtenir le modèle pivot :

$$\exists m_{pivot} \in L(MM_{pivot}), f_{c_i}(m_i) = m_{pivot} \quad (12)$$

Avec  $i$  pour BPA ou BPI.

Grâce à ces fonctions, nous déduisons le lien permettant pour passer de l'environnement de modélisation à celui d'implémentation selon leurs modèles respectifs  $m_{BPA}$  et  $m_{BPI}$ , considérés comme sémantiquement équivalents. La Figure 43 positionne les différents modèles, métamodèles et les relations les liant. Ainsi si nous reprenons la relation 9 et utilisons les éléments évoqués dans ce paragraphe, nous obtenons bien le modèle pivot comme élément central et nécessaire permettant l'équivalence sémantique entre modèles d'analyse et d'implémentation:

$$m_{BPA} \xrightarrow{f_{c_{BPA}}} m_{pivot} \xleftarrow{f_{c_{BPI}}} m_B \quad (13)$$

Il convient de noter que si  $f_{c_i}(m_i) = m_{pivot}$  avec  $m_i \in L(MM_i)$  et  $m_{pivot} \in L(MM_{pivot})$ , il n'en demeure pas moins que  $MM_i \not\equiv MM_{pivot}$ . En effet, nous avons vu que chacun de ces métamodèles utilisent des vues et considèrent des aspects différents du processus.

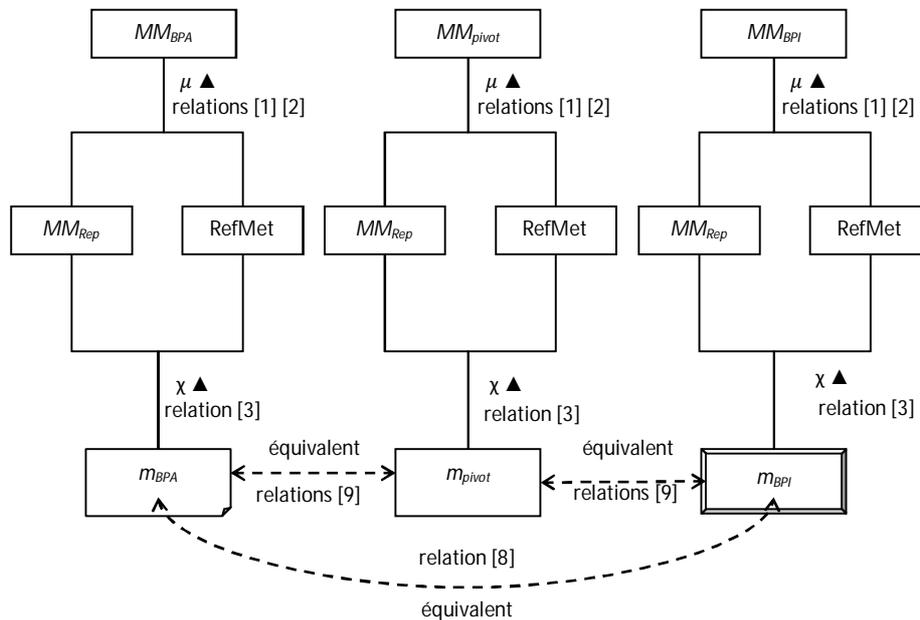


Figure 43. Modèles, Métamodèles, Métamodèle Pivot

### 9.3 TRANSFORMATIONS HORIZONTALES, TRANSFORMATIONS VERTICALES ET INTEROPERABILITE

Les approches MDA et BPM considèrent un ensemble de niveaux d'abstraction pour leurs modèles de processus. Des transformations s'opérant sur ces modèles peuvent en modifier l'abstraction pour les raffiner (modèle BPA → modèle BPI) ou pour les rendre plus abstraits (rétro-ingénierie). Ces transformations sont dites verticales et nous avons situé notre pivot au sein de cette transformation dans la section précédente (modèle BPA ↔ modèle pivot ↔ modèle BPI).

Il existe également des transformations dites horizontales. Celles-ci ne modifient pas le niveau d'abstraction d'un modèle. Elles sont utilisées pour restructurer ou compléter un modèle. Afin d'être indépendante des environnements de modélisation et d'implémentation, notre approche considère également ce type de transformation. En effet, si l'analyste métier désire changer d'environnement de modélisation, il opère une série de transformations horizontales sur ces modèles de processus afin de passer d'un environnement de modélisation A vers un environnement A' (Figure 44).

(Grangel et al. 2007) ont par exemple proposé un cadre de transformation permettant de transformer un modèle d'entreprise défini en langage GRAI en un modèle d'activité UML de même niveau d'abstraction CIM (« Computational Independent Model »).

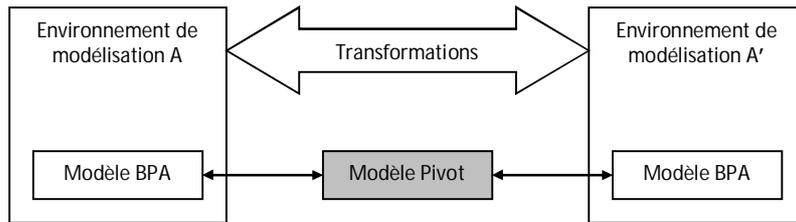


Figure 44. Environnements de modélisation et transformations horizontales

Pour obtenir cette interopérabilité, il faut considérer ces transformations entre modèles mais également l'alignement sémantique entre environnements. Différents travaux de recherche (Panetto et al. 2004) ; (Dassiti 2006) se penchent sur les possibilités d'échanger de manière transparente des données et modèles à un niveau sémantique. Par exemple, les concepts MDA ont été étendus afin de gérer les problèmes d'interopérabilité. Ainsi le projet ATHENA<sup>11</sup> propose un cadre d'interopérabilité orienté modèles ou *Model-Driven Interoperability*, - MDI et utilise une approche semblable à MDA.

Ces considérations sur l'interopérabilité des modèles de processus sont plus particulièrement étudiées au sein du réseau InterOP-VLab<sup>12</sup>. Il préconise entre autre l'emploi d'UEML (*Unified Enterprise Modelling Language*) (Anaya et al. 2008). UEML est un langage de modélisation conçu pour faciliter l'intégration de différents langages de modélisation d'entreprises. A l'inverse de notre approche où notre pivot adopte une généricité « relative », UEML adopte un aspect unificateur, représentant plusieurs langages de modélisation du niveau organisationnel de l'entreprise. Il est à cet effet reconnu comme l'unificateur sémantique de langages de modélisation comme GRAI, IEM et EEML. Ainsi, selon (Panetto 2007), l'interopérabilité offerte par UEML se situe essentiellement entre les CIM et PIM. A travers notre approche nous cherchons à atteindre une interopérabilité verticale approfondie, entre le PIM (« Platform Independent Model ») et le PSM (« Platform Specific Model »), autrement dit, entre modèles BPA et BPI. UEML est utilisé pour son aspect unificateur et est présenté en détail dans le manuscrit de thèse de (Baïna 2006).

## 9.4 METAMODELE ET METAMODELES PIVOT

Des efforts de standardisation ont été réalisés autour de la standardisation du workflow. (Zur Muehlen 2004) constate l'évolution représentée Tableau 3.

Tableau 3. Evolution de la standardisation du workflow

Année	1994	2004
Groupe de standardisation montrant un intérêt pour le workflow	1	Plus de 10
Type de standards	1 modèle de référence + 5 interfaces standards	Plus de 7 standards seulement pour les modèles de processus
Taille moyenne des spécifications	Environ 40 pages	Plus de 100 pages

11. <http://www.modelbased.net/mdi/index.html>

12. <http://interop-vlab.eu/>

La classification proposée par (Hollingsworth 2004) montre à quel point le nombre de standards s'intéressant plus particulièrement au BPM est élevé.

Nous cherchons à définir quels sont les standards typiques et les plus représentatifs du cycle BPM. Ils nous serviront ensuite à décrire les différents éléments constituant notre métamodèle pivot.

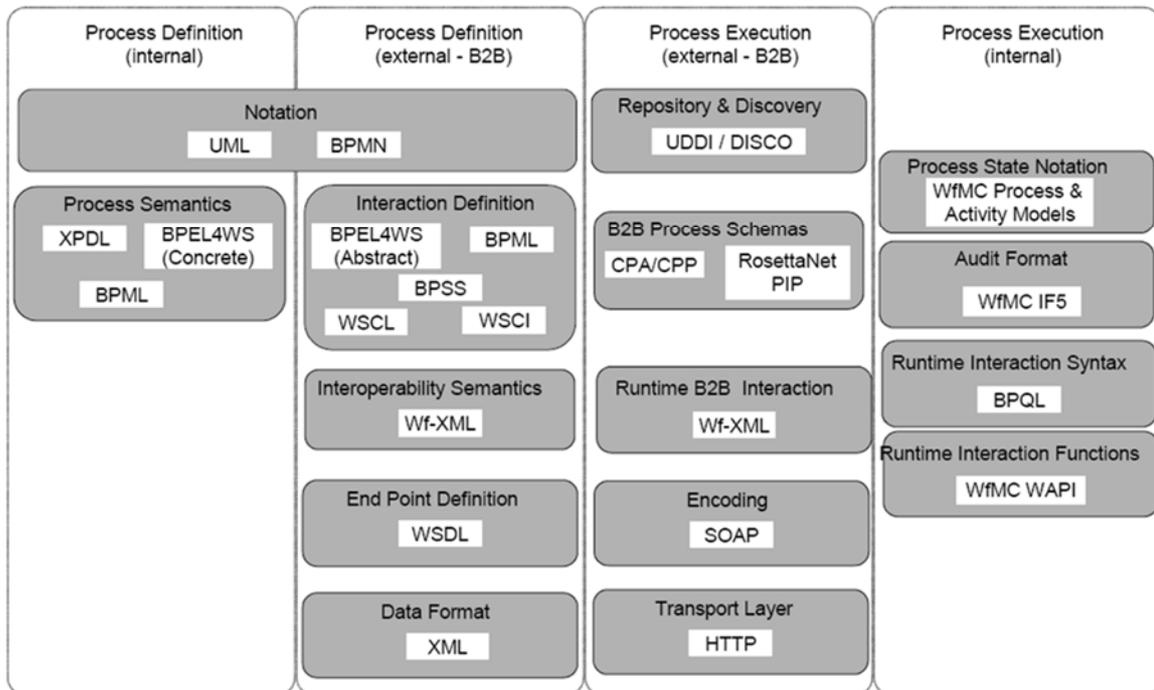


Figure 45. Classification des standards liés à BPM selon (Hollingsworth 2004)

Des efforts d'évaluations de langages de modélisation (List and Korherr 2006), de caractérisation de telles approches (Roser and Bauer 2005) ou encore la comparaison de ces langages et standards (Mendling et al. 2004) nous indiquent lesquels sont adéquats à notre approche (Tableau 4).

Tableau 4. Récapitulatifs des standards

Langages de modélisation	Standards Pivot	Langages d'exécution
UML AD, EPC, BPMN	BPDM, XPDL	BPEL, ebXML BPSS

Notre choix s'est porté sur BPMN, BPEL et XPDL, standards les plus représentatifs de leur catégorie selon (Zur Muehlen 2008). Nous allons en donner une brève description.

**BPMN** (Business Process Modelling Notation) est développé par l'OMG<sup>13</sup> (Object Management Group). BPMN fournit une notation graphique qui cible en premier lieu les analystes domaine et est supporté par plus de cinquante outils de modélisation. Très souple, BPMN est un langage dans lequel les nœuds d'action et de contrôle peuvent être connectés presque arbitrairement (Ouyang, Van der Aalst, Dumas, & Ter Hofstede 2006). Dans la forme présentée dans (OMG 2008), BPMN 1.2 n'est pas encore

<sup>13</sup> Voir [www.bpmn.org](http://www.bpmn.org)

capable de capturer tous les détails nécessaires à un traitement automatisé d'un modèle de processus au sein d'un moteur d'exécution BPM. Il lui manque la précision sémantique nécessaire lui permettant de capturer des processus métier implémentés. Une transformation depuis un modèle BPMN vers un langage exécutable devient alors nécessaire.

Développé dans ce sens, **BPEL** (Business Process Execution Language for Web Services) est un langage d'exécution définissant les processus complexes comme étant une composition de services web développé par OASIS<sup>14</sup>. BPEL 2.0 est essentiellement structuré en blocs et supporté par plusieurs plateformes d'exécution, il cible les développeurs logiciels. Dans l'état actuel, traduire des modèles BPMN en code est une étape nécessaire au regard d'un environnement de développement de processus métier basés sur les standards. Cette étape est d'autant plus compliquée car BPMN et BPEL représentent deux classes de langages différentes. La transformation BPEL-BPMN ne pose pas de réelles difficultés, la plupart des activités BPEL peuvent être transformées en BPMN (Weidlich et al. 2008). Cependant, nous assistons à des pertes de considérations de conception dans les modèles implémentés lors du passage BPMN-BPEL. De ce fait, le modèle obtenu après une transformation BPEL-BPMN est illisible pour un analyste métier (et donc inutile). Par exemple, l'absence de labels sur les activités et transitions et une répartition des éléments à travers le modèle différente de celui d'origine le rende difficile à lire.

**XPDL** (XML Process Definition Language) est un format standardisé par la WfMC (*Workflow Management Coalition*). La version actuelle, 2.1, a pour vocation de représenter tous les concepts de BPMN sous un format XML, de devenir son format d'échange standard<sup>15</sup>. Ainsi un mapping direct (d'éléments simples) est possible depuis un modèle BPMN vers un modèle XPDL. XPDL décrit la représentation graphique des éléments de manière textuelle (coordonnées 'XY', taille des nœuds...). Associé au BPMN, il permettrait à ce dernier de disposer d'un format d'échange de modèles et ainsi augmenter sa portabilité. Associé au sein d'une chaîne BPMN-XPDL-BPEL (Palmer 2006), il permettrait de conserver et restituer l'aspect original d'un modèle, même après sa transformation en BPEL.

Si BPEL sert d'exemple dans notre manuscrit et aide à caractériser les éléments du métamodèle pivot, XPDL et BPMN resteront les standards sur lesquels nous nous appuyerons pour développer notre plateforme Solution pour la Cohérence et l'ALignement des Processus (SCALP) (cf. chapitre 10).

#### REMARQUE : BPMN 2.0

---

Selon (Silver 2008) la version 2.0 n'a pas répondu à toutes les attentes mais a fourni des solutions concrètes sur un certain nombre de points. Les points nous intéressants le plus, vis-à-vis de notre approche sont les suivants :

- BPMN 2.0 permet-il une meilleure portabilité, facilitant les transformations entre environnements de modélisation et d'implémentation ?

BPMN 2.0 affiche la volonté de devenir un langage de modélisation exécutable et ainsi de se passer de BPEL. Néanmoins, le schéma d'échange standard basé sur XML proposé pour l'échange de modèles exécutables se révèle

---

<sup>14</sup> Voir [www.oasis-open.org](http://www.oasis-open.org)

<sup>15</sup> Voir <http://www.wfmc.org/xpdl.html>

insuffisant. De la même manière, l'exportation d'un modèle BPMN depuis un outil à un autre, se réalise non pas à l'aide d'un schéma XML standard mais avec une librairie nommée "M1 library". Ceci contribue à compliquer la portabilité du contenu du modèle avec des outils basés sur XML.

- BPMN 2.0 permet-il de spécifier des informations non-exécutables, relevant de la sémantique métier ?

BPMN ne propose pas de solutions permettant une distinction franche entre sémantique des modèles exécutables et non exécutables (BPMS.Info 2009). BPMN présume qu'essentiellement la sémantique des modèles exécutables et non exécutables est la même, en réalité. Hors ceci n'est pas forcément vérifié. Considérons une orchestration où, lorsqu'une activité A est terminée, l'activité B démarre immédiatement. Un moteur de workflow fonctionne traditionnellement de cette manière. Considérons maintenant que nous désirons, dans un modèle non-exécutable, pouvoir spécifier que l'activité B ne s'exécute pas nécessairement tout de suite après (cela arrive souvent à un niveau peu détaillé). Cet ajout d'information non visible dans le modèle n'est pas pris en compte par le BPMN 2.0. Ainsi l'information ne peut être transmise d'un outil à un autre.

Cette présentation succincte de BPMN 2.0 peut sembler très négative. Il faut cependant se rappeler qu'il ne s'agit pas d'une évaluation du standard. Nous cherchons juste à mettre en avant les caractéristiques manquantes à BPMN 1.2 et à BPMN 2.0 et ayant un impact sur notre approche.

---

## 9.5 DEFINITION DU METAMODELE PIVOT

Pour définir notre métamodèle pivot, nous commençons par déterminer les éléments le caractérisant. S'inspirant des trois standards décrits précédemment, ces éléments sont choisis selon deux critères : leur fréquence d'utilisation au sein de modèles de processus et leur appartenance ou non à une des classes de conformité du standard XPD.L.

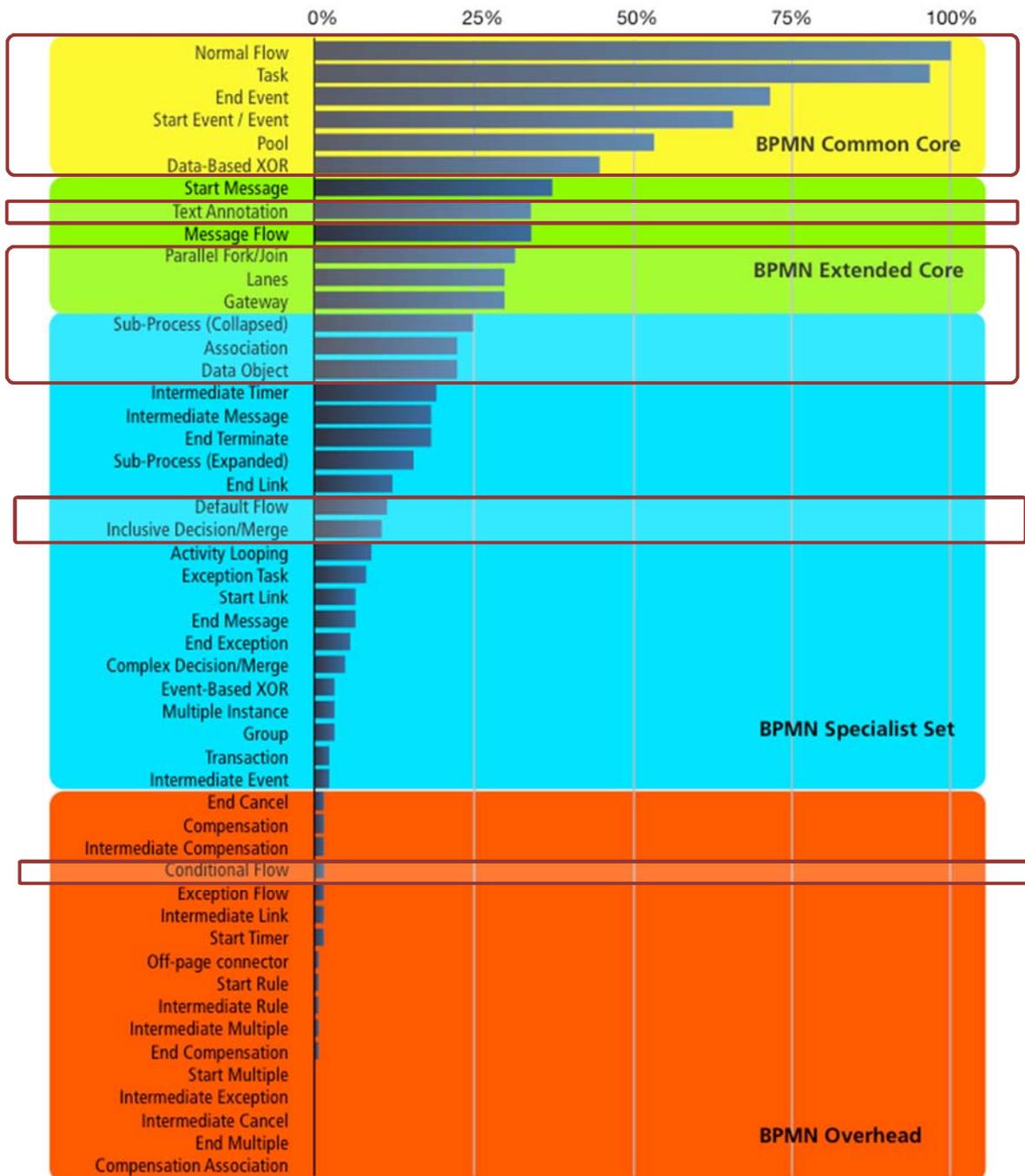
### 9.5.1 *Éléments de modélisation des processus métier*

Afin de permettre une représentation plus fine des processus métier, les langages de modélisation se sont complexifiés au fil des années. Cette complexité se justifie de la manière suivante : en détaillant plus finement un processus, nous facilitons sa transformation vers le modèle d'implémentation correspondant.

Cependant, l'effet est tout autre. La prolifération des concepts nuit à la compréhension et augmente la difficulté d'une transformation. (Zur Muehlen and Recker 2008) a réalisé une étude portant sur 126 modèles BPMN, issus de corps de métier hétéroclites. L'objectif de cette étude est de déterminer quels concepts définissant le vocabulaire de BPMN sont les plus utilisés, et à quelle fréquence. Cette fréquence est représentée sur le Tableau 5.

Sur les 50 concepts définissant le vocabulaire de BPMN 1.2, moins de 20% est en réalité utilisé. Seulement quatre éléments sont utilisés à plus de 50% par les analystes métier : l'activité de type tâche, la représentation du flux normal, les événements de début et de fin d'une séquence d'activité, la porte logique XOR (OU exclusif) et l'élément Pool (groupement).

Tableau 5. Fréquence d'occurrence des constructs BPMN



Notre approche utilise 18 objets qui constituent notre métamodèle pivot. Ils sont entourés sur le Tableau 5. Nous voyons qu'ils représentent l'ensemble du groupe *BPMN Common Core* et la majorité des éléments du groupe *BPMN Extended Core*. Il s'agit majoritairement des éléments les plus utilisés en entreprises. Pour plus de clarté, ces éléments sont catégorisés selon cinq groupes et sont reportés sur le Tableau 6. Nous réduisons ainsi l'expressivité du langage (Ulmer et al. 2008) en limitant son nombre d'éléments, réduisant notre champ d'étude et assurant une transformation des modèles de ce langage vers d'autres types de langages (comme des langages d'exécution) sans ambiguïté. En effet, les langages de modélisation utilisés ont évolué au cours du temps afin d'augmenter leur expressivité et d'être plus complets. Ces évolutions ont permis, dans une certaine mesure, une meilleure interprétation

de ces langages par des experts SITI. Néanmoins, ces évolutions ont rendu ces langages plus complexes, sans pour autant pallier les divergences sémantiques et structurelles existantes entre modèles conceptuels issus des langages de modélisation et modèles techniques provenant des langages d'implémentation. Ce nombre restreint d'éléments permet néanmoins de modéliser la plupart des processus rencontrés en industrie comme nous venons de le voir (Zur Muehlen & Recker 2008).

Certains éléments, bien qu'utilisés à 25 pour cent ou moins par les entreprises font partie de notre sélection. Ce choix d'éléments n'est pas anodin, l'ensemble des éléments choisis constituent la classe simple de conformité de portabilité de modèle ou *model portability conformance class*. Ces classes, définies dans (WfMC, 2008) sont au nombre de trois (Simple, Standard, Complète). Un outil de modélisation certifiant appartenir à l'une de ces classes doit pouvoir importer et comprendre chacun des éléments constituant cette classe. Nous étendons cette définition à notre étude, dans laquelle un outil BPI doit pouvoir importer et comprendre un modèle BPA et un outil BPA un modèle BPI. Chacun de ces modèles doit également posséder un modèle pivot équivalent. La Figure 46 illustre les concepts utilisés dans notre démarche et leurs relations structurelles à l'aide d'un diagramme de classe, ici en Ecore.

Tableau 6. Eléments constituant le métamodèle pivot

N°	Famille d'éléments	Type		N°	Famille d'éléments	Type
1	Node	Event	Start	11	Edge	Uncontrolled
2			End	12		Conditional
3		Action	Task	13		Default
4			Activity	14		Association
5			Sub-Process	15	Objet data	
6		Logical	Exclusive	16	Artefact	Annotation
7			Inclusive	17	Special	BPElement
8			Parallal	18	Process	
9	Swimlane	Pool				
10		Lane				

Le langage BPMN étant largement répandu, nous ne fournissons pas une présentation exhaustive de tous ses éléments. La thèse de (Touzi 2007) présente en détail les différents éléments de ce langage et (Silver 2009) en définit les bonnes pratiques.

Chacun des éléments de notre métamodèle pivot possède les attributs suivants :

Tableau 7. Attributs communs à tous les éléments

Nom de l'attribut	Type	Description
name	string	Nom de l'élément
id	string	Id de l'élément (unique)
hasSpecialProperties	BPElement	Définit les propriétés spécifiques de l'élément

Dans la suite de ce chapitre, nous définissons les éléments et les attributs spécifiques qui y sont rattachés.

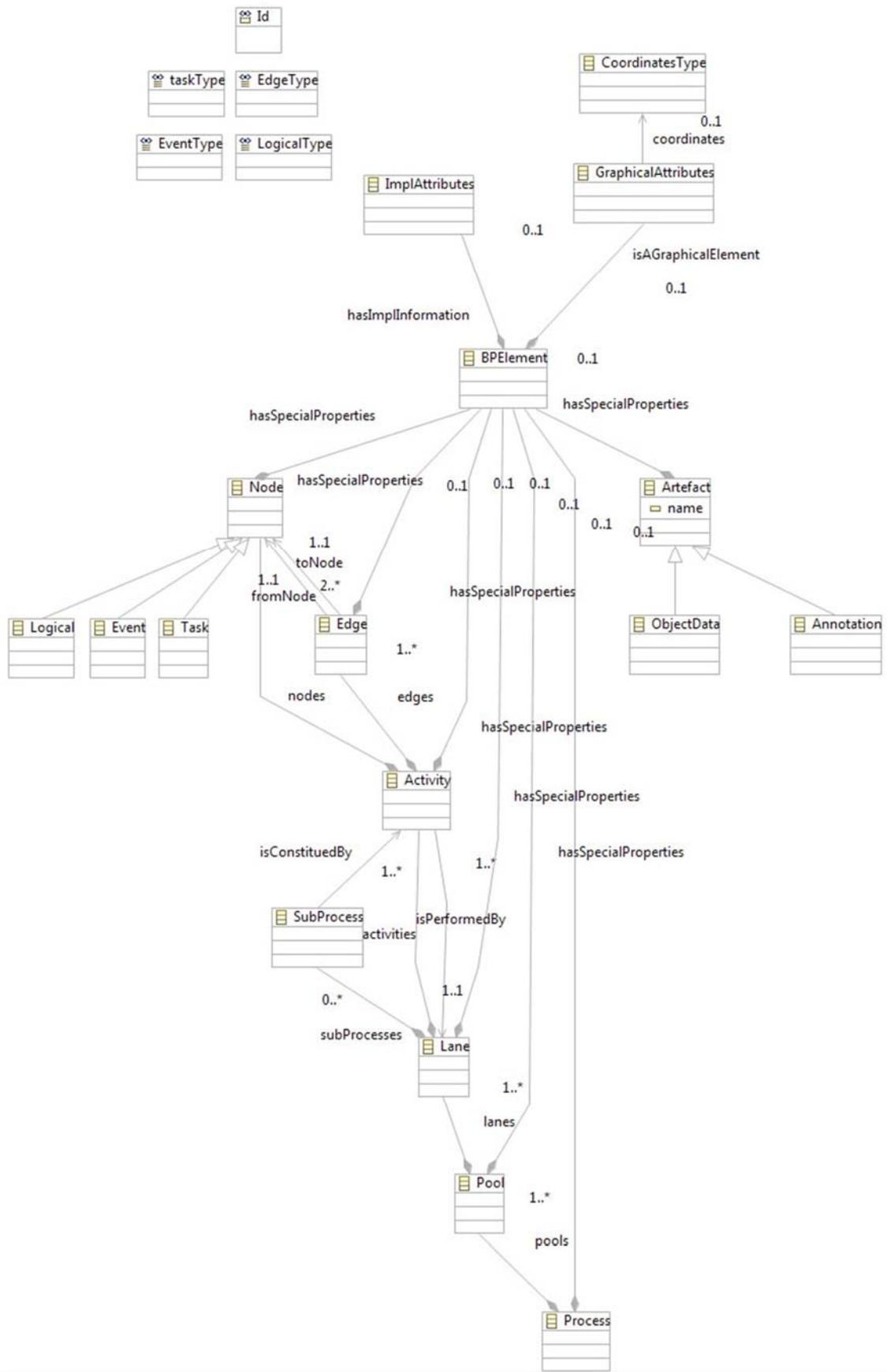


Figure 46. Métamodèle pivot

### 9.5.2 Famille d'éléments : Special

Ce groupe rassemble deux éléments supplémentaires par rapport à ceux choisis : *Process* et *BPElement*. L'élément *Process* est l'objet englobant l'ensemble des autres éléments exprimés dans le Tableau 6. Dans notre approche, nous ne considérons qu'un processus par modèle. Ainsi cet élément *Process* peut être assimilé au diagramme de processus métier, *Business Process Diagram - BPD* (Tableau 7).

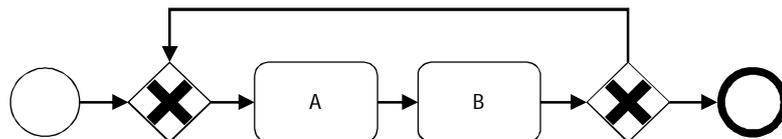
Tableau 8. Attribut de l'élément *Process*

Nom de l'attribut	Type	Description
pools	Pool	Indique les éléments <i>Pool</i> contenus dans l'élément

Le deuxième élément est la classe *BPElement*. Cette classe permet de prendre en considération des informations n'influençant pas le modèle BPA et/ou le modèle BPI mais jugées nécessaires dans un but de rétro-ingénierie des processus.

#### *BPElement*

Reconsidérons les modèles de l'exemple vus au chapitre 7. Lors de la transformation du modèle (a) vers le modèle (b), les informations graphiques n'étaient pas considérées. Dans l'approche que nous proposons, la transformation ne se fait pas directement, nous passons par un modèle intermédiaire, le modèle pivot.



(a)

```

...
<repeatUntil ...>
  <sequence>
    <invoke name="A"
      parterLink="LinkA" .../>
    <invoke name="B"
      parterLink="LinkB" .../>
  </sequence>
  <condition>...</condition>
</repeatUntil>
...

```

(b)

Figure 47. Modèle BPA et modèle BPI

A l'aide de la classe *BPElement*, nous conservons les données relatives à l'aspect graphique du modèle BPA et de manière plus générale, relatives aux notions de métier exprimées dans le modèle BPA et non prises en compte dans le modèle BPI. Considérons l'extrait du modèle pivot Figure 48.

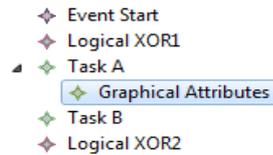


Figure 48. Extrait du modèle pivot

La tâche A possède les données retranscrites dans le modèle pivot en tant que *GraphicalAttributes* (Figure 49). Nous opérons de la même manière lors de la transformation inverse (modèle BPI vers le modèle BPA), les données inutilisées par l'analyste métier étant retranscrites en tant que *ImplAttributes*.

Les attributs de *BPElement* sont définis par l'architecte des processus en accord avec l'analyste métier et l'expert SITI (Tableau 8). La présence de ces deux experts est requise afin de déterminer les données qui sont utiles et communes aux deux modèles et les données qui sont spécifiques à chacun de ces modèles.

Property	Value
Attributes	
borderColor	-1
businessData	none
coordinates	("32", "96")
height	17
isVisible	true
laneId	default_lane1
shape	2
width	42

Figure 49. Propriétés graphiques de l'élément « Task A »

Tableau 9. Attribut de l'élément *BPElement*

Nom de l'attribut	Type	Description
isAGraphicalElement	GraphicalAttributes	Définit les éléments graphiques de l'élément (coordonnées, forme,...)
hasImplInformation	ImplAttributes	Définit les éléments utilisés lors de l'exécution du processus (import, type de données,...)

### 9.5.3 Groupe Node

Cette famille regroupe les composants de modélisation décrivant le *control-flow* du processus. Chacun de ces éléments peut être perçu comme un nœud (par opposition au terme lien) et peut posséder un lien entrant ou sortant. Ce groupe se subdivise en trois sous-types : les événements, les actions et les éléments logiques.

#### Évènement

Les événements décrivent un événement qui se produit durant l'exécution du processus. Ces événements affectent le *control-flow* du processus et possèdent usuellement une cause (déclencheur) ou un impact (résultat). Dans notre cas, nous considérons qu'un processus débute par un *Start Event* et se termine avec un *End Event*. Ces deux éléments sont représentés par l'élément *Event* du métamodèle pivot. L'attribut *eventTypes* est une énumération indiquant si l'évènement ciblé est de type *start* ou de type *end* (Tableau 9).

Tableau 10. Attribut de l'élément Event

Nom de l'attribut	Type	Description
eventTypes	Énumération	Définit le type de l'élément (0 : start ; 1 : end)

*Action*

Les nœuds de type action représentent les éléments réalisant une activité, une tâche ou un ensemble de ces derniers durant le déroulement du processus.

L'élément *Task* représente une activité atomique, signifiant qu'elle ne peut être représentée comme un ensemble de nœuds dans ce diagramme de processus. Le métamodèle pivot réalise la distinction entre une *User task* réalisée par une personne et une *Service task* réalisée de manière automatisée. Cette distinction est faite à l'aide de l'attribut *taskTypes* (Tableau 10).

Tableau 11. Attribut de l'élément Task

Nom de l'attribut	Type	Description
taskTypes	Énumération	Définit le type de l'élément (0 : none ; 1 : user task ; 2 : service task)

L'élément *Activity* regroupe un ensemble de tâches - *Task* reliées entre elles par des liens - *Edge* (Tableau 11).

Tableau 12. Attribut de l'élément Activity

Nom de l'attribut	Type	Description
nodes	Node	Indique les éléments <i>Node</i> contenus dans l'élément
edges	Edge	Indique les éléments <i>Edge</i> contenus dans l'élément
isPerformedBy	Lane	Indique l'élément <i>Lane</i> le contenant

Et l'élément *Sub-Process* regroupe un ensemble d'activités et de liens (Tableau 12).

Tableau 13. Attribut de l'élément SubProcess

Nom de l'attribut	Type	Description
isConstituedBy	Activity	Indique les éléments <i>Activity</i> contenus dans l'élément
hasEdges	Edge	Indique les éléments <i>Edge</i> contenus dans l'élément

*Logical*

Une porte logique représente un point de contrôle dans le *sequence-flow* du processus. Ces portes permettent de séparer ou de rassembler les multiples flux d'un processus sous certaines conditions. Dans notre métamodèle, ces portes sont représentées par l'élément *Logical* et l'attribut *logicalTypes* permet de faire la distinction entre les types OU, ET et OU Exclusif (Tableau 13).

Tableau 14. Attribut de l'élément Logical

Nom de l'attribut	Type	Description
logicalTypes	Énumération	Définit le type de l'élément (0 : AND ; 1 : OR ; 2 : XOR)

9.5.4 *Groupe Edge*

Ce groupe décrit les différents éléments permettant le *sequence-flow* du processus. Ces éléments représentent les liens existants entre les nœuds et avec les artefacts. Dans notre métamodèle pivot, un seul élément permet de tous les représenter, l'élément *Edge*. L'attribut *edgeType* permet de définir quel type de lien est représenté par l'élément *Edge* (Tableau 14):

- (0) Uncontrolled : le lien est franchi dès que possible (si le nœud source est exécuté) ;
- (1) Conditional : le lien est franchi dès que possible et si la condition portée sur la transition est vraie ;
- (2) Default : ce lien est franchi par défaut ;
- (3) Association : ce lien permet de relier un artefact à un autre élément du diagramme.

Tableau 15. Attribut de l'élément Edge

Nom de l'attribut	Type	Description
toNode	Node	Indique le noeud-cible de l'arc
fromNode	Node	Indique le noeud-source de l'arc
edgeType	Enumération	Définit le type de l'élément (0 : Uncontrolled ; 1 : Conditional ; 2 : Default ; 3 : Association)

### 9.5.5 Groupe Swimlane

Ce groupe rassemble les éléments utilisés pour définir la typologie des participants.

#### Pool

L'élément Pool est le contenant d'un processus. Techniquement, il représente un participant (un rôle ou une entité métier) et contient une ou plusieurs *Lanes* (Tableau 15).

Tableau 16. Attribut de l'élément Pool

Nom de l'attribut	Type	Description
lanes	Lane	Indique les éléments <i>Lane</i> contenus dans l'élément

#### Lane

Une *Lane* est une sous-division du processus et représente le rôle d'un exécutant ou d'une unité organisationnelle du processus. Elle contient une ou plusieurs activités et sous-processus (Tableau 16).

Tableau 17. Attribut de l'élément Lane

Nom de l'attribut	Type	Description
activities	Activity	Indique les éléments <i>Activity</i> contenus dans l'élément
subProcesses	SubProcess	Indique les éléments <i>SubProcess</i> contenus dans l'élément

#### REMARQUE : ELEMENT POOL

Bien que l'élément « pool » soit présent, cette approche ne prend pas en considération, pour le moment, la chorégraphie entre processus de collaboration.

### 9.5.6 Groupe Artefact

Les éléments de ce groupe sont atypiques par rapport aux autres présentés précédemment. Ils ne possèdent pas de sémantique clairement définie. L'*ObjectData* peut être utilisé pour représenter une donnée ou un document présent dans le flux entre les activités et les événements d'un processus. L'*Annotation* est un texte arbitraire qui peut être inséré dans le diagramme de processus et être relié à un autre élément.

## 9.6 CARACTERISATION DES LIENS SEMANTIQUES

Pour identifier les liens sémantiques existants entre les éléments de deux modèles, nous nous appuyerons sur les définitions d'équivalences sémantiques issues de (Rizopoulos et Mçbrien 2005).

### 9.6.1 Typologie des liens sémantiques

Les auteurs considèrent que les relations sémantiques sont définies selon les domaines intentionnels (notés  $Di(A)$  et  $Di(B)$ ), c'est-à-dire que les objets du monde réels sont associés aux concepts des modèles A et B. Nous avons les quatre règles suivantes :

- Equivalence : les concepts de A et B sont équivalents si et seulement si

$$A \stackrel{s}{=} B \text{ ssi } Di(A) = Di(B) \quad (14)$$

- Subsumption ou généralisation : A est une subsumption de B si et seulement si

$$B \stackrel{s}{\subset} A \text{ ssi } Di(B) \subset Di(A) \quad (15)$$

A l'opposé, B est une spécialisation de A (ou subsumption inverse).

- Intersection : les concepts de A et B s'entrecroisent si et seulement si

$$A \stackrel{s}{\cap} B \text{ ssi } Di(A) \cap Di(B) \neq \emptyset, \exists C : Di(A) \cap Di(B) = Di(C) \quad (16)$$

- Disjonction : A et B sont disjoints si et seulement si

$$A \stackrel{s}{\not\cap} B, \exists C : Di(A) \cup Di(B) \subseteq Di(C) \quad (17)$$

Le « s » présent sur les symboles désigne le fait que les éléments sont considérés d'un point de vue purement sémantique. A l'aide de ces relations, nous pouvons identifier les liens sémantiques existants entre les éléments constituant notre métamodèle pivot et ceux des métamodèles BPA et BPI.

Ces relations seront utilisées afin de déterminer les équivalences sémantiques existantes entre les différents métamodèles manipulés par notre plateforme SCALP (Chapitre 10). Avant cela, nous présentons dans la section suivante un exemple d'application de ces relations avec les standards les plus représentatifs des processus métier : BPMN, BPEL et XPDL.

### 9.6.2 Exemple d'équivalence sémantique

Afin d'illustrer l'utilisation des liens sémantiques présentés précédemment, nous utilisons le triptyque suivant : <BPMN, XPDL, BPEL>. Le Tableau 18 montre les équivalences sémantiques existantes entre les différents éléments de ces métamodèles.

Le « one-to-one mapping » est réalisable entre éléments du standard BPMN et ceux du standard XPDL. Nous remarquons que les éléments de la famille "Swimlane" ne sont pas pris en compte par BPEL. En effet, l'élément « Pool » n'a pas son équivalent sémantique dans les langages d'exécution. De

manière identique, l'élément « Lane » n'a pas d'équivalent sémantique dans les langages d'exécution. Son rôle étant d'indiquer comment les activités sont groupées au sein du processus métier, il est obsolète dans un langage de modélisation.

Tableau 18. Exemples d'équivalence sémantique entre BPMN, XPDL et BPEL

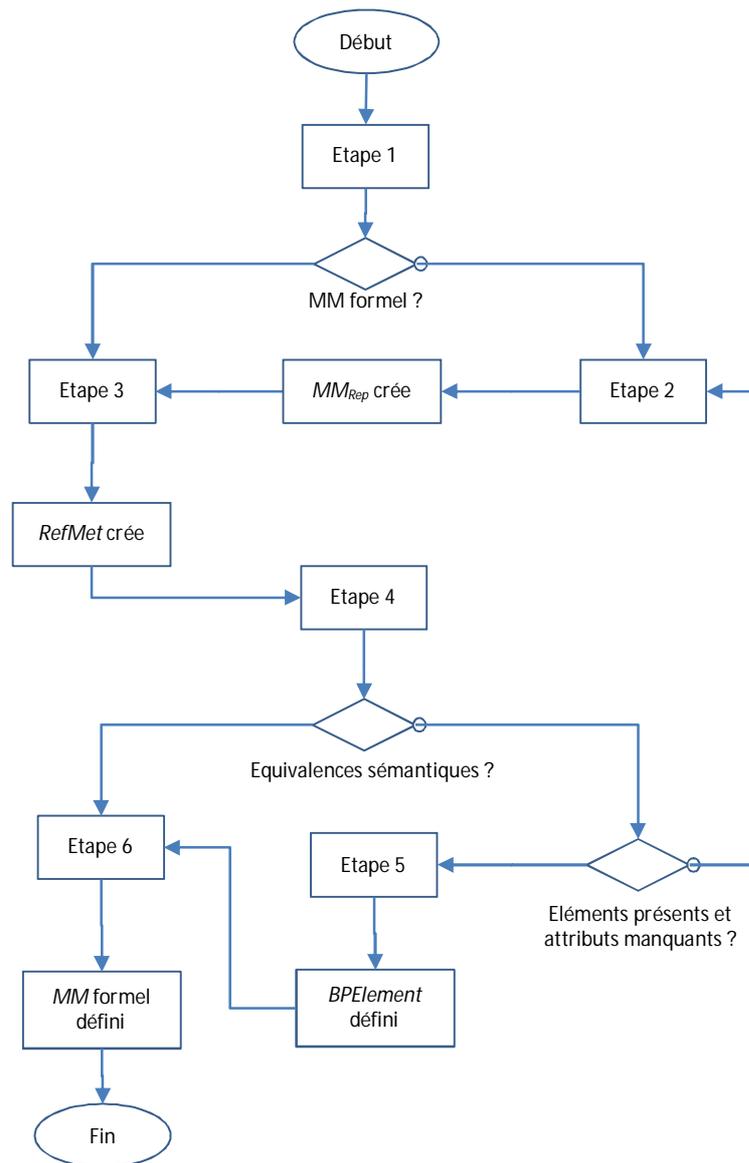
Eléments BPMN (A)	Relation sémantique	Eléments XPDL (B)	Relation sémantique	Eléments BPEL (C)
None Start	$A = B$	Start (None)	$C \subset B$	{receive, pick}
Collapsed Sub-Process	$A = B$	Activity Set/ Block Activity	$B = C$	scope
Parallel Gateway	$A = B$	Route (Join OR/Join AND), (Split AND)	$B \subset C$	flow
Pool	$A = B$	Pool		Aucune correspondance
Lane	$A = B$	Lane		Aucune correspondance
Data Object	$A = B$	Artifact (DataObject)	$B \cap C$	documentation
Text Annotation	$A = B$	Artifact (Annotation)	$B \cap C$	documentation

## 9.7 FORMATION DES METAMODELES BPA ET BPI

Dans ce paragraphe, nous allons définir d'une manière très séquentielle la formation d'un métamodèle  $MM_i$ . Nous supposons qu'un langage de modélisation ou d'une application SI permettant l'exécution des processus, selon nous concevons un métamodèle BPA ou BPI, est disponible.

Maintenant, détaillons la création du métamodèle étapes par étapes, retranscrites Figure 50 :

- Etape 1. Le métamodèle  $MM_i$  existe-t-il ? S'il en existe une représentation formelle, alors nous pouvons passer à l'étape 3. Dans les autres cas, il faut considérer l'étape 2.
- Etape 2. Nous commençons par effectuer une comparaison entre éléments appartenant au langage de modélisation. Nous constituons le  $MM_i$  en nous inspirant du  $MM_{Pivot}$ , en identifiant les éléments communs aux deux métamodèles. Le  $MM_{Rep}$  est ainsi créé.
- Etape 3. Nous cherchons maintenant à déterminer le  $RefMet$  du métamodèle. Nous définissons les règles non-structurelles devant s'appliquer aux différents éléments déterminés durant l'étape précédente.
- Etape 4. Puis nous procédons à l'identification des éléments appartenant au  $MM_i$  ayant un équivalent sémantique au sein du  $MM_{Pivot}$ . Les règles d'équivalences exposées section 9.6 sont utilisées à cet effet. Si tous les éléments du  $MM_i$  ont un équivalent, alors nous pouvons poursuivre avec l'étape 6. Dans le cas contraire, il faut se poser la question suivante : s'agit-il d'un problème d'attributs ? Si c'est le cas, alors il faut considérer l'étape 5. Sinon, il faut reprendre à l'étape 2.
- Etape 5. Ici, l'élément  $BPElement$  du  $MM_{Pivot}$  est utilisé pour définir les attributs manquants d'un élément. L'élément  $BPElement$  appartient au  $MM_{Pivot}$ . Il permet de rajouter des attributs de type BPA ou BPI à un élément quelconque du  $MM_{Pivot}$ .
- Etape 6. Le métamodèle est correctement formé. Nous pouvons dès lors constituer notre plateforme solution permettant la transformation d'un modèle issu d'un environnement vers un autre. Tout ceci est expliqué dans la quatrième partie : Mise en œuvre.

Figure 50. Formation d'un  $MM_i$ 

## 9.8 CONCLUSION

A l'aide de concepts issus de l'IDM, nous pouvons justifier les relations de cohérence entre modèles. En appliquant la méthode présentée dans la section précédente, nous systématisons l'utilisation de métamodèles BPA et BPI dans notre approche. Afin d'atteindre un alignement opérationnel, il nous faut au préalable obtenir une transformation bidirectionnelle entre modèle BPA et modèle BPI. Nous avons à cet effet présenté la nécessité de passer par un système intermédiaire, le modèle pivot. Son métamodèle est défini selon deux critères : selon la classe simple de conformité proposée par le standard XPDL et selon les éléments les plus utilisés au sein des modèles de processus métier.

Les transformations utilisées dans l'approche proposée permettant d'assurer la synchronisation et la cohérence intermodèle, sont justifiées à l'aide de fonctions de conformité constructive. De part la complexité en résultant, l'utilisation d'une telle approche n'est pas indiquée dans des démarches de

transformations unilatérales, d'un domaine vers un autre. Cette approche devient réellement utile dans des démarches de rétro-ingénierie, ou pour conserver un modèle d'analyse « à jour ».

# QUATRIEME PARTIE

---

## D. MISE EN ŒUVRE

10.	Plateforme Solution pour une Cohérence et un ALignement des Processus	121
11.	Démonstration	137
12.	Ingénierie des processus au service de l'ingénierie des procédés	157

---

## RESUME

---

Pour confirmer notre approche théorique évoquée dans la partie précédente, nous menons un projet de développement logiciel. Le résultat de ce projet, la plateforme Solution pour la Cohérence et l'Alignement des Processus – SCALP, est décrit. Puis nous présentons un cas d'étude afin de mettre en œuvre notre approche. Un processus utilisé en industrie est manipulé depuis sa modélisation jusqu'à sa transformation en module fonctionnel pour un logiciel de type ERP. Nous présentons aussi nos travaux de recherche relatifs à la possible adéquation entre l'ingénierie des processus et l'ingénierie des procédés. Nous pensons qu'une telle adéquation est possible à travers l'utilisation de notre approche pivot, et nous cherchons à le démontrer.

---

---

## Plateforme Solution pour une Cohérence et un ALignement des Processus - SCALP

---

Afin de valider les différentes propriétés proposées tout au long de ce manuscrit, nous présentons un prototype logiciel mettant en œuvre et validant notre approche. L'objectif de ce prototype est de permettre concrètement la cohérence des modèles issus du domaine métier et du domaine SITI et améliorer ainsi l'alignement opérationnel entre processus d'entreprise et système d'information.

## 10.1 ARCHITECTURE GENERALE DE LA PLATEFORME SCALP

Cette plateforme solution est basée sur les trois environnements pris en compte dans notre approche :

- Un environnement de modélisation, où un modèle conceptuel de processus est dessiné par un analyste métier. La fonctionnalité découlant de cet environnement est la modélisation du processus métier. A cet effet, nous utilisons l'outil de modélisation Intalio Designer 6.0.3<sup>16</sup>;
- Un environnement d'implémentation, comprenant la plateforme cible d'exécution. C'est au sein de cet environnement que l'expert SITI modifie le modèle d'entrée obtenu, de manière à le rendre exécutable. Le moteur d'exécution de processus utilisé est un ERP, OpenERP 5.0.14<sup>17</sup> ;
- Un environnement pivot, contenant la plateforme pivot. Cette plateforme comporte deux fonctionnalités bien distinctes : la transformation d'un modèle provenant d'un environnement (le modèle d'entrée) vers un autre (le modèle de sortie) et la préservation de l'intégrité des informations contenues dans le modèle d'entrée. Pour instancier cette plateforme pivot, nous utilisons le framework EMF 1.4.0 (Eclipse Modelling Framework)<sup>18</sup> de l'IDE (Integrated Development Environment) Eclipse 3.4.2<sup>19</sup>. La métamodélisation est réalisée à l'aide d'Ecore 0.7.0 et est appuyée par le langage Kermeta<sup>20</sup> 1.3.0 (« Kernel Metamodeling »).

La plateforme SCALP et les environnements considérés sont représentés Figure 51. Si le prototype obtenu repose sur des outils, l'approche mise en œuvre reste indépendante des outils technologiques utilisés. Nous décrivons maintenant les différents outils composant notre plateforme et les environnements.

## 10.2 ENVIRONNEMENT PIVOT

### 10.2.1 Eclipse Modeling Framework

EMF, dont l'interface est exposée Figure 52, est une plateforme de modélisation et de génération de code pour la construction d'outils et d'autres applications basées sur une structure de modèle de données (Steinberg et al. 2009).

EMF est donc une partie intégrante de la plate-forme Eclipse, permettant l'utilisation des technologies et cadres comme l'éditeur Eclipse Visual, SDO, UML ou encore XSD. EMF est également développé pour supporter les caractéristiques de la technologie Java, telles que les types énumérés, annotations, etc.

---

<sup>16</sup> <http://www.intalio.com/bpms/designer>

<sup>17</sup> <http://www.openerp.com/>

<sup>18</sup> <http://www.eclipse.org/emf/>

<sup>19</sup> <http://www.eclipse.org/>

<sup>20</sup> <http://www.kermeta.org/>

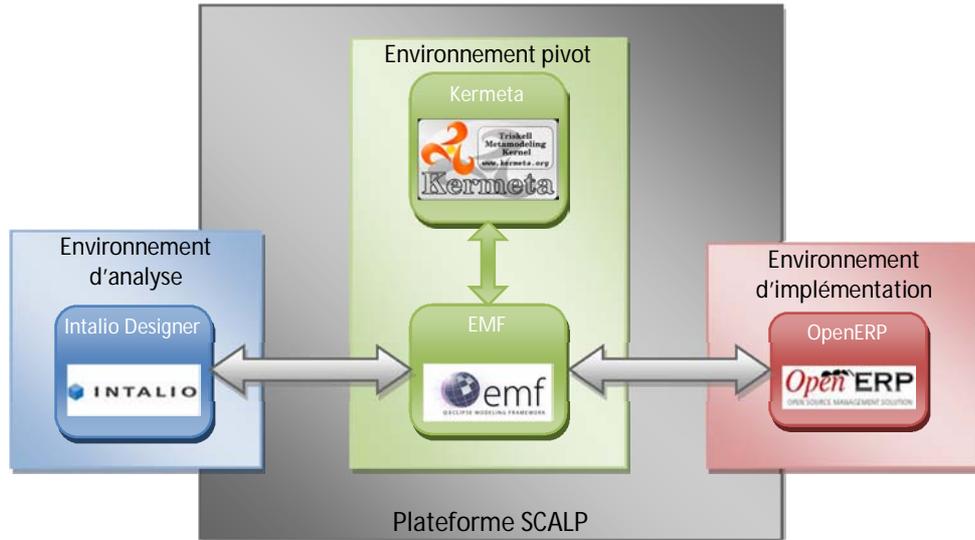


Figure 51. Plateforme SCALP

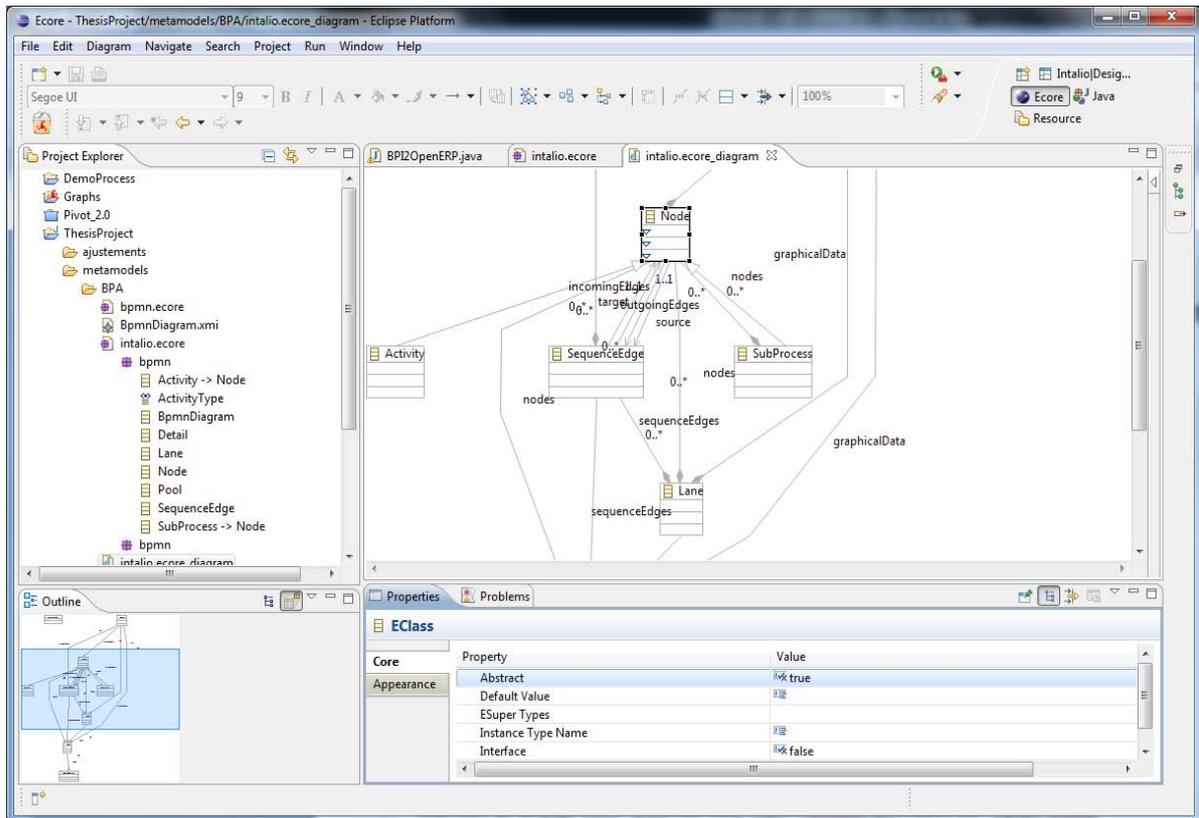


Figure 52. Interface EMF

Développé par la fondation Eclipse, EMF s'appuie sur un méta-méta-modèle : Ecore. Ecore est basé sur MOF (« Meta-Object Facility ») 2.0 et est identique à l'EMOF (« Eclipse Meta-Object Facility »), tous deux présentés section 5.2.2. Ecore permet de décrire les modèles en représentant leurs métamodèles ainsi qu'une sérialisation des modèles en XMI.

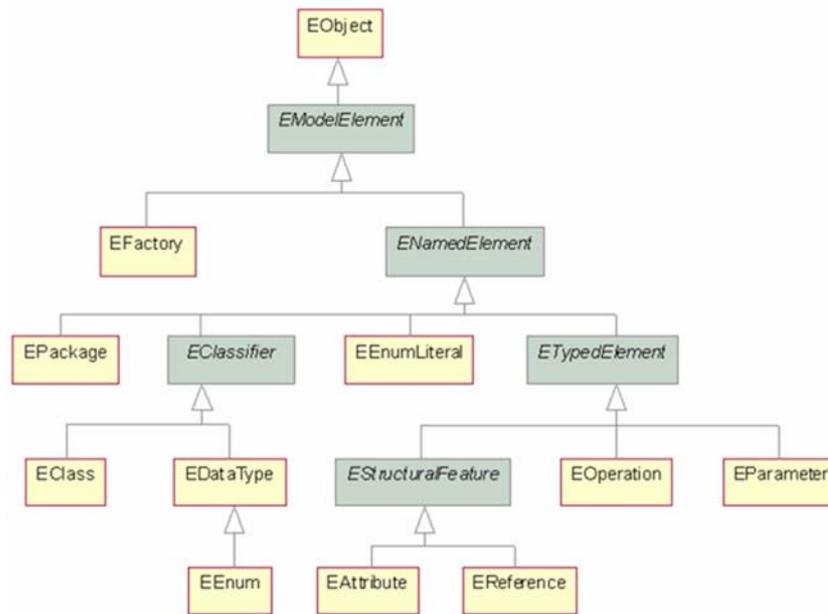


Figure 53. Métamodèle Ecore

L'objectif d'EMF est de fournir un ensemble d'outils permettant la manipulation de modèles et non le comportement. Pour cela, la métamodélisation, définie à l'aide des outils Ecore, est complétée par l'utilisation de Kermeta.

### 10.2.2 Kermeta

La métamodélisation se fait à l'aide d'Ecore et est appuyée par la plateforme open-source de métamodélisation Kermeta (« Kernel Metamodeling »). Le langage Kermeta est une extension du langage EMOF (Essential Meta-Object Facilities) développée par l'équipe Triskell<sup>21</sup>. Il s'agit d'une extension nécessaire car les langages de métamodélisation tel que EMOF sont uniquement en mesure de modéliser des structures comme des classes, attributs... Le langage Kermeta permet de définir la sémantique opérationnelle et dénotationnelle d'un métamodèle et le comportement des structures grâce à son langage d'action. Dénotationnelle car elle peut décrire les relations entre différents types d'éléments appartenant à des formalismes différents (Muller, Fleurey, & Jézéquel 2005). Ce langage définit également des mécanismes de liaison dynamique et de gestion des exceptions et des structures de contrôle classiques en langage impératif. L'une des caractéristiques clés de Kermeta est sa capacité à étendre un métamodèle existant avec des contraintes, de nouveaux éléments structuraux (méta-classes, classes, propriétés, et opérations) ou encore des fonctionnalités définies avec d'autres langages. Il s'agit du tissage d'aspect (Klein and Fleurey 2006). Ce tissage permet d'ajouter du code (une simple classe ou un élément d'un langage spécifique) au sein d'un métamodèle cible sans pour autant en modifier la structure (Mosser and Blay-Fornarino 2009).

<sup>21</sup> [https://www.irisa.fr/triskell/software-fr/kermeta/index\\_html](https://www.irisa.fr/triskell/software-fr/kermeta/index_html)

Tout ceci contribue à rendre le langage Kermeta comme étant un langage particulièrement adapté à la définition de transformations de modèles ou des contraintes s'établissant sur les éléments de ces derniers. Kermeta permet de décrire les transformations ainsi que les règles métier de manière impérative. Les règles métier seront décrites selon le standard SBVR (Semantic Business Vocabulary and Business Rules).

### 10.3 ENVIRONNEMENT DE MODELISATION

L'outil BPA est représenté par l'application Intalio Designer 6.0.2<sup>22</sup>. Cet éditeur permet de modéliser des modèles de processus en utilisant une palette de symboles BPMN. Ainsi cet outil respecte globalement la spécification du langage BPMN 1.2. Intalio Designer génère deux fichiers exploitables et libres au format XML. Prenons par exemple le diagramme représenté Figure 54.

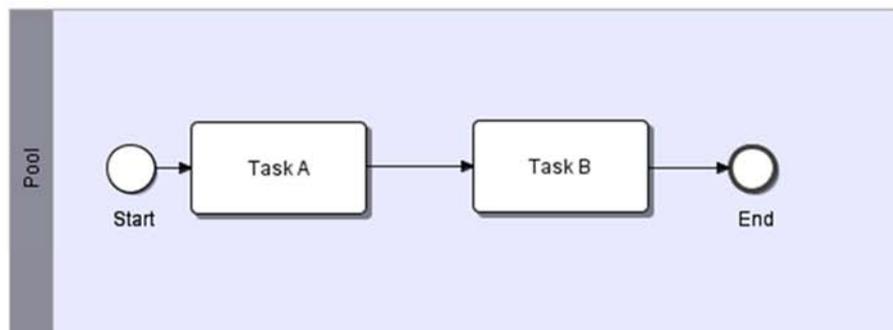


Figure 54. Diagramme Intalio Designer

Intalio intègre une fonction de formatage de source afin de la rendre accessible à un éditeur XML. Suite à cette action nous obtenons deux fichiers en sortie, le premier fichier détaille la logique du processus (Figure 55), le second comporte les données relatives à l'aspect graphique du processus (Figure 56). Ces deux fichiers sont le point de départ de notre plateforme.

```

<?xml version="1.0" encoding="UTF-8"?>
<bpmn:BpmnDiagram xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
  xmlns:bpmn="http://stp.eclipse.org/bpmn" xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore"
  xmi:id="_LbCiAc8uEd-egYv4lhUmSv" id="_LbCiAM8uEd-egYv4lhUmSv">
  <pools xmi:type="bpmn:Pool" xmi:id="_LbVc8c8uEd-egYv4lhUmSv" id="_LbVc8M8uEd-egYv4lhUmSv"
    name="Pool">
    <eAnnotations xmi:type="ecore:EAnnotation" xmi:id="_Xd1G0M8uEd-egYv4lhUmSv"
      source="executablepool">
      <details xmi:type="ecore:EStringToStringMapEntry" xmi:id="_Xd1G0c8uEd-egYv4lhUmSv"
        key="poolIsExecutable" value="false" />
    </eAnnotations>
    <vertices xmi:type="bpmn:Activity" xmi:id="_LbVc888uEd-egYv4lhUmSv"
      id="_LbVc8s8uEd-egYv4lhUmSv" outgoingEdges="_RE_NEc8uEd-egYv4lhUmSv"
      incomingEdges="_Oc5Vgc8uEd-egYv4lhUmSv" name="Task A" activityType="Task" />
  </pools>
  </bpmn:BpmnDiagram>
  
```

Figure 55. Extrait du fichier d'entrée « Description logique »

<sup>22</sup> <http://www.intalioworks.com/products/bpm/opensource-edition/designer/>

```

<children xmi:type="notation:Node" xmi:id="_RE2DI8uEd-egYv4lhUmSw"
  type="2001">
  <children xmi:type="notation:Node" xmi:id="_RE2DKM8uEd-egYv4lhUmSw"
    type="4001" />
  <styles xmi:type="notation:FontStyle" xmi:id="_RE2DI88uEd-egYv4lhUmSw"
    FontName="Arial" />
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_RE2DJM8uEd-egYv4lhUmSw" />
  <styles xmi:type="notation:FillStyle" xmi:id="_RE2DJc8uEd-egYv4lhUmSw" />
  <styles xmi:type="notation:LineStyle" xmi:id="_RE2DJs8uEd-egYv4lhUmSw"
    lineColor="0" />
  <element xmi:type="bpmn:Activity" href="modeler.bpmn#_RE2DIc8uEd-egYv4lhUmSw" />
  <layoutConstraint xmi:type="notation:Bounds"
    xmi:id="_RE2DJ88uEd-egYv4lhUmSw" x="249" y="62" />
</children>

```

Figure 56. Extrait du fichier d'entrée « Attributs graphiques »

## 10.4 ENVIRONNEMENT D'IMPLEMENTATION

Nous utilisons un ERP comme plateforme cible de l'environnement BPI, OpenERP. Avant de décrire l'implication de cette application au sein de la plateforme, nous présentons succinctement les ERP.

### 10.4.1 SI et ERP

Les ERP (Enterprise Resource Planning) ont été implémentés dans les années 90 dans des logiciels de gestion supportés par des bases de données relationnelles, sans que leur concept de gestion sous-jacent soit explicité. Traduits en français par « Progiciels de Gestion Intégré » (PGI) (Lequeux 2008), la plupart des solutions informatiques actuelles proposent un traitement intégré et synchronisé des données. Ainsi, ces solutions reposent essentiellement sur des concepts issus du traitement de l'information et non sur des concepts issus de la gestion et des sciences de l'organisation. Et bien que les ERP doivent s'appuyer sur les processus, (Bidan 2004) constate qu'ils s'intègrent par les données.

L'un des enjeux avoués du BPM est de parvenir à extirper les processus métier des applications où ils ont été dissimulés lors d'une approche ERP.

### 10.4.2 Côté technique

Un module OpenERP se traduit par un dossier contenant cinq types de fichiers qui sont sous format python (.py) et xml. Exploitant notre framework développé en Java-XML, cet exemple repose donc sur le triptyque Intalio Designer, EMF-Kermeta et OpenERP. La Figure 57 est un extrait de la perspective technique du modèle.

Le fichier `__terp__.py` (a) est une description du module indiquant le nom du module, sa version, ses dépendances vis-à-vis d'autres modules, etc. Le fichier `__init__.py` (b) est un fichier de démarrage indiquant les imports à effectuer lors du lancement du module, notamment les *wizards* associés au module. Un fichier `nomModule.py` décrit les classes spécifiques au module (c). Ces classes décrivent les formulaires associés au module. Elles définissent également la structure de l'interface. Enfin la dernière catégorie de fichiers regroupe les fichiers en .xml, dans lesquels nous fournissons une description du séquençement d'activités du module (`nomModule_workflow.xml`) (d), ou encore son interface (`nomModule_view.xml`) (e).

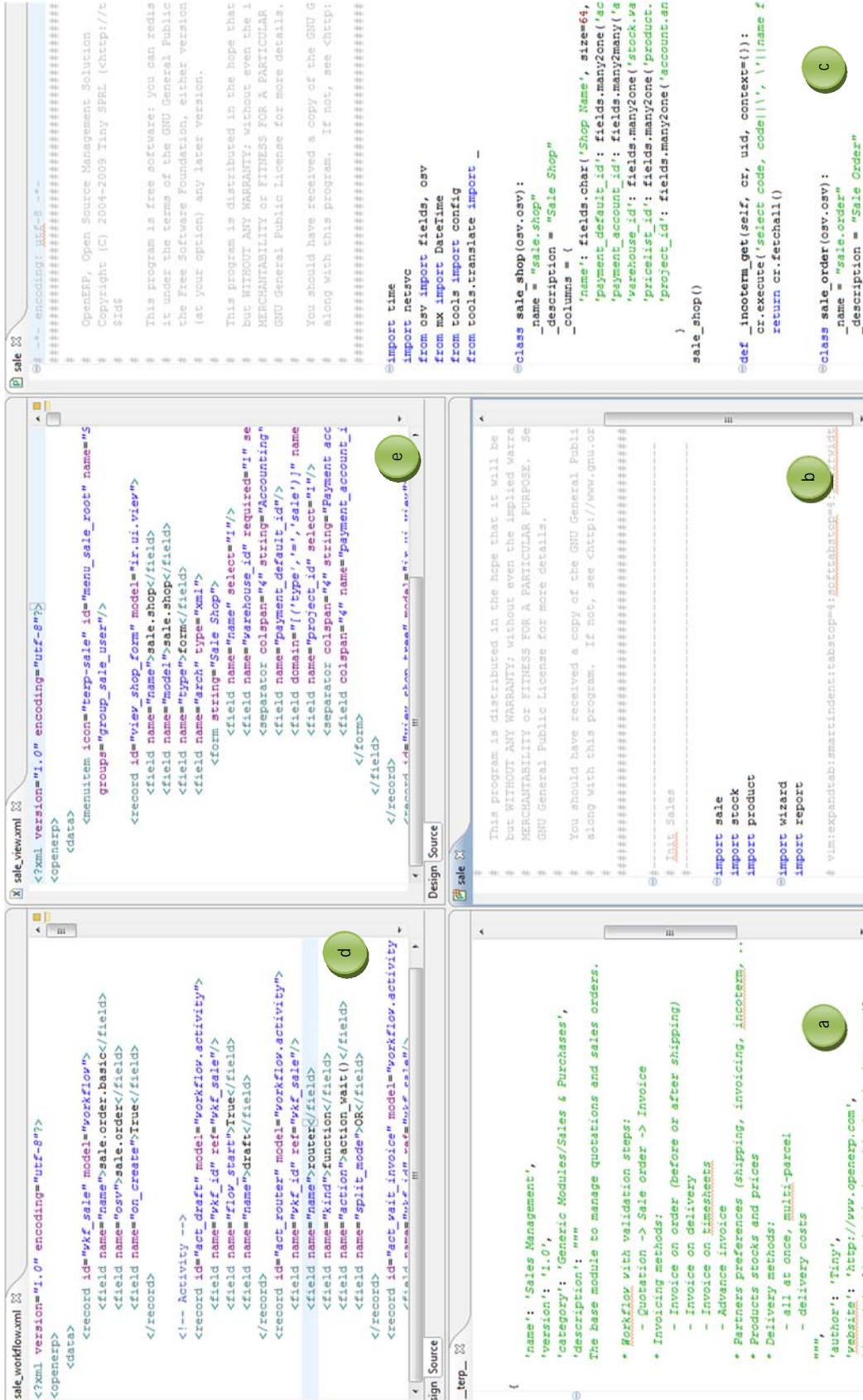


Figure 57. Fichiers de sortie pour le module OpenERP

Un exemple de fichiers constituant le module OpenERP est présenté en annexes, section 15.4.3.

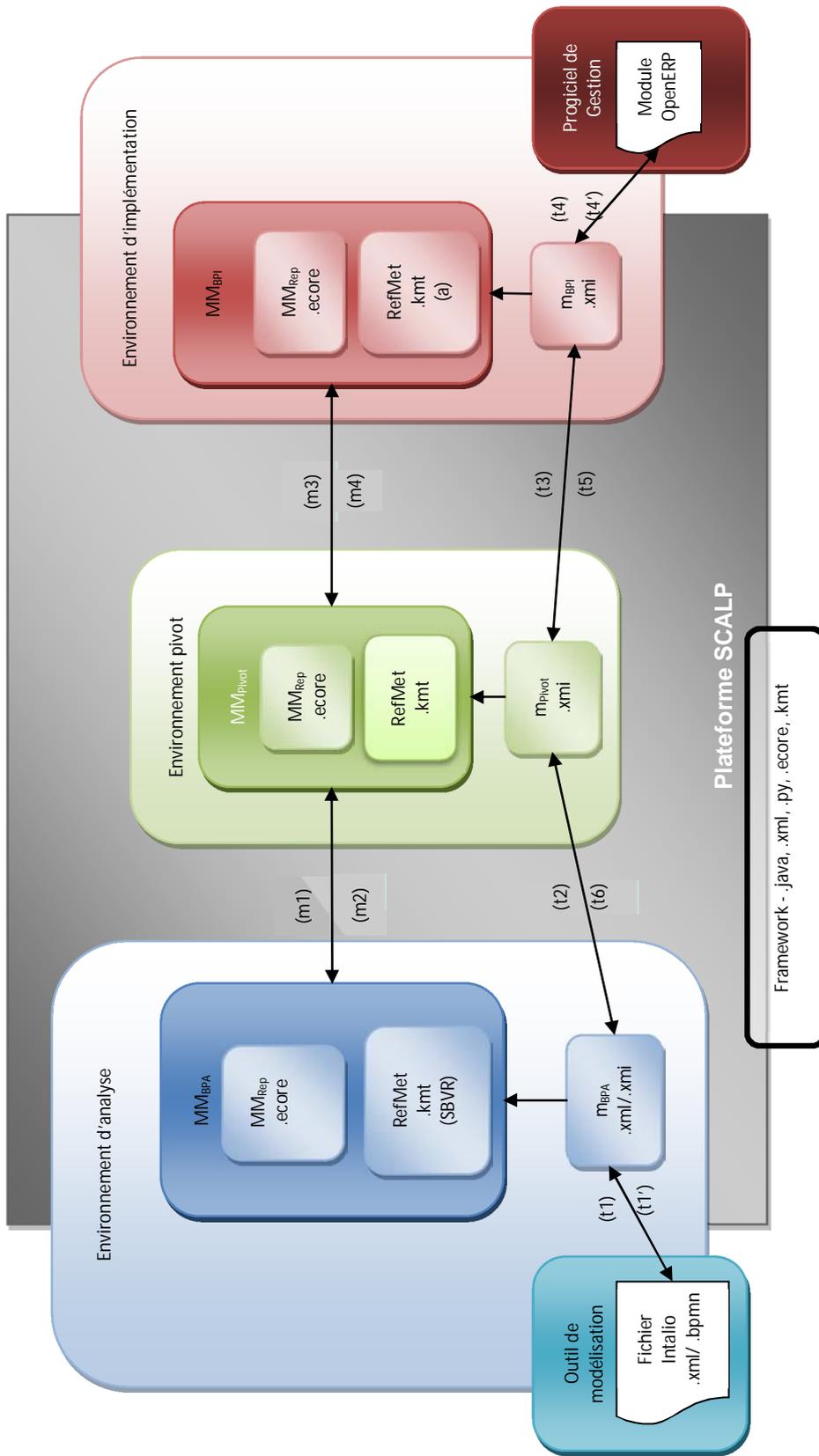


Figure 58. Architecture du framework logiciel

## REMARQUE

Nous considérons qu'un processus métier composé de n-pools équivaut à n modules OpenERP. Dans un souci de simplification, l'exemple considéré dans ce manuscrit traite le cas d'un processus métier constitué d'un pool unique. Par ailleurs, nous nous concentrons sur les processus métier de type *control-flow*. Ils nous permettent d'obtenir la description d'un séquençement d'activités au sein du module ERP.

## 10.5 IMPLEMENTATION DES MAPPINGS ET TRANSFORMATIONS

En reconsidérant la Figure 43 (chapitre 9) et la Figure 51, nous obtenons la Figure 58. Elle illustre les opérations nécessaires au passage d'un modèle conceptuel issu de l'outil de modélisation Intalio Designer vers un modèle technique et son module OpenERP correspondant.

Le (a), présent dans le *RefMet* du  $MM_{BPI}$ , désigne l'ensemble des règles et contraintes que nous avons rencontrées lors de la réalisation de l'environnement d'implémentation. Nous avons assimilé cet ensemble comme étant le *RefMet* de notre métamodèle BPI.

Les mappings m1 et m2 représentent les opérations de mappings que nous effectuons depuis le  $MM_{BPA}$  vers le  $MM_{Pivot}$  et inversement. De même, les mappings m3 et m4 représentent les mappings depuis le  $MM_{Pivot}$  vers le  $MM_{BPI}$  et inversement.

Les transformations t1 et t4' désignent les opérations nécessaires pour importer les modèles ou fichiers depuis les applications Intalio Designer et OpenERP vers la plateforme SCALP. A l'inverse, les transformations t1' et t4 permettent de transformer les modèles issus de la plateforme SCALP en modèles utilisables par lesdites applications. Enfin, les transformations t2 et t5 permettent respectivement de transformer un  $m_{BPA}$  en un  $m_{Pivot}$  et un  $m_{BPI}$  en un  $m_{Pivot}$ . Les transformations t6 et t3 réalisent les opérations inverses.

### 10.5.1 Métamodèles BPA et BPI

Préalablement à ces transformations, les mappings (m1, m2, m3, m4) entre  $MM_i$  et  $MM_{Pivot}$  doivent être réalisés. Or nous n'avons pas, ici, de métamodèles existants ou bien-définis. Nous déterminons les métamodèles en suivant la méthode indiquée section 9.7 puis nous les retranscrivons sous format Ecore. Les équivalences sémantiques sont détaillées dans la partie 11 : Démonstration. Les deux métamodèles obtenus sont représentés Figure 59 et Figure 60. La définition du métamodèle pivot est discutée section 9.5.

Les différents mappings sont réalisés à l'aide de Kermeta. Kermeta peut être utilisé comme un tisseur d'aspect adapté aux métamodèles Ecore, capable de les manipuler sans les modifier. Pour cela, Kermeta utilise le *pattern* Visiteur que présente la section suivante.





Figure 60. Métamodèle BPI

### 10.5.2 Pattern Visiteur

La définition d'une transformation, selon ce *pattern* Visiteur, revient à définir un (ou plusieurs) visiteur(s), au sens motif de conception Visiteur (Gamma, Helm, Johnson, & Vlissides 1999) (Figure 61).

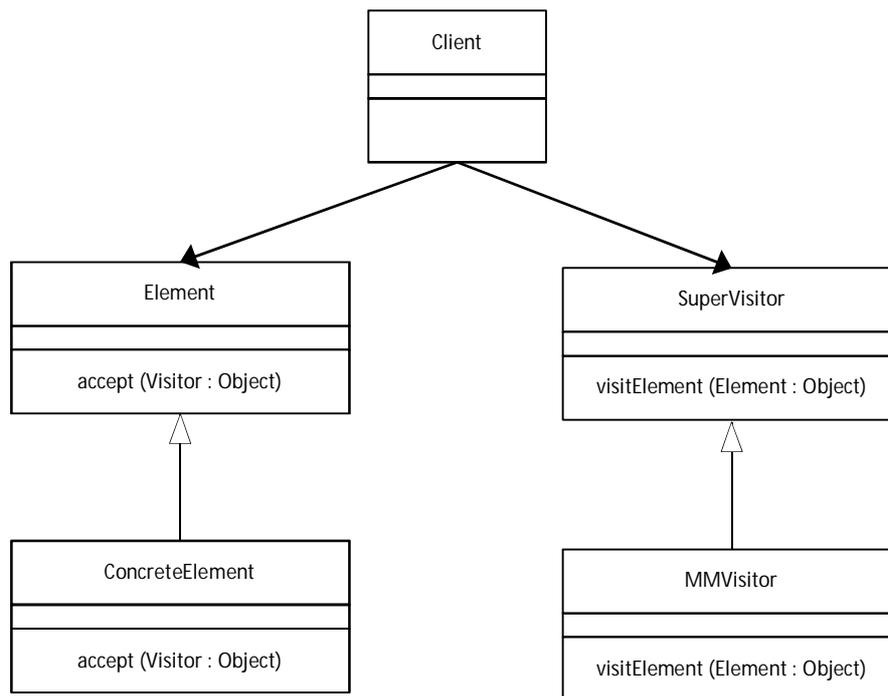


Figure 61. Diagramme de classe du pattern Visiteur selon (Gamma et al. 1999)

Le pattern Visiteur requiert d'implanter une opération *visitElement* dans la classe *MMVisitor* (Figure 62 (a) – ici *IntalioVisitor*) et une opération *accept* dans la classe visitée *ConcreteElement* (Figure 62 (b) – par exemple *BpmnDiagram*). Lors d'une transformation (autre que  $t_1$ ,  $t_1'$  et  $t_4$ ,  $t_4'$ ) depuis un modèle source vers un modèle cible, les classes du modèle cible ont une association dirigée vers les classes du métamodèle du modèle source (dans cet exemple, le  $m_{BPA}$  vers le  $m_{Pivot}$ ). Ainsi, si une classe du modèle cible a besoin de récupérer une valeur d'une propriété d'un élément du modèle source, elle peut directement le faire au niveau de l'interface visiteur. Les propriétés ainsi que leurs valeurs ne sont pas redéfinies au niveau de la classe du modèle cible, elles sont conservées. Cette approche permet donc au modèle cible de récupérer les valeurs des attributs du modèle source sans avoir à modifier leur métamodèle.

L'utilisation de ce motif facilite l'addition d'opérations pouvant être requises lors des transformations. En effet, chaque nouvelle opération sur le métamodèle se traduit par l'ajout d'un nouveau visiteur. A l'opposé, l'addition d'un nouvel élément est difficile : pour chaque élément une nouvelle opération dans chaque visiteur doit être créée. Néanmoins, si un certain niveau de maturité est atteint dans l'entreprise, nous présumons qu'à travers notre approche, nous modifions/ajoutons/retirons plus souvent les opérations réalisées par les métamodèles qu'ils ne changent.

Après avoir créé nos métamodèles et décrit la notion de visiteur, nous abordons la mise en œuvre des mappings et des transformations de modèles.

```

package bpmn; //Package du MM

require kermeta
//nsURI MM_under_study ou path
require "platform:/resource/ThesisProject/metamodels/BPA/intalio.ecore"

using kermeta::standard
using kermeta::persistence

/* Ajout d'une superclasse abstraite: SuperVisitor*/
abstract class SuperIntalioVisitor {
    operation accept (v: IntalioVisitor ): Void is abstract
}

/*Déclaration d'une transformation abstraite : transform */
package transformation {
    abstract class IntalioTransformation {
        operation transform ( input : SuperIntalioVisitor) : Object is abstract
    }
}

/* Déclaration d'un visiteur abstrait: IntalioVisitor,
avec une opération par élément composant le MM_under_study*/
abstract class IntalioVisitor {
    operation visitBpmnDiagram (bpmnDiagram : BpmnDiagram): Void is abstract
    operation visitPool (pool : Pool): Void is abstract
    operation visitLane (lane : Lane): Void is abstract
    operation visitNode (node : Node): Void is abstract
    operation visitActivity (activity : Activity): Void is abstract
    operation visitSequenceEdge (sequenceEdge : SequenceEdge): Void is abstract
    operation visitDetail (detail : Detail): Void is abstract
    operation visitSubProcess (subProcess : SubProcess): Void is abstract
}

/* Ajout d'un héritage sur les différents éléments du métamodèle
* -> On déclare une opération visit pour chaque élément*/

aspect class BpmnDiagram inherits SuperIntalioVisitor {

    method accept (v: IntalioVisitor ): Void is do
        v.visitBpmnDiagram (self)
    end

    operation loadFromModel(model: String): Void is do
        //Variable pour le modèle d'entrée EMF
        var loaded: BpmnDiagram
        /* Initialize the EMF repository */
        var repository: EMFRepository init EMFRepository.new
        /* Create|Load an EMF Resource */
        var resource: Resource init repository.getResource(model)

        //Charger l'instance de l'élément
        loaded ?= resource.one
        self.name := loaded.name
        self.pools.clear()
        self.pools.addAll(loaded.pools)
    end

    operation writeToModel(model: String): Void is do
        var repository: Repository init EMFRepository.new
        var resource: Resource init repository.createResource(model,
            "platform:/resource/ThesisProject/metamodels/BPA/intalio.ecore")
        /* Une fois la ressource chargée, l'attribut 'instances' contient tous les objets créés*/
        resource.instances.add(self)
        stdio.writeln("*****Modele sauvegardée: "+resource.instances.toString)/**/
        resource.save()
        //resource.saveWithNewURI(model.toString+"_save.xml")
    end
}

```

Figure 62. Pattern de conception Visiteur, en langage Kermeta

### 10.5.3 Mappings et transformations

La transformation  $t_1$  permet d'associer les deux fichiers issus de l'outil de modélisation Intalio Designer. Il s'agit des fichiers « Description logique » et « Attributs graphiques » que nous avons présentés précédemment. Leur association permet d'obtenir notre  $m_{BPA}$ . Cette première transformation se fait à l'aide du parseur XML JDOM<sup>23</sup>. Afin de valider le modèle obtenu, nous lui précisons le schéma XML, représenté ici par le  $MM_{BPA}$ .

A l'inverse,  $t_1'$  sépare les informations contenues dans  $m_{BPA}$  selon qu'il s'agisse d'une information destinée au fichier « Description logique » ou au fichier « Attributs graphiques ».

Les transformations  $t_2$ ,  $t_3$ ,  $t_5$ , et  $t_6$  se font à l'aide de l'atelier de métamodélisation utilisant Kermeta. Le mapping  $m_1$  se réalise après l'utilisation d'un visiteur déterminant et associant les éléments des métamodèles BPA et pivot. Ceci permet au framework d'effectuer la transformation  $t_2$ , du modèle BPA vers le modèle pivot, selon une méthode Builder-Linker. Nous opérons de la même manière pour transformer le modèle pivot en un  $m_{BPI}$  ( $m_3$ ,  $t_3$ ).

La transformation  $t_4$  est atypique par rapport aux autres transformations. En effet, la transformation  $t_4$  du  $m_{BPI}$  fournit un module OpenERP contenant les cinq fichiers explicités en 3.1. Cette transformation est semi-automatisée ainsi que son inverse ( $t_4'$ ). En effet, si des informations issues du  $m_{BPI}$  sont utilisées lors de la génération du fichier module\_View.xml, ce dernier se concentre essentiellement sur l'interface utilisateur. Cet aspect n'étant pas pris en compte dans nos processus métier, le fichier module\_View.xml nécessite d'être manipulé manuellement.

## 10.6 CONCLUSION

Afin de montrer l'intérêt de notre approche, nous proposons une plateforme mettant en œuvre une Solution pour la Cohérence et l'ALignement des Processus. Notre plateforme exploiteur des applications logicielles sans compatibilité particulière et est développée à partir des technologies de l'information. Elle manipule et fournit ainsi des modèles hétérogènes (Tableau 19).

Tableau 19. Applications et technologies utilisées par la plateforme SCALP

Niveau	Outil de modélisation	Environnement d'analyse	Environnement pivot	Environnement d'implémentation	Progiciel de Gestion
Applications, applicatifs	Intalio Designer 6.0.2	Ecore 0.7.0, EMF 1.4.0, XML JDom	Ecore 0.7.0, EMF 1.4.0, XML JDom, Kermeta 1.3.0	Ecore 0.7.0, EMF 1.4.0, XML JDom	OpenERP 5.0.14 PGAdmin III
Technologies, langages	BPMN, XML	XML, XMI, Java	XML, XMI, Java	XML, XMI, Java	Python, PostgreSQL, XML
Format des fichiers	.bpmn, .bpmn_diagram	.xmi, .ecore, .java	.xmi, .ecore, .java, .kmt	.xmi, .ecore, .java	.xml, .py

L'environnement d'analyse et l'environnement d'implémentation ne proposent pas de métamodèles au sens strict. Ainsi, après avoir constitué nos métamodèles selon la méthode exposée au chapitre précédent, nous avons défini les différentes étapes permettant la transformation d'un modèle de processus depuis un environnement source vers un environnement cible. Pour y parvenir, notre

<sup>23</sup> <http://www.jdom.org/>

plateforme s'appuie sur l'EMF et utilise le langage Kermeta. Ce dernier permet de définir les règles de comportement régissant nos métamodèles et d'utiliser le motif de conception Visiteur.

Nous proposons une démonstration exploitant les différentes briques technologiques exposées dans ce chapitre.



---

## Démonstration

---

La démonstration que nous présentons dans ce chapitre est un exemple de génération de modèles depuis un diagramme de processus vers un module ERP. Nous montrons les différentes étapes réalisant cette transformation et exposons certains de leurs détails techniques.

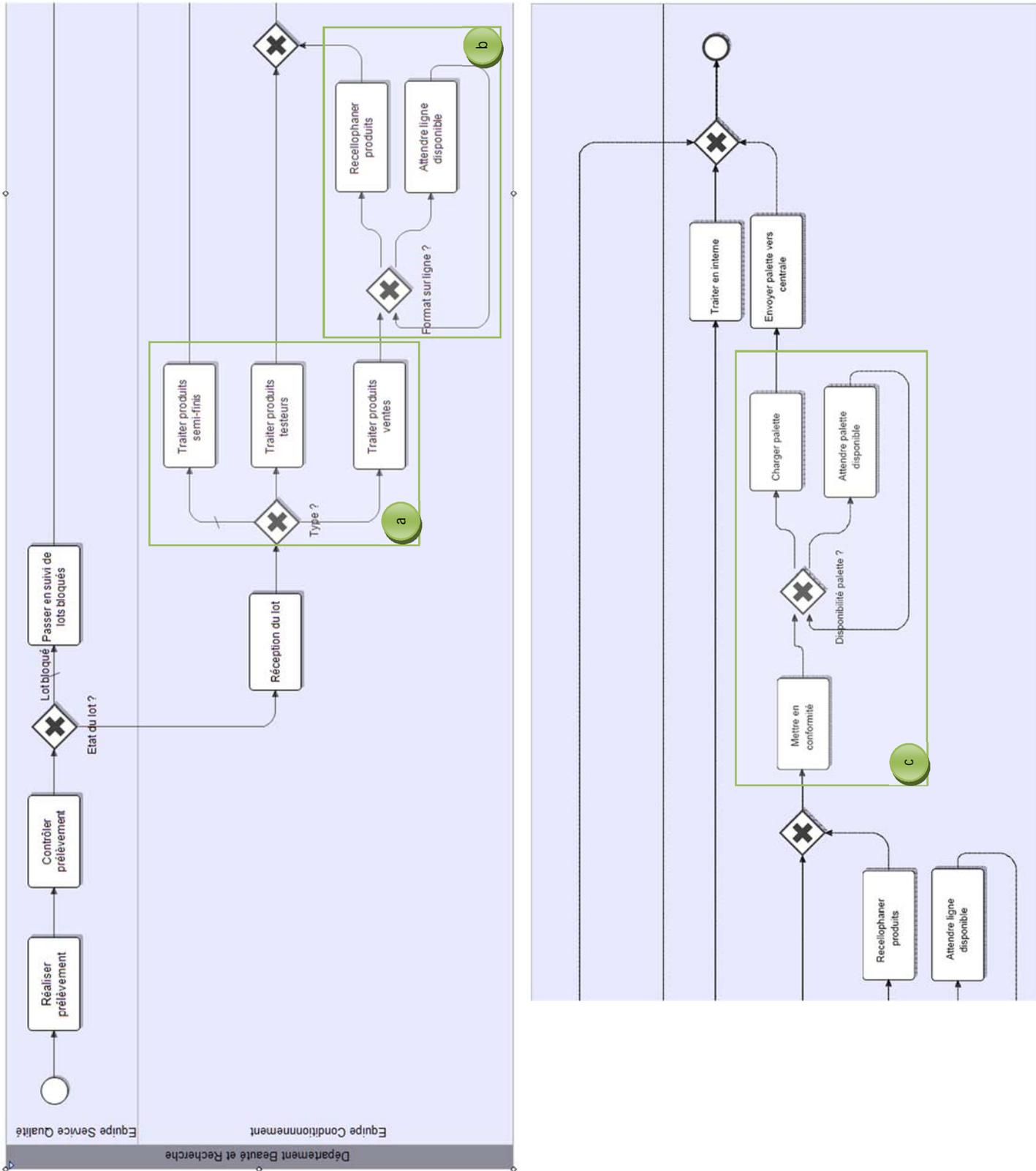


Figure 63. Processus de réintégration de parfums

## 11.1 PRESENTATION DU PROCESSUS ETUDIE

Nous présentons une expérimentation de notre plateforme logicielle avec un exemple de processus représenté Figure 63. Il s'agit d'un processus de réintégration de produits ayant subi un contrôle qualité. Ce processus a été récemment mis en place par un grand groupe de cosmétique pour ses unités de production de parfum. Nous l'utilisons ici comme un exemple de référence pour valider notre plateforme et démontrer notre approche explicitée partie C.

Durant le conditionnement des produits, un pourcentage de la production est prélevé afin d'être contrôlé par le département Qualité. Le lot prélevé doit ensuite être réinjecté dans le flux de production afin d'éviter des pertes financières importantes. Cependant, les produits testés constituant ce lot sont ouverts durant ce contrôle. La réintégration concerne un flux de 300000 produits par an, selon 94 formats de produits différents. Ceci implique de reconstruire l'emballage du produit avec un film cellophane sur une ligne de fabrication adéquate et de l'encrypter contre le détournement de produits. Un processus de réintégration de parfums est alors défini afin de rendre la majorité du lot réutilisable. Il est mis en place au sein de l'organisation

Le processus commence lors du prélèvement d'un lot de produits sur une ligne de production. Le lot est contrôlé par le laboratoire de Qualité. Les produits du lot prélevé à un instant  $t$  peuvent être de trois types différents (Figure 63 - a). (1) Les produits destinés à la vente nécessitent une étape de cellophanage avant d'être envoyés à une centrale de distribution. (2) Les produits testeurs ne nécessitent pas de cellophanage avant leur envoi en centrale distribution. (3) Les produits semi-finis ne nécessitent pas non plus une étape de cellophanage mais requièrent un traitement interne pour une éventuelle utilisation future. Il faut également considérer que l'étape de cellophanage ne s'effectue que sur une ligne de production vide afin d'éviter des mélanges de numéro de lot (Figure 63 - b). Avant leur envoi en centrale, les lots constitués de produits de type 1 ou 2 suivent une étape de remise en conformité puis une étape de constitution de palette (Figure 63 - c).

Le processus ainsi défini est utilisé à travers le scénario de validation que nous exposons dans la section suivante. Ses différents éléments sont définis en détail section 9.5. Nous allons voir comment ce processus dessiné à l'aide d'Intalio Designer devient un module ERP opérationnel ainsi que l'impact de notre démarche sur les modèles obtenus permettant leur cohérence intermodèle.

## 11.2 SCENARIO DE VALIDATION

Afin de valider notre démarche, nous proposons un scénario exposant les différentes opérations subies par le processus et ses modèles au cours de son cycle de vie. En partant de la phase BPA et d'un modèle conceptuel, ce scénario définit les étapes nécessaires pour que ce processus puisse atteindre la phase BPI et générer un modèle technique. Afin de réaliser son exécution, ce modèle peut être modifié. Nous présentons alors les étapes nécessaires permettant de revenir en phase BPA et générer un modèle conceptuel adéquat. Via ce scénario, nous souhaitons mettre en œuvre les concepts de notre approche et les mécanismes de notre plateforme. Il s'agit de démontrer que nous obtenons bien un meilleur alignement opérationnel entre le domaine métier et le domaine SITI. Un tel scénario est présenté Figure 64. Ce scénario est décomposé en trois grandes phases :

- (1) Transformation d'un modèle conceptuel (BPA) vers un modèle technique (BPI). Ce modèle technique nécessite un apport d'information afin d'être utilisable. Cette première phase permet de montrer que la plateforme SCALP est en mesure de réaliser une transformation « standard », depuis BPA vers BPI. Néanmoins l'utilisation des fonctions de conformité constructive et du modèle pivot résultant permet également d'obtenir systématiquement un modèle mBPI conforme à son métamodèle ;
- (2) Restitution du modèle graphique à partir du modèle technique ayant reçu un ajout d'information. Nous considérons que mBPI a reçu les modifications nécessaires à son bon fonctionnement par l'expert SITI. Pour simuler ces manipulations, nous procédons à un ajout d'information fonctionnelle au niveau du module OpenERP et de ses fichiers (essentiellement des fonctions codées en langage python) ainsi qu'à des modifications structurelles (modification du control-flow par un ajout d'activité et des changements au niveau du séquençement). Au final, nous montrons que l'aspect graphique du diagramme de processus peut être restitué et est conservé, à l'exception des éléments liés à l'activité ajoutée ;
- (3) Modification du modèle conceptuel obtenu et restitution du modèle technique à partir du modèle conceptuel. Lors de cette phase, nous simulons un changement d'ordre métier et ayant des répercussions directes sur la représentation du processus de réintégration. Nous considérons que l'activité « Réceptionner le lot » est désormais supervisé par un Responsable de Ligne, et que suite à cette activité, le responsable doit rédiger et archiver un document de type nomenclature du lot. Le but de cette phase est de montrer que les informations définies lors de la première phase sont réutilisées lors de la génération du mBPI, et que ce dernier prend correctement en compte les différentes modifications effectuées sur le diagramme de processus.

Le Tableau 20 résume les actions menées et les objectifs recherchés de ces trois phases.

Tableau 20. Récapitulatifs du déroulement et des objectifs du scénario

Phase *	Déroulement	Objectifs
1	Transformation Diagramme de processus → Module OpenERP	Transformation « standard » BPA → BPI
2	Modifications fonctionnelles et structurelles du module OpenERP Transformation Module OpenERP → Diagramme de processus	Propagation des modifications (BPI vers BPA) Intégrité et cohérence des informations
3	Modifications graphiques et structurelles du diagramme de processus Transformation Diagramme de processus → Module OpenERP	Propagation des modifications (BPA vers BPI) Intégrité et cohérence des informations

A noter que si les objectifs des phases deux et trois sont atteints, alors nous obtenons une synchronisation ainsi qu'une équivalence sémantique entre  $m_{BPA}$  et  $m_{BPI}$  ; et donc une cohérence intermodèle.

Ces phases peuvent elles-mêmes être décomposées en plusieurs étapes. Dans les sections suivantes, nous détaillons ces différentes étapes. Nous nous attarderons plus particulièrement sur la première phase. En effet, les mécanismes employés dans la deuxième et la troisième phase sont essentiellement les mêmes que ceux utilisés dans cette première phase de notre scénario. Nous nous appuyerons sur la Figure 58 qui détaille les différentes étapes de l'approche proposée.

**REMARQUE**

---

Le scénario décrit ci-dessus est représentatif de l'ensemble des fonctionnalités de la plateforme SCALP. Mais, il n'a pas vocation à être une validation formelle au sens d'un projet de développement logiciel.

---

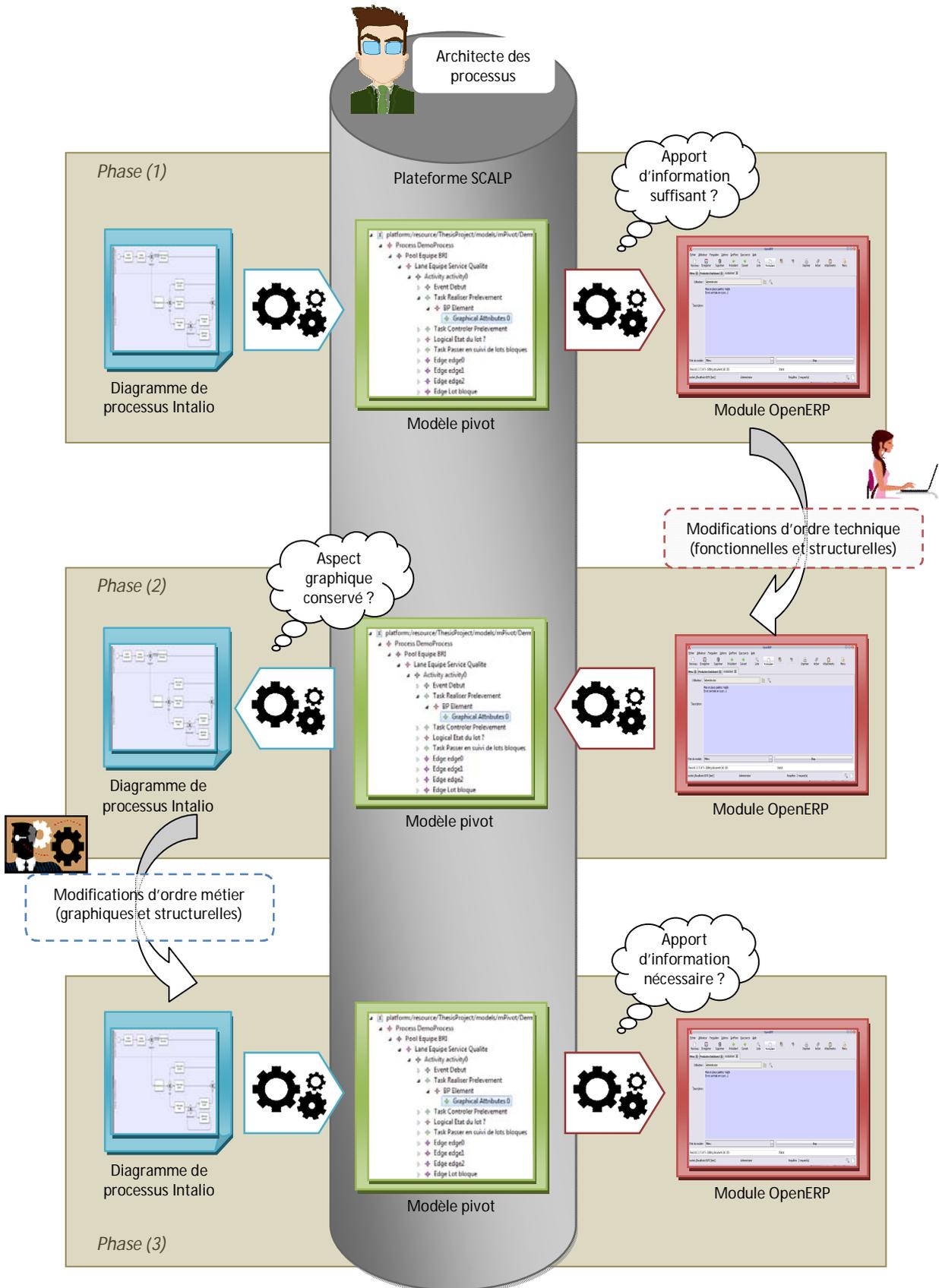


Figure 64. Scénario de validation

### 11.3 PREMIERE PHASE : DU DIAGRAMME DE PROCESSUS AU MODULE ERP

Nous considérons que les métamodèles d'analyse, d'implémentation et pivot, soit respectivement  $MM_{BPA}$ ,  $MM_{BPI}$  et  $MM_{Pivot}$ , existent et sont définis comme nous l'avons vu paragraphe 9.7. La première phase consiste à transformer un diagramme de processus respectant une charte graphique en un module exécutable sous un ERP. La Figure 65 reprend la Figure 58 et met en avant les différentes opérations réalisées dans cette première phase

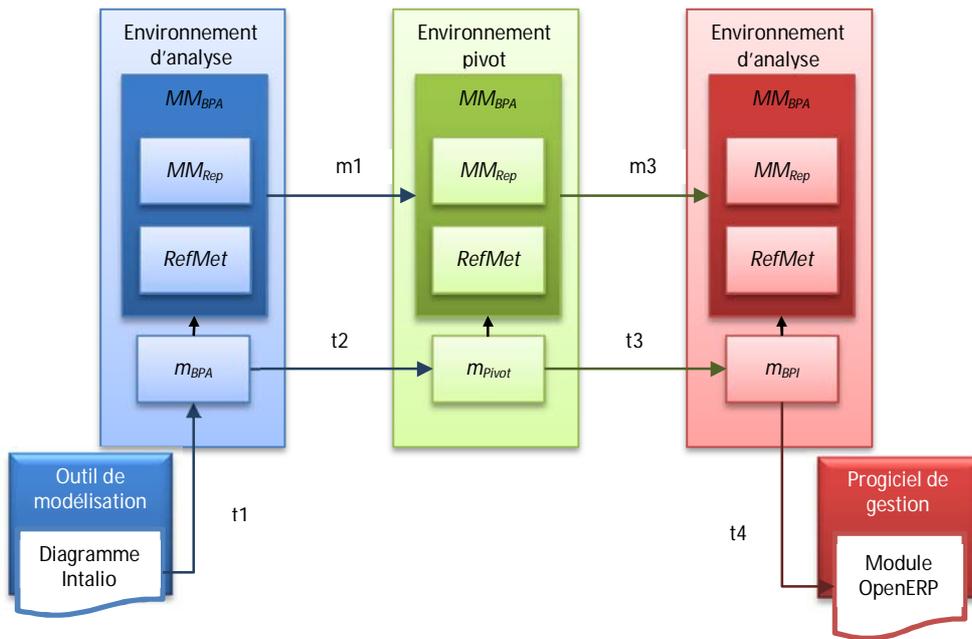


Figure 65. Déroulement de la première phase

#### 11.3.1 Du diagramme de processus Intalio vers un modèle d'analyse $m_{BPA}$

Nous importons les fichiers XML obtenus à partir d'Intalio Designer, comme décrit dans le paragraphe 10.3, au sein de notre plateforme SCALP.

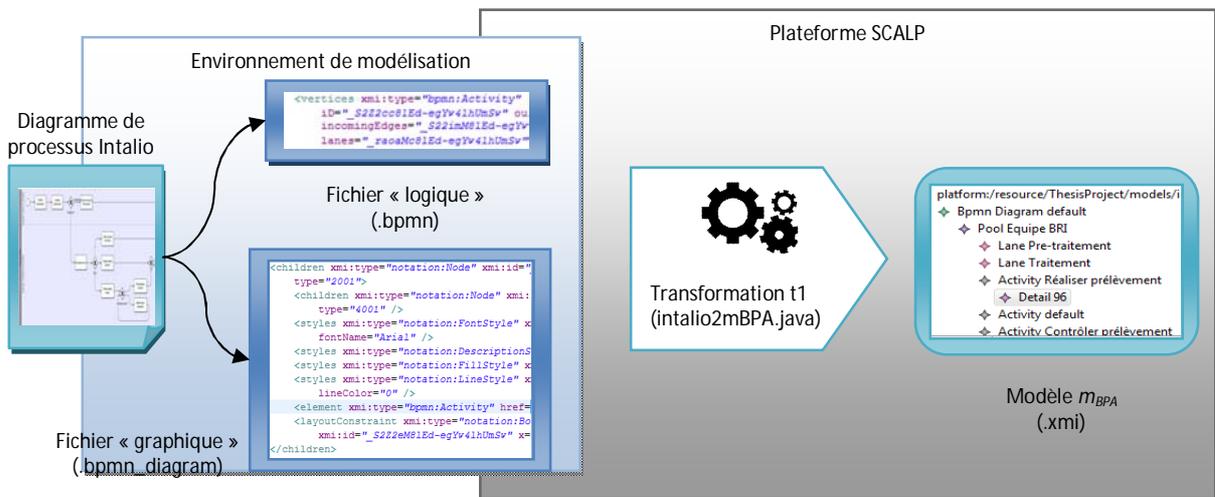


Figure 66. Du diagramme de processus au modèle  $m_{BPA}$

## DETAILS TECHNIQUES

Les Figure 67 et Figure 68 montrent respectivement un extrait du fichier « logique » et du fichier « graphique » obtenus à l'aide d'Intalio Designer. Il s'agit des représentations de l'activité « Réaliser prélèvement » du processus étudié.

```
<vertices xmi:type="bpmn:Activity" xmi:id="_S2Z28c81Ed-egYv4lhUmSv"
  id="_S2Z2cc81Ed-egYv4lhUmSv" outgoingEdges="_S2Z2io881Ed-egYv4lhUmSv"
  incomingEdges="_S2Z2imM81Ed-egYv4lhUmSv" name="Réaliser prélèvement"
  lanes="_raoaMc81Ed-egYv4lhUmSv" activityType="Task" />
```

Figure 67. Extrait du fichier « logique »

```
<children xmi:type="notation:Node" xmi:id="_S2Z2cs81Ed-egYv4lhUmSv"
  type="2001">
  <children xmi:type="notation:Node" xmi:id="_S2Z2c881Ed-egYv4lhUmSv"
    type="4001" />
  <styles xmi:type="notation:FontStyle" xmi:id="_S2Z2dM81Ed-egYv4lhUmSv"
    fontName="Arial" />
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_S2Z2dc81Ed-egYv4lhUmSv" />
  <styles xmi:type="notation:FillStyle" xmi:id="_S2Z2ds81Ed-egYv4lhUmSv" />
  <styles xmi:type="notation:LineStyle" xmi:id="_S2Z2d881Ed-egYv4lhUmSv"
    lineColor="0" />
  <element xmi:type="bpmn:Activity" href="modeler.bpmn#_S2Z28c81Ed-egYv4lhUmSv" />
  <layoutConstraint xmi:type="notation:Bounds"
    xmi:id="_S2Z2eM81Ed-egYv4lhUmSv" x="96" y="45" width="111" height="61" />
</children>
```

Figure 68. Extrait du fichier « graphique »

La transformation  $t_1$  est un programme java utilisant le parseur XML JDOM<sup>24</sup>. Cette transformation crée le modèle  $m_{BPA}$  à partir des deux fichiers cités précédemment. Le modèle obtenu est au format .xmi et est conforme à son schéma XML, qui correspond ici au métamodèle BPA, le  $MM_{BPA}$  (le fichier intalio.ecore) :

```
xsi:schemaLocation="http://ensiacet.org/lgc-psi/bpmn ../../metamodels/BPA/intalio.ecore".
```

Chaque modèle  $m_i$  rencontré durant les différentes phases du scénario de validation doit être conforme à son métamodèle  $MM_i$  associé afin d'être manipulable par Kermeta. Cette conformité est vérifiée à l'aide du menu contextuel proposé par Ecore (clic-droit → « Validate »)

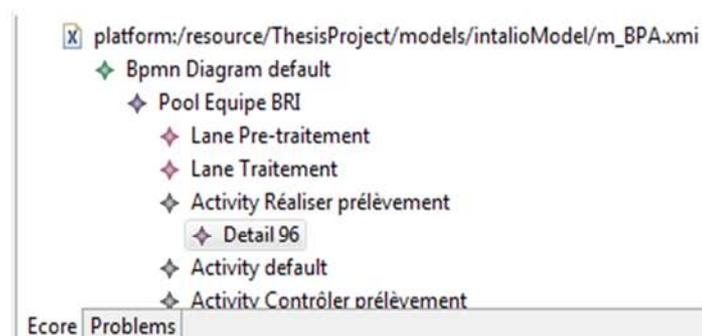


Figure 69. Extrait de  $m_{BPA}$

<sup>24</sup> <http://www.jdom.org/>

Un extrait du modèle obtenu est visible Figure 69. Nous y retrouvons l'activité « Réaliser prélèvement ». Un extrait du code source du fichier est disponible en annexe, section 15.4.1. A partir de ce modèle  $m_{BPA}$ , nous effectuons la transformation t2 afin d'obtenir notre modèle  $m_{Pivot}$ .

### 11.3.2 Du modèle d'analyse vers le modèle pivot

La transformation t2 permettant d'obtenir un modèle pivot  $m_{Pivot}$  à partir de  $m_{BPA}$  est un programme kermeta. Avant d'être réalisée, les mappings m1 ( $MM_{BPA} \rightarrow MM_{Pivot}$ ) et m2 ( $MM_{Pivot} \rightarrow MM_{BPA}$ ) doivent être effectués. Le Tableau 21 reprend ces mappings :

Tableau 21. Correspondance sémantique entre  $MM_{BPA}$  et  $MM_{Pivot}$

Élément $MM_{BPA}$ (A)	Correspondance sémantique	Élément $MM_{Pivot}$ (B)
BpmnDiagram	$A \stackrel{s}{=} B$	Process
Pool	$A \stackrel{s}{=} B$	Pool
Lane	$A \stackrel{s}{=} B$	Lane
Activity	$A \stackrel{s}{=} B$	Activity
SubProcess	$A \stackrel{s}{=} B$	SubProcess
SequenceEdge	$A \stackrel{s}{=} B$	Edge
Detail	$A \stackrel{s}{\subset} B$	BPElement

La transformation t2 est effectuée en reprenant le paradigme du pattern visiteur (paragraphe 10.5.2). Le fichier Visiteur détermine les éléments (et leurs attributs) appartenant à  $MM_{BPA}$ . Le fichier transformation les associe aux éléments et attributs constituant le  $MM_{Pivot}$ . La Figure 70 représente l'ensemble de cette étape et les différents fichiers manipulés.

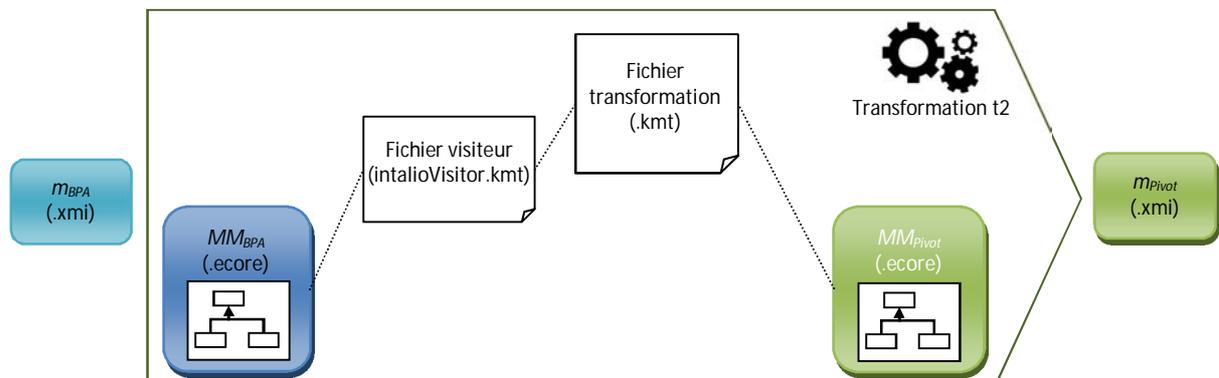


Figure 70. Du  $m_{BPA}$  au  $m_{Pivot}$

DETAILS TECHNIQUES

Considérons le mapping de l'élément Activity. Le tableau ci-dessous définit les correspondances existantes entre les différents attributs selon qu'il s'agisse d'une activité définie par le  $MM_{BPA}$  ou par le  $MM_{Pivot}$ .

Nous pouvons constater que de manière générale, la définition des éléments du  $MM_{Pivot}$  englobe celles des éléments du  $MM_{BPA}$ . Il en va logiquement de même entre  $MM_{Pivot}$  et  $MM_{BPI}$ . Les informations concernant l'aspect graphique de l'élément Activity sont contenues dans l'élément BPElement que nous avons présenté paragraphe 9.5.2. Cet élément permet de conserver les informations non utilisées par l'expert SITI (Figure 71).

Tableau 22. Mapping de l'élément Activity entre  $MM_{BPA}$  et  $MM_{Pivot}$

Elément $MM_{BPA}$ (A)	Attribut		Elément $MM_{Pivot}$ (B)	Attribut	Attribut équivalent
Activity		$A \subset B$	Activity		
	name	$A = B$		name	
	iD	$A \subset B$		BPElement	BPElement:GraphicalAttributes :businessData
	activityType	$A \subset B$		Node	Node:Logical:LogicalType
	x	$A \subset B$		BPElement	BPElement:GraphicalAttributes :CoordinatesType :xCoordinate
	y	$A \subset B$		BPElement	BPElement:GraphicalAttributes :CoordinatesType :yCoordinate
	height	$A \subset B$		BPElement	BPElement:GraphicalAttributes :height

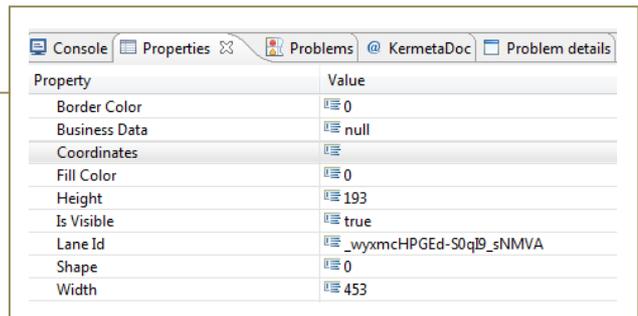
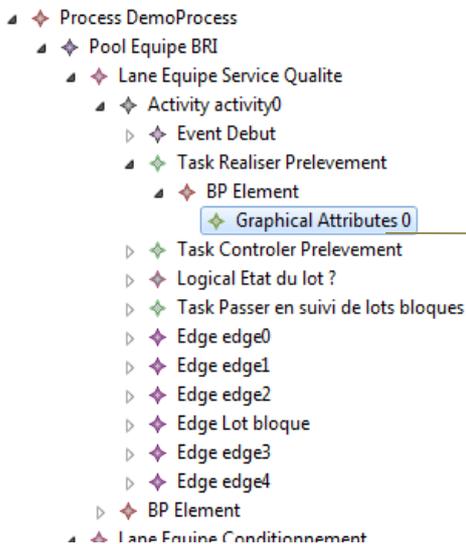


Figure 71. Exemple de  $m_{Pivot}$

### 11.3.3 Du modèle pivot vers le modèle d'implémentation

La transformation t3 possède les mêmes mécanismes que la transformation t2. Elle nécessite un modèle XMI en entrée, le  $m_{Pivot}$ , et fournit un modèle XMI en sortie, le  $m_{BPI}$  (Figure 72).

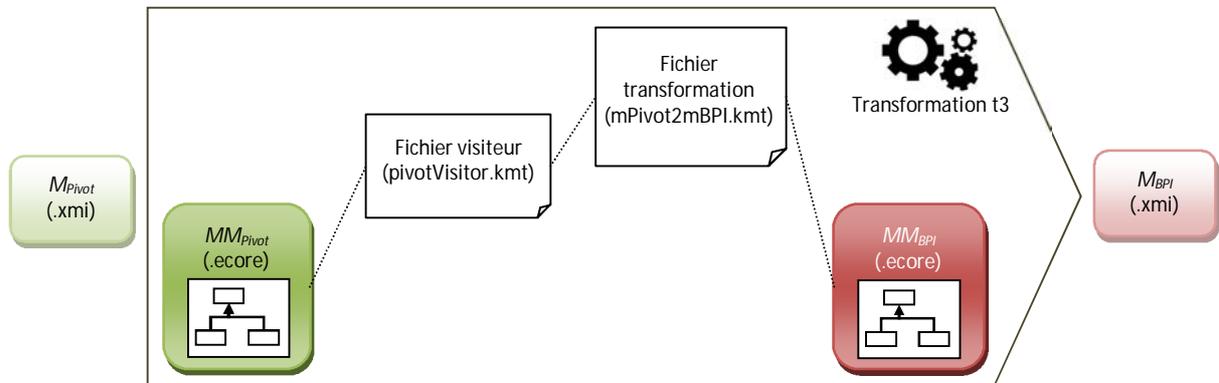


Figure 72. Du  $m_{Pivot}$  au  $m_{BPI}$

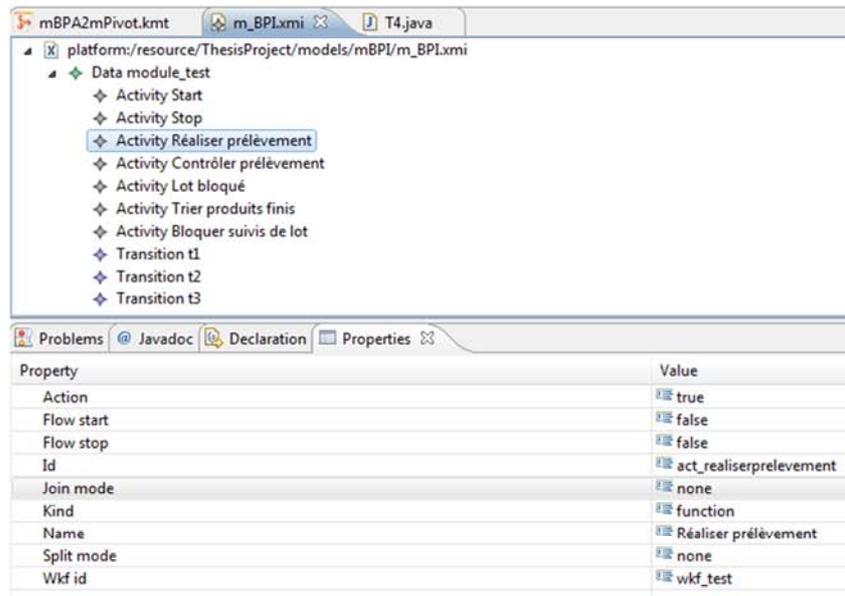
Les équivalences sémantiques entre  $MM_{Pivot}$  et  $MM_{BPI}$  (mappings m3,m4) sont définies Tableau 23.

Tableau 23. Correspondance sémantique entre  $MM_{Pivot}$  et  $MM_{BPI}$

Élément $MM_{Pivot}$ (A)	Correspondance sémantique	Élément $MM_{BPI}$ (B)
Process	$B \subset A$	Data
Pool	$B \subset A$	Workflow
Lane	$A \cap B$	*
Activity	$B \subset A$	Activity
SubProcess	$B \subset A$	Activity
Edge	$B \subset A$	Transition

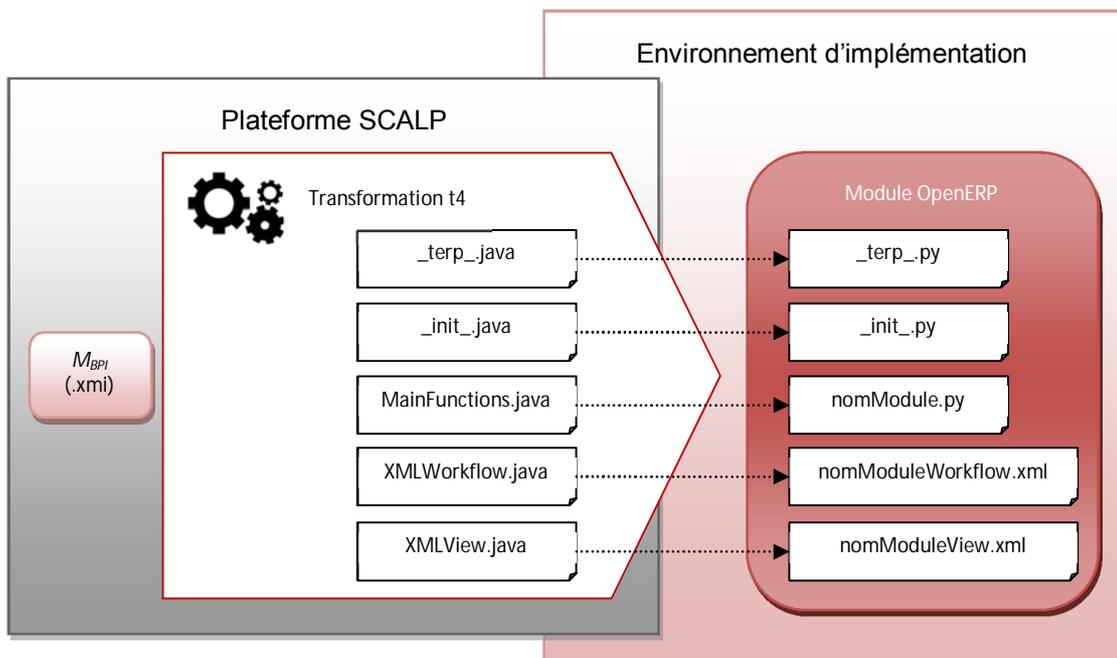
\* : L'élément Lane n'existe pas en tant que tel dans un modèle mBPI. Cependant, son attribut name définit l'attribut role\_id de l'élément Transition. Concrètement, lors de l'exécution du module, seule une catégorie d'utilisateurs reconnue ou disposant des mêmes droits que la catégorie role\_id pourra valider une transition possédant l'attribut role\_id.

La Figure 73 montre un extrait du  $m_{BPI}$  obtenu, un extrait du code source est fourni en annexe, section15.4.2.

Figure 73. Extrait de  $m_{BPI}$ 

#### 11.3.4 Du modèle d'implémentation vers le module OpenERP

A partir du  $m_{BPI}$  obtenu, nous constituons le module OpenERP, il s'agit de la transformation t4. Pour cela, cinq programmes java sont définis. Chacun de ces programmes fournit en sortie un fichier constituant le module OpenERP. Ces fichiers sont détaillés dans le paragraphe 10.4.2. Nous retrouvons ces fichiers en annexe, section 15.4.3.

Figure 74. Du  $m_{BPI}$  au module OpenERP

## 11.4 DEUXIEME PHASE : DU MODULE ERP AU DIAGRAMME DE PROCESSUS

Le module OpenERP obtenu à l'étape précédente n'est pas utilisable en l'état. Les fichiers nomModule.py et nomModuleView.xml demandent un apport d'information afin d'être utilisables. A cet effet, les fichiers sont amenés à être modifiés par l'expert SITI. La Figure 75 représente les principaux mécanismes mis en œuvre au cours de cette phase afin d'obtenir un diagramme de processus à partir des fichiers du module OpenERP.

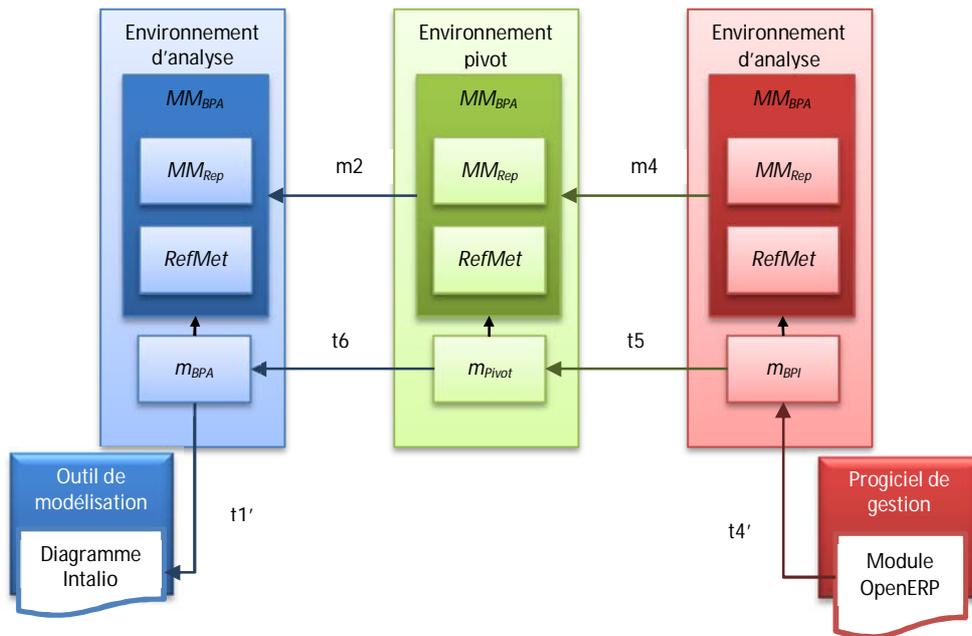


Figure 75. Déroulement de la seconde phase

### 11.4.1 Modifications apportées

Le fichier nomModule.py décrit la vue fonctionnelle de notre processus, des fonctions logiques exprimées en langage python sont y ajoutées. Ces fonctions représentent les actions réalisées par les différentes activités du processus de réintégration. Par exemple, nous désirons ajouter une fonctionnalité évaluant le temps effectif passé sur un lot ciblé (Figure 76).

```
def _calc_effective(self, cr, uid, ids, name, args, context):
    res = {}
    for bl in self.browse(cr, uid, ids):
        res.setdefault(bl.id, 0.0)
        for task in bl.tasks_id:
            res[bl.id] += task.effective_hours
    return res
```

Figure 76. Fonction python extraite du fichier nomModule.py

Nous effectuons également des modifications de type structurel. Pour cette partie du scénario, nous simulons l'ajout d'une activité permettant de mettre à jour la liste des lots de produits semi-finis utilisés dans le processus de réintégration. Cette modification se traduit par une (modification du control-flow de par l'ajout de l'activité et des changements au niveau du séquençement. Les fichiers nomModule.py et nomModule\_workflow.xml sont modifiés en conséquence à l'aide de l'éditeur d'OpenERP.

## REMARQUE : INTERFACE DU MODULE OPENERP

Le fichier nomModuleView.xml définit l'interface utilisateur du processus. Il est évident que lors de la modélisation du processus à l'aide de l'outil Intalio Designer, l'analyste métier n'a pas fourni d'informations spécifiques quant à sa représentation et sa manipulation sous le progiciel OpenERP. Le développement d'une telle interface ne rentre pas dans le cadre de notre démonstration. Il est donc à la charge de l'expert SITI de développer une interface ergonomique. Pour cela, il doit utiliser de manière adéquate les informations contenues dans les autres fichiers du module OpenERP et se baser sur la structure offerte par le fichier nomModuleView.xml. A titre d'illustration, la Figure 77 représente un exemple d'interface épurée possible et développée pour l'exemple.

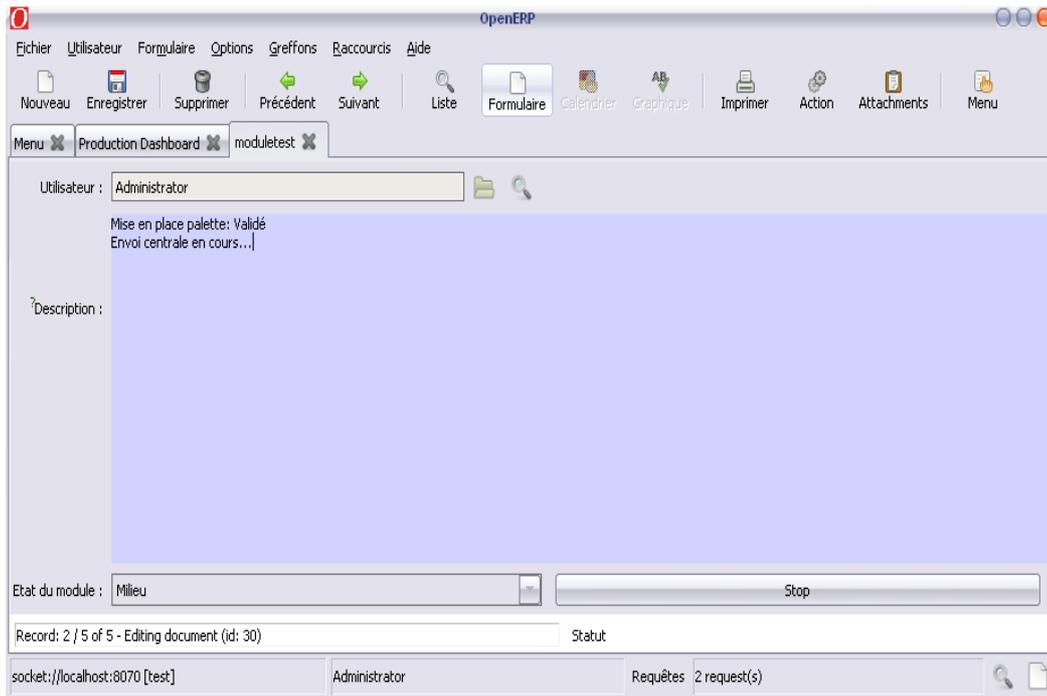


Figure 77. Exemple d'interface pour le module « Processus de réintégration » sous OpenERP

### 11.4.2 Du module OpenERP vers le modèle d'implémentation

La transformation t4' est un programme java se décomposant en trois étapes bien distinctes. Une première étape consiste à parcourir les différents fichiers python et à relever les informations nécessaires au  $m_{BPI}$ . De même la deuxième étape utilise le parseur XML JDOM sur les fichiers XML du module OpenERP. Enfin, cette transformation se termine par la création d'un modèle  $m_{BPI}$ , également réalisée à l'aide du parseur XML JDOM. Le modèle obtenu est au format .xmi et est conforme à son schéma XML, qui correspond au métamodèle BPI, le  $MM_{BPI}$ . La Figure 78 retrace le déroulement de cette transformation.

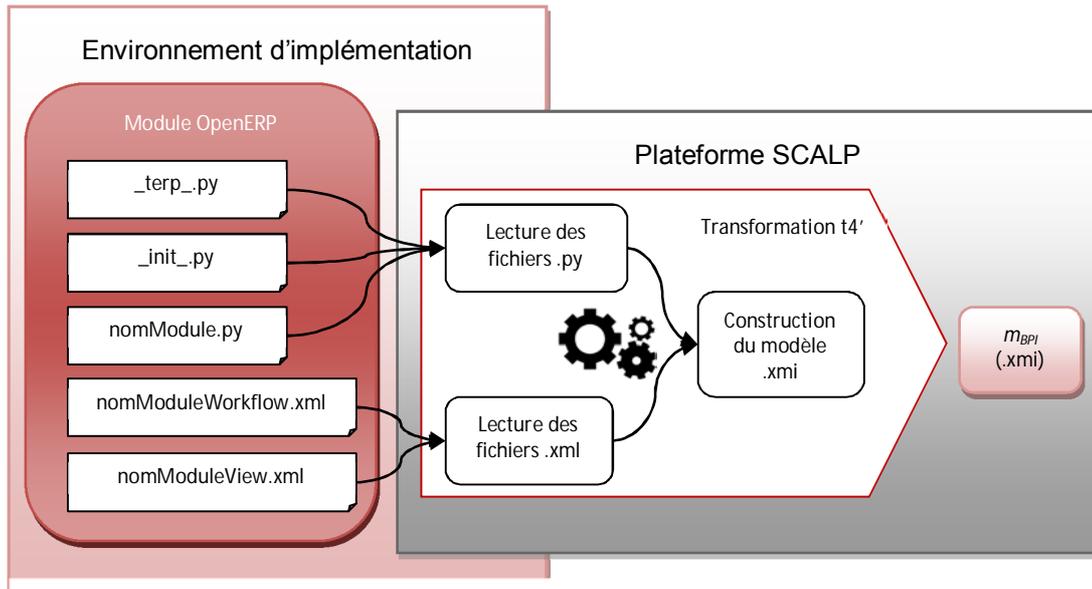


Figure 78. Du module OpenERP au le  $m_{BPI}$

## DETAILS TECHNIQUES

Les fonctionnalités ajoutées au fichier `nomModule.py` définissent les actions réalisées par les activités. Lors de la transformation  $t4'$ , les différentes modifications forment l'attribut `Action` de l'élément `Activity` auquel elles sont assignées.

Property	Value
Action	<code>def _calc_effective(self, cr, uid, ids, name, arg</code>
Flow start	<code>false</code>
Flow stop	<code>false</code>
Id	<code>act_envoi_centrale</code>
Join mode	<code>none</code>
Kind	<code>function</code>
Name	<code>Envoyer palette vers centrale</code>
Split mode	<code>none</code>
Wkf id	<code>wkf_id</code>

Figure 79. Extrait du  $m_{BPI}$  modifié

Les informations apportées au fichier `nomModuleView.xml` ne sont pas prises en compte : nous ne considérons pas la définition de l'interface utilisateur comme nécessaire à la bonne compréhension du processus étudié.

### 11.4.3 Du modèle d'implémentation au diagramme de processus

Les transformations  $t5$  et  $t6$  permettent respectivement d'obtenir  $m_{Pivot}$  à partir de  $m_{BPI}$  et  $m_{BPA}$  à partir dudit  $m_{Pivot}$ . Ces transformations se réalisent de la même manière que les transformations  $t2$  et  $t3$  :

- à l'aide du langage Kermeta ;
- et en utilisant le paradigme du pattern visiteur.

Cependant, deux différences notables sont à signaler. Tout d'abord, lors de la transformation  $t_5$ , nous apportons des informations supplémentaires concernant le modèle de processus. Ces informations, issues de la plateforme d'implémentation sont notifiées à travers le sous-élément *ImplAttributes* de l'élément *BPElement*. Dès lors, en récupérant les données contenues dans le sous-élément *GraphicalAttributes* du  $m_{Pivot}$  précédent, nous générons un modèle pivot contenant à la fois les données spécifiques au  $m_{BPA}$  et au  $m_{BPI}$ . Afin d'éviter toutes ambiguïtés, le modèle pivot généré est appelé  $m_{Pivot}^{\square 2}$  (Figure 80).

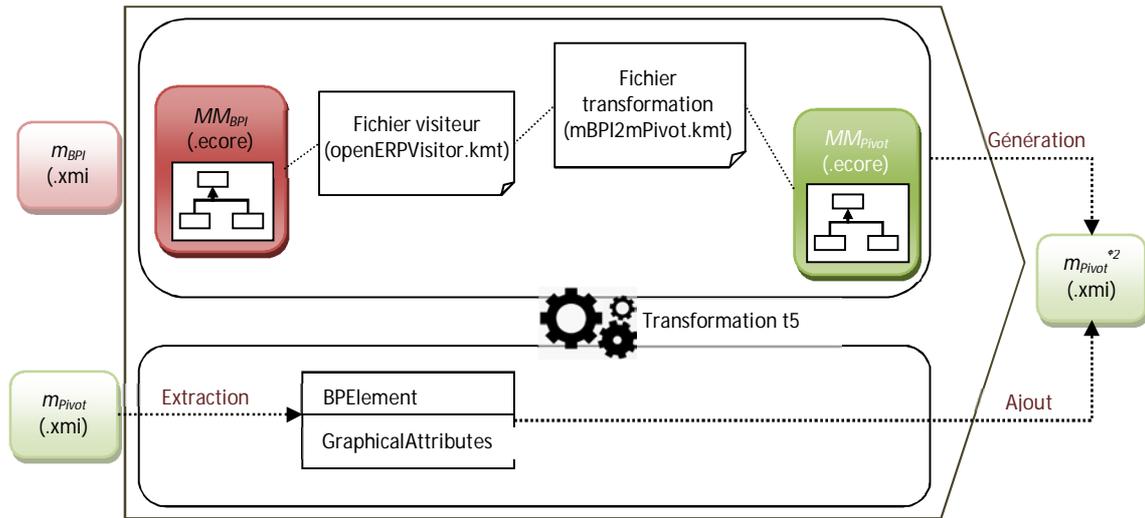


Figure 80. Du  $m_{BPI}$  au  $m_{Pivot}$

Puis, lors de la transformation  $t_6$ , nous utilisons les données contenues dans *BPElement* afin de générer un modèle d'analyse possédant les mêmes informations graphiques que le  $m_{BPA}$  généré lors de la première phase. Comme précédemment, pour éviter toutes ambiguïtés, le modèle obtenu est nommé  $m_{BPA}^{\square 2}$  (Figure 81).

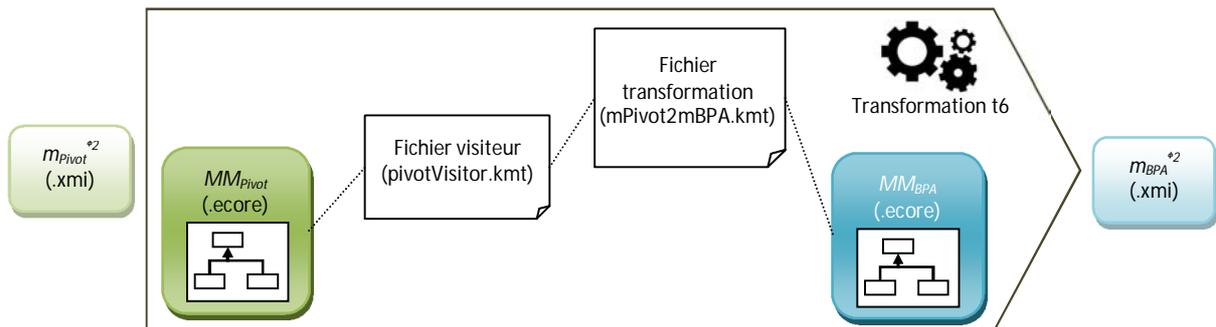
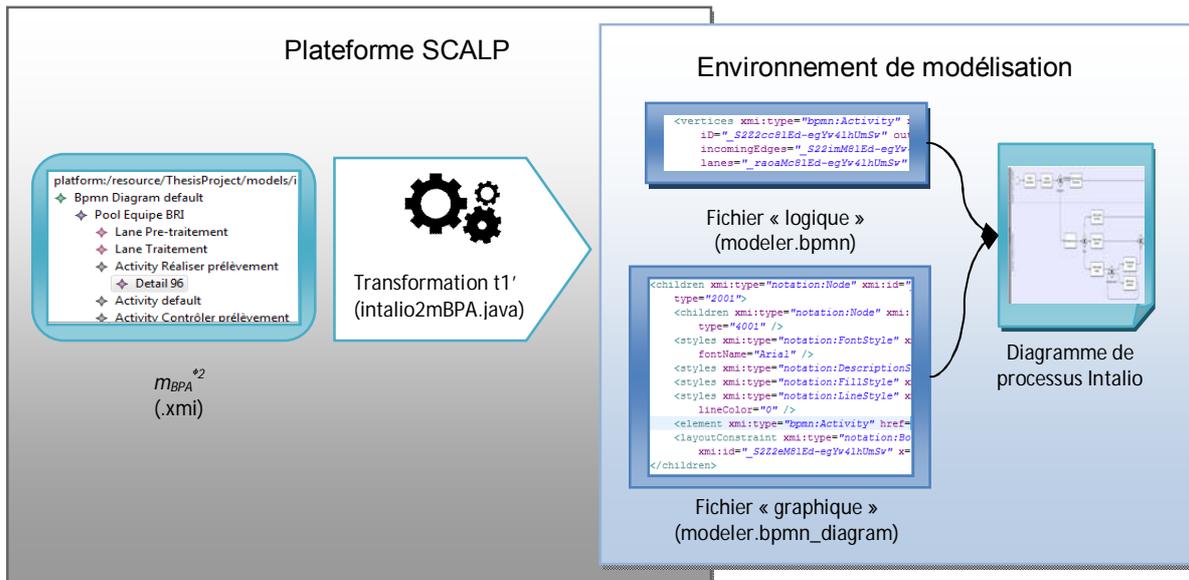


Figure 81. Du  $m_{Pivot}$  au  $m_{BPA}$

Pour finir, la transformation  $t_1'$  permet de transformer le modèle d'analyse obtenu durant cette deuxième phase en un diagramme de processus. Pour cela, nous utilisons un programme java utilisant le parseur XML JDOM. Ce programme génère les deux fichiers nécessaires à l'outil Intalio Designer pour représenter un processus : un fichier « logique » et un fichier « graphique » (Figure 82).

Figure 82. Du  $m_{BPA}$  au diagramme de processus

## DETAILS TECHNIQUES

Si lors de cette deuxième phase, les modifications apportées ne concernaient que les fonctionnalités des activités du processus (autrement dit leur exécution sous OpenERP) et l'apparence du module OpenERP, le diagramme de processus obtenu est identique à celui utilisé lors de la première phase.

Cependant, des modifications portant sur l'orchestration desdites activités peuvent être effectuées au début de la deuxième phase par l'expert SITI. Considérons, à titre d'exemple, l'ajout de l'activité « Mettre la liste à jour », activité suivant l'activité « Traiter les produits semi-finis » et précédant l'activité « Traiter en interne ». Lors de la génération du modèle pivot, les attributs graphiques de cet élément ont une valeur par défaut (« 0 », « null » ou « empty » selon les cas). Ces données sont visibles sur la Figure 83, représentant le fichier « graphique » résultant de la transformation t1'.

```

.. -----
<children xmi:type="notation:Node" xmi:id="_0ZzZcPXZEd-zdJZ02KDLOv"
  type="2001">
  <children xmi:type="notation:Node" xmi:id="_0Z0AgPXZEd-zdJZ02KDLOv"
    type="4001" />
  <styles xmi:type="notation:FontStyle" xmi:id="_0ZzZcfXZEd-zdJZ02KDLOv"
    fontFamily="Arial" />
  <styles xmi:type="notation:DescriptionStyle" xmi:id="_0ZzZcvXZEd-zdJZ02KDLOv" />
  <styles xmi:type="notation:FillStyle" xmi:id="_0ZzZc_XZEd-zdJZ02KDLOv" />
  <styles xmi:type="notation:LineStyle" xmi:id="_0ZzZdPXZEd-zdJZ02KDLOv"
    lineColor="0" />
  <element xmi:type="bpmn:Activity" href="modeler.bpmn#_0Zv9MfXZEd-zdJZ02KDLOv" />
  <layoutConstraint xmi:type="notation:Bounds"
    xmi:id="_0ZzZdfXZEd-zdJZ02KDLOv" x="0" y="0" />
</children>
<styles xmi:type="notation:DrawerStyle" xmi:id="_VeZP4e_0Ed-zNq19Y599ag" />
<styles xmi:type="notation:SortingStyle" xmi:id="_VeZP4u_0Ed-zNq19Y599ag" />
<styles xmi:type="notation:FilteringStyle" xmi:id="_VeZP4-_0Ed-zNq19Y599ag" />
</children>

```

Figure 83. Extrait du fichier « graphique » obtenu

La Figure 84 montre la représentation de processus correspondante, avec l'activité située à l'origine du diagramme (avec ses coordonnées x=0 et y=0).

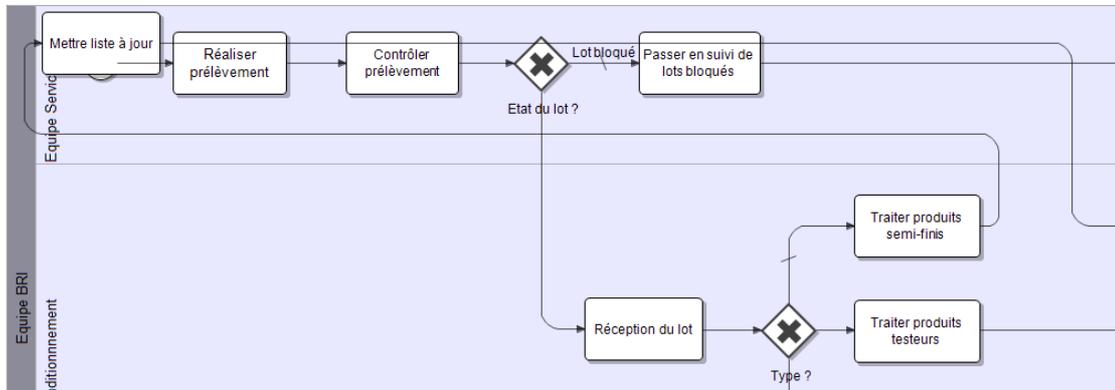


Figure 84. Extrait du diagramme de processus obtenu

Ce diagramme est représentée en intégralité en annexes, section 15.4.4.

#### REMARQUE : UTILISER LES FICHIERS « LOGIQUE » ET « GRAPHIQUE » AU SEIN D'INTALIO DESIGNER

Par défaut, Intalio Designer ne possède pas de fonctionnalités permettant d'utiliser les fichiers « logique » et « graphique » créés par un éditeur tiers. Pour obtenir un diagramme de processus découlant de ces deux fichiers, il faut effectuer la manipulation suivante :

- Créer un nouveau diagramme de processus vierge sous Intalio Designer ;
- Créer les sources comme expliquer paragraphe 10.3 ;

Remplacer ces sources par nos fichiers « logique » et « graphique », modeler.bpmn et modeler.bpmn\_diagram.

## 11.5 TROISIEME PHASE : DU DIAGRAMME DE PROCESSUS AU MODULE ERP

Considérons maintenant la troisième phase. Nous supposons que durant cette phase, l'analyste métier apporte des modifications au diagramme de processus et que nous disposons d'un modèle pivot contenant des informations relatives aux divers éléments composant le modèle de processus. Nous ne re-détaillons pas les étapes de la transformation permettant d'obtenir un module OpenERP à partir d'un diagramme de processus réalisé sous Intalio Designer. Ces étapes sont détaillées lors de la définition de la première phase. Le déroulement de cette troisième phase reprend en effet celui de la première phase, représenté par la Figure 65. Néanmoins, le résultat de cette transformation est variable. En effet, elle dépend des modifications effectuées sur le diagramme de processus.

- Modifications graphiques : l'analyste métier ré-agence les différents éléments du diagramme. Par exemple, il aligne les activités « Mettre la liste à jour », « Traiter les produits semi-finis » et « Traiter en interne ». A l'aide du modèle pivot, les fichiers constituant le module OpenERP sont générés automatiquement. Le fichier monModuleView.xml reste inchangé.

- Modifications ponctuelles : l'analyste métier ajoute, supprime, modifie le séquencement d'un ou plusieurs éléments de type nœud. Par exemple, il ajoute un acteur, le Responsable de Ligne, au processus, lui réattribue l'activité « Réceptionner le lot » et ajoute ensuite une activité « Archiver les données du lot ». Lors de la constitution du module OpenERP, seules les fonctionnalités liées aux nouveaux éléments (comme l'activité « Archiver les données du lot ») sont à définir et celles liées aux éléments modifiés (comme l'activité « Receptionner le lot ») sont à redéfinir, ainsi que les transitions adjacentes à ces éléments. Le fichier monModuleView.xml reste inchangé. Les différentes modifications réalisées sont représentées sur la Figure 100, dans les annexes, section 15.4.5.

## 11.6 CONCLUSION

Le scénario basé sur un processus issu de l'industrie cosmétique permet de valider notre approche et la plateforme SCALP résultant de notre projet de développement logiciel.

Lors de la première phase, nous avons réalisé la transformation d'un modèle conceptuel, un diagramme de processus, en un modèle technique, un module ERP. A chaque étape de cette transformation, la conformité entre le modèle xmi obtenu et son métamodèle est validée à l'aide d'Ecore. Durant cette phase, nous avons pu montrer comment nous stockons les données non-utilisables par le modèle technique à l'aide du modèle pivot. Ces données sont restituées lors de la seconde phase permettant d'obtenir un modèle conceptuel complet. Et comme précédemment, les données non-utilisables par le modèle conceptuel sont stockées par le modèle pivot. La troisième phase montre également que le modèle technique ne nécessite pas d'apport d'information si les modifications effectuées sur le modèle conceptuel ne sont que d'ordre graphique.

A travers les différentes transformations, nous générons un modèle de sortie prenant directement en compte les modifications subies par le modèle d'entrée (par exemple la création d'une activité) ou indirectement (modifications des instructions python associées à une activité). Nos deux modèles sont donc synchronisés comme défini paragraphe 7.1.1. Dans les deux cas, les données sont retranscrites au sein du modèle pivot. Les métamodèles associés aux modèles d'analyse et d'implémentation découlent du métamodèle pivot. Les différents éléments d'un métamodèle possèdent leurs équivalents, comme nous l'avons montré à l'aide des différents mappings spécifiant notre approche. Par ailleurs, comme nous l'avons indiqué, nos modèles restent synchronisés. Ils sont donc sémantiquement équivalents (paragraphe 7.1.2).

Ces trois phases démontrent l'utilité de notre approche. Elle transforme des modèles hétérogènes et d'abstractions différentes, tout en stockant et ajoutant des données nécessaires lors du passage BPA-BPI et inversement. Les modifications étant propagées lors de ces transformations et l'intégrité des informations étant assurée, notre approche permet une synchronisation et une équivalence sémantique des modèles. Nous obtenons alors une cohérence intermodèle telle que définie chapitre 7.1.3.



---

## Ingénierie des processus au service de l'ingénierie des procédés

---

(Debauche & Megard 2004) : « Deux termes sont souvent utilisés pour désigner une définition ou un modèle de processus : « procédure », qui s'applique davantage à des processus impliquant des personnes et de l'immatériel (procédure bancaire, procédure budgétaire, procédure de justice, etc.) ; « procédé », plutôt utilisé dans la fabrication de produits à partir de matières premières ».

Dans ce chapitre nous essayons de déterminer la corrélation existante entre processus et procédé. Pour cela, nous adoptons un point de vue structurel, où nous cherchons à établir les liens existants entre les différents modèles de processus et de procédés. Puis nous nous situons selon un point de vue sémantique et définissons les liens existants entre ontologies de domaines, et règles métiers.

## 12.1 NOTIONS AUTOUR DE L'INGENIERIE DES PROCÉDES

L'ingénierie des procédés (*Process Systems Engineering* - PSE) est une discipline mature qui a évolué au fil du temps au sein de différents champs de recherche : le génie chimique, les mathématiques appliquées, l'informatique et le génie logiciel. Dans ce chapitre nous étudions la partie informatique/ génie logiciel avec notamment les outils et les méthodes dirigés par les modèles. L'informatique (au sens PSE) doit concilier la complexité inhérente d'un procédé avec la nature « multi-objectif » de la prise de décision durant le cycle de vie dudit procédé. Afin d'accomplir cette tâche, des outils et des méthodes basés sur des modèles phénoménologiques ont été développés. Les notions de contrôle et de production des procédés basés sur des modèles ne sont pas nouvelles (Edgar 2004). De même, la notion d'entreprise vue dans sa globalité ou encore la notion de gestion de sa chaîne logistique sont au cœur de nombreux travaux (Grossmann 2004), (Varma et al. 2007).

L'étude proposée par (American Chemical Society et al. 1996) identifie quatre disciplines techniques comme étant de futurs acteurs majeurs dans le développement de l'ingénierie des procédés physico-chimiques :

- Les nouvelles sciences chimiques (synthèse chimique, technologie des matériaux, ...) et technologies d'ingénierie (mesures chimiques, science du procédé,...) ;
- La gestion de la chaîne logistique ;
- L'industrie et les opérations (capacité de production, construction de sites, contrôle des procédés et des informations) ;
- Et les systèmes d'information.

Selon (Schneider and Marquardt 2002) la discipline liée aux technologies de l'information a longtemps été traitée d'un point de vue « approvisionnement, fabrication et distribution de produits et matériaux ». Depuis, plusieurs auteurs ont évoqué l'intérêt d'utiliser des techniques de modélisation de processus métier. Toutefois, la conception des procédés à l'aide de méthodes de modélisation comme CLiP (Bayer et al. 2001) ou plus récemment IDEF0 (National Institute of Standards and Technology (NIST) 1993), (Hirao et al. 2008), ne permet pas l'association des différents concepts liés au procédé avec ceux de l'entreprise. Au contraire, nous pensons qu'une approche « top-down », depuis l'entreprise vers le procédé, est possible. Une telle démarche permettrait de mettre en adéquation les objectifs métier d'une entreprise et ceux induis par la mise en œuvre d'un procédé bio-physico-chimique, quel que soit son type (continu ou batch), (statique ou dynamique).

Dans ce chapitre, nous nous concentrons sur la conception des procédés avec le support des concepts, méthodes et outils de l'Ingénierie des Entreprises et des Systèmes d'Information (IESI). Nous exposons en l'état nos réflexions sur les termes de processus et procédés. Nous cherchons notamment à déterminer comment coupler l'ingénierie des processus avec l'ingénierie des procédés et les bénéfices que nous pourrions obtenir. En particulier, nous analysons et déterminons de quelle manière l'IDM et les technologies de l'information peuvent influencer le cycle de vie des procédés.

### *Ontologie et ontologie domaine*

Toute activité humaine spécialisée développe un langage ayant un vocabulaire propre à sa spécialité. L'existence de tels langages entraîne des problèmes de compréhension et des difficultés à

partager des connaissances entre les acteurs d'une entreprise, les services d'une entreprise, les entreprises d'un même groupe, faisant des métiers différents. Le rôle des ontologies est d'améliorer la communication entre acteurs humains puis entre humains et ordinateurs et finalement entre ordinateurs.

Selon (Uschold and Gruninger 1996), « une ontologie peut prendre différentes formes, mais elle inclura nécessairement un vocabulaire de termes et une spécification de leur signification. Cette dernière inclut des définitions et une indication sur la façon dont les concepts sont reliés entre eux, les liens imposant collectivement une structure sur le domaine et contraignant les interprétations possibles des termes ». Plus précisément, (Gruber 1993) une ontologie est une spécification de conceptualisation. La thèse de (Izza 2006) traite plus en profondeur la notion d'ontologie, sa construction et son usage pour les SI.

A travers nos travaux, nous nous sommes intéressés en particulier à la notion d'ontologie de domaine. Les ontologies de domaine sont des ontologies construites sur un domaine particulier de connaissance (Guarino 1998), (Izza 2006). Elles décrivent le vocabulaire relié à ce domaine, en particulier ses concepts, les relations existantes entre ceux-ci, les activités propres à ce domaine et les théories le régissant. Plusieurs ontologies de domaine propres à l'ingénierie des procédés ont été développées comme VeDa, PML/GML (Bayer and Marquardt 2003) ou plus récemment OntoCAPE (Marquardt et al. 2010) dont les concepts-clés sont représentés sur la Figure 85.

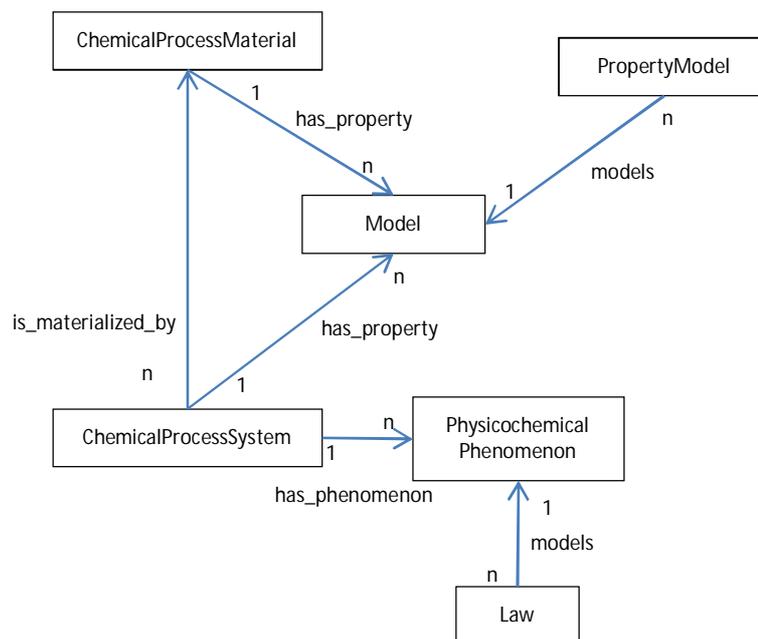


Figure 85. Concepts-clés d'OntoCAPE pour la modélisation de procédé (Yang et al. 2008)

## 12.2 VERS UNE APPROCHE PROCESSUS-PROCEDE

Comme nous l'avons énoncé précédemment le PSE s'intéresse aux méthodes et outils d'aide à la décision s'articulant autour des procédés. Ces méthodes doivent en particulier être en mesure de planifier, concevoir, rendre opérationnel et contrôler tous types d'opérations unitaires, de procédés et leur production (Takamatsu 1983). La modélisation assistée par ordinateur pour la conception de procédé ou *Computer-Aided Modelling and Process Design*, se base sur la modélisation du cycle de vie du procédé et

exploite fortement les technologies de l'information (Schneider & Marquardt 2002). Selon (Bayer, Weidenhaupt, Jarke, & Marquardt 2001) ce cycle de vie se décompose en sept étapes.

Tout d'abord une étape d'exploration durant laquelle nous exprimons les besoins, les caractéristiques techniques générales du procédé et les divers paramètres (économiques, environnementaux,...) s'articulant autour de celui-ci. S'ensuit une étape de recherche et développement. Une recherche de la chimie est effectuée puis la définition de nouvelles techniques et modèles mathématiques sont amenées à être améliorés. C'est à l'issue de cette étape qu'est produit le diagramme de procédé préliminaire, ou *Block-Flow Diagram* - BFD. L'étape suivante, la conception fonctionnelle, rend la vision du procédé plus concrète. Une étude des aspects structurels est réalisée puis nous évaluons un ou plusieurs procédés conceptuels ainsi que les alternatives de conception associées. De cette étape résulte le diagramme de procédé fonctionnel, ou *Process-Flow Sheet* - PFS. Lors de l'étape de conception détaillée, ce PFS est transformé en un plan de procédés, *Piping and Instrumentation Diagram*, où figurent tous les équipements et leur caractérisation. Les étapes de construction, exploitation et démantèlement s'intéressent plus particulièrement au(x) site(s) géographique(s) où les procédés seront mis en œuvre. La figure 10 montre ces sept étapes et ainsi que les quatre grandes familles d'outils-CAPE : Simulation pour l'analyse et la conception, Simulation pour la formation et la validation des systèmes de contrôle, Optimisation hors-ligne et en ligne, et Contrôle avancé.

Nous pensons que le BPM et en particulier la phase BPA peuvent intervenir en amont de ce cycle de vie et ainsi s'inscrire au sein des trois premières étapes du cycle (Figure 86).

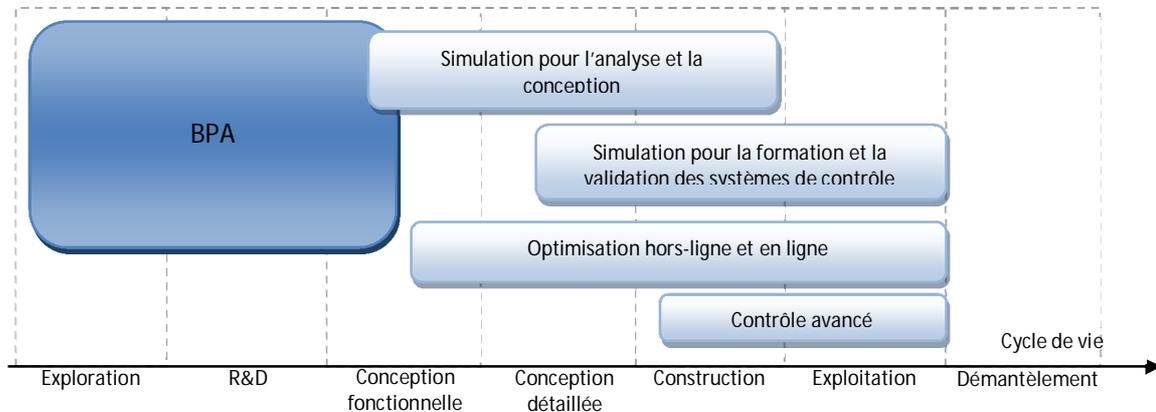


Figure 86. Découpage du PSE (Belaud 2002) et influence potentielle du BPA sur le cycle de vie du procédé

Formaliser les relations existantes entre le niveau processus et le niveau procédé de l'entreprise permet de considérer le procédé sous différents aspects de représentation et ainsi contribuer à une vue plus complète du procédé. En effet, selon (Klatt and Marquardt 2009), adopter une démarche multi-vue et « à travers différentes échelles de résolution chimique, spatiale et temporelle est recommandé pour amener la conception (du procédé) à un vrai optimum ». En termes de modèles, le diagramme de processus d'entreprise ou *Business Process Diagram* – BPD, se situe en amont d'un BFD (Figure 87), un BPD est un modèle pouvant essentiellement décrire une entreprise selon un point de vue fonctionnel.

Dans les paragraphes suivants nous proposons de définir les relations existantes entre l'ingénierie des processus, le génie industriel et le génie des procédés selon un point de vue structurel (logique). Puis nous définissons les liens sémantiques pouvant exister entre leurs modèles respectifs.

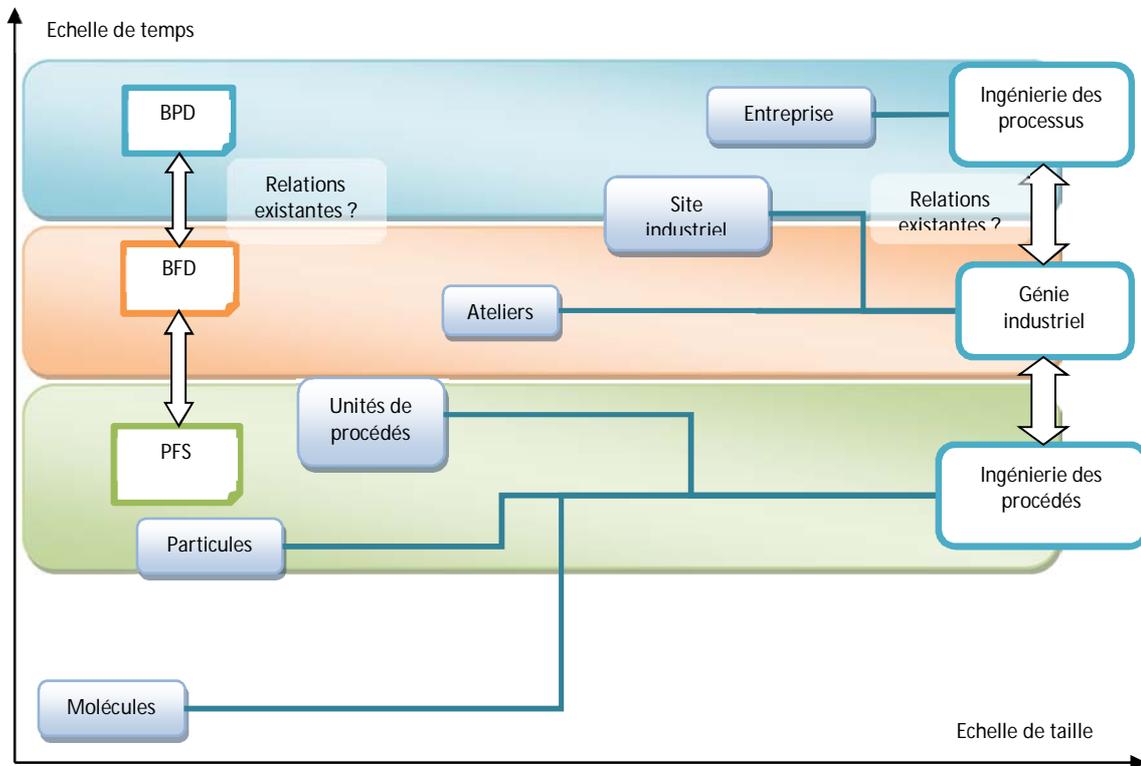


Figure 87. BPD, BFD, PFS à partir du « Chemical Supply-Chain » (Marquardt et al. 1999)

Un *Block-Flow Diagram* est défini comme étant une représentation généralement élaborée pour définir un ou plusieurs procédés et unités de productions. Sur ce diagramme peuvent figurer les bilans de matières globaux d'une installation, ainsi que les bilans des charges et produits passant d'une unité de production à une autre. Cette représentation peut également décrire les besoins en stockage et de réception/expédition des différents flux (de procédés et/ou d'utilités). Dès lors, nous pouvons assimiler les éléments modélisés par un BFD comme une activité de type sous-processus au sein d'un processus métier. Effectivement, selon (Godart and Perrin 2009) une activité est une description d'un fragment de travail qui constitue une étape logique à l'intérieur d'un processus. Les actions réalisées par une activité peuvent être manuelles ou automatiques. Pour s'exécuter, une activité utilise des ressources humaines et/ou des équipements. Si nous pouvons décomposer cette activité en plusieurs activités alors il s'agit d'une activité de type *sous-processus*. Un sous-processus est forcément appelé et initialisé par le processus global dont il fait partie. Plusieurs niveaux d'abstraction d'imbrication de processus peuvent être supportés.

Nous obtenons les relations suivantes représentées sur la Figure 88 :

- Une entreprise peut être représentée par un ou plusieurs processus ;
- Un processus peut être composé d'un ou plusieurs sous-processus, certains de ces sous-processus pouvant être décrits par des BFD ;

- Un sous-processus décrit par un BFD peut être représenté par un ou plusieurs diagrammes de procédé fonctionnels.

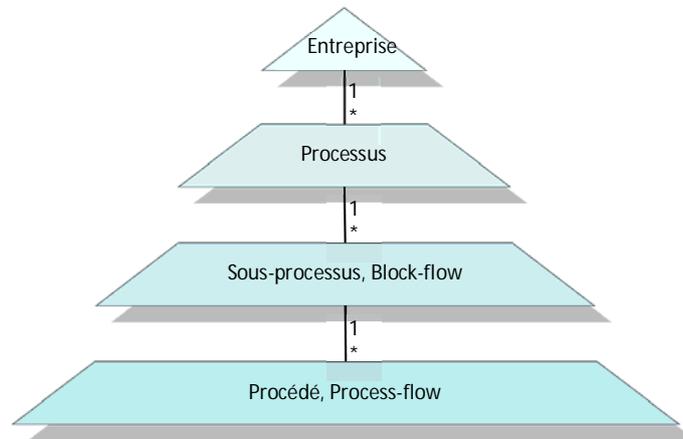


Figure 88. Cardinalités entre entreprise, processus, sous-processus et procédé

Ainsi, nous pouvons représenter le BFD lors de la modélisation d'un processus, comme par exemple dans la Figure 89, où le BFD est décrit par un sous-processus et dont le détail révèle celui-ci. La recherche d'équivalences entre les éléments de modélisation d'un BPD et d'un BFD est d'ailleurs en cours d'étude.

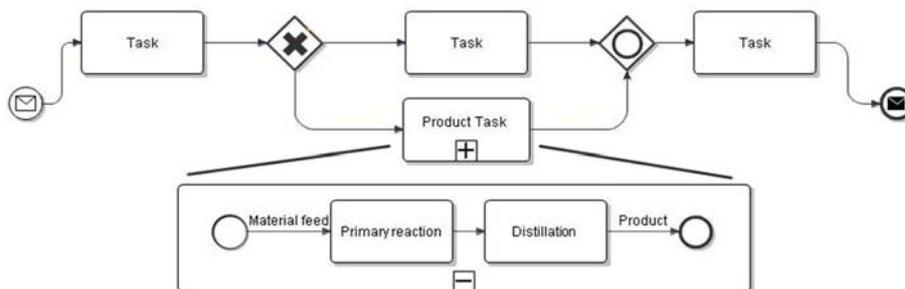


Figure 89. Approche multi-échelle Processus-Procédé (sous format BPMN)

Comme définis dans la section précédente, une ontologie de domaine pour système de procédés sert de source de concepts et permet de construire un modèle de procédé conceptuel. Dans l'ingénierie des processus, nous utiliserons la notion de gestion de règles métier ou *Business Rules Management - BRM*. D'après le Business Rules Group<sup>25</sup> (The Business Rules Group (BRG) 2007), une règle métier est une directive, destinée à régir, contrôler ou influencer le comportement métier d'une entreprise. Les règles métier s'appliquent aux processus mais sont modélisées indépendamment du modèle logique. Nous considérons ces règles comme étant similaires à des règles d'ontologie de domaine. En effet, les règles métier comme l'ontologie de domaine sont dérivées d'ontologies de plus haut niveau (Figure 90).

<sup>25</sup> <http://www.businessrulesgroup.org/>

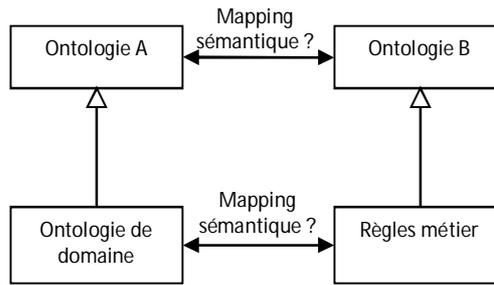


Figure 90. Equivalences sémantiques possibles entre ontologie de domaine et règles métier ?

La recherche d'équivalences sémantiques entre les concepts des règles métier et d'ontologies de domaine type-procédé aurait pour conséquence :

- de transmettre les contraintes déterminées lors de l'analyse d'un processus au modèle conceptuel de procédé (ici le BFD) ;
- et ainsi d'améliorer l'adéquation entre objectifs métier et conception du procédé ;
- afin d'obtenir un modèle multi-échelle processus/procédé cohérent ;
- de structurer et de formaliser via modèles, métamodèles et ontologies les approches *top-down* et *bottom-up*.

### 12.3 CONCLUSION

Considérer la dimension métier apportée par l'ingénierie des processus métier lors de la phase d'analyse du cycle de vie du procédé est une perspective intéressante d'un point de vue logique ainsi que d'un point de vue sémantique. Nous avons démontré la possibilité d'inclure un diagramme de procédés préliminaire (BFD) au sein d'un modèle de processus (BPD). Néanmoins, il faut encore définir formellement les différents éléments permettant de modéliser un BFD au sein de ce BPD. Cette approche verticale de l'analyse des processus et procédés doit être réalisée avec le soutien d'ontologies. Formaliser les relations sémantiques existantes entre les différents concepts liés au procédé avec ceux liés au processus sera le sujet de travaux futurs.

Le choix des différents niveaux d'abstraction assurant la cohérence multicouches des trois modèles appartient à l'équipe de modélisation. Néanmoins, au-delà de l'activité de modélisation des différents concepts et des bénéfices connus qui en résultent, l'ingénierie des procédés nécessite traditionnellement une activité de simulation lors de la phase « conception fonctionnelle ». Or si nous tenons compte des outils actuels de simulation du champ de l'ingénierie des systèmes de procédés (PSE), le « niveau de granularité » du schéma de principe de procédé (PFS) s'impose à nous à travers le périmètre d'une opération unitaire (*unit operation*).

Couplé à notre approche BPM, le PFS pourrait donc se traduire par un fichier de données pour un simulateur PSE type ProSimPlus. Ce fichier représentant le modèle Procédé serait alors en cohérence avec le modèle Processus. Ainsi l'alignement des différents modèles multi-échelles de l'entreprise serait permis. Pour réaliser cet alignement, une approche telle que celle présentée dans ce manuscrit peut être

utilisée. Il sera possible d'appliquer notre méthode pivot de manière à pouvoir générer depuis un BPD, un fichier de simulation type ProSimPlus (Figure 91).

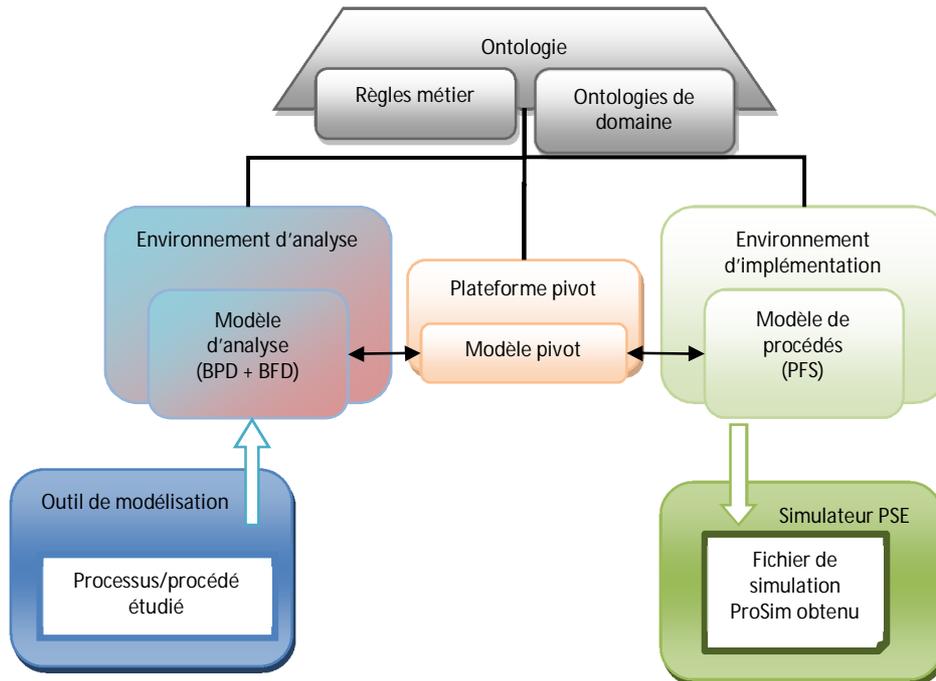


Figure 91. Approche pivot pour la génération d'un PFS à partir d'un BPD

Aussi, une telle approche pourrait s'appliquer au-delà du périmètre d'une seule entreprise. Le système global à étudier serait donc interentreprise. Cet aspect-là est hors de notre champ d'étude actuellement. Notons cependant que les travaux issus de l'interopérabilité des entreprises (Vernadat 2007) peuvent nous apporter des solutions au niveau Processus (hub d'entreprises, processus public-privé, architectures orientées services, UEML<sup>26</sup>, ...). Il resterait à investiguer le niveau Procédé, tout en conservant l'alignement et la cohérence sémantique des modèles, non seulement d'un point de vue multi-échelle mais également interentreprise.

<sup>26</sup> Unified Enterprise Modeling Language, (Anaya, Berio, Harzallah, Heymans, Matulevicius, Opdahl, Panetto, & Verdecho 2008)

# CINQUIEME PARTIE

---

## E. EPILOGUE

13.	Conclusion et Perspectives	.....	167
14.	Bibliographie	.....	175
15.	Annexes	.....	185

## RESUME

---

Cette dernière partie comprend une conclusion générale. Nous présentons également un bilan de nos travaux ainsi que les contributions réalisées à travers notre approche et la plateforme SCALP. Nous discutons des limites et des perspectives liées à la poursuite de ces travaux selon d'un point de vue conceptuel et selon un point de vue technique.

Cette partie s'achève par la bibliographie utilisée dans le manuscrit et les différentes annexes.

---

---

## Conclusion et Perspectives

---

### 13.1 CONCLUSION

Nos travaux de recherche abordent une approche générique pour la modélisation et l'implémentation des processus. Cette approche cherche à caractériser la notion d'alignement au niveau opérationnel et à maintenir la cohérence entre les modèles issus des environnements d'analyse et d'implémentation, la synchronisation entre ces modèles et leur équivalence sémantique.

Pour cela, notre approche adopte une démarche d'ingénierie des processus métier (BPM). Son objectif principal est de réduire l'écart existant entre l'environnement d'analyse (domaine métier) et l'environnement d'implémentation (domaine « Systèmes d'Information et des Technologies d'Information » - SITI). Cette approche se base sur un élément pivot garantissant la préservation et l'intégrité des informations des modèles manipulés et appartenant à des domaines différents. Pour cela, un dialogue entre les acteurs des différents environnements (l'analyste métier et l'expert SITI) doit s'effectuer et être supervisé. Ce rôle incombe à l'architecte des processus que nous avons défini dans ce manuscrit. Notre approche permet ainsi de :

- Garantir une cohérence intermodèle entre modèles hétérogènes et d'abstractions différentes, grâce aux mécanismes de transformations de modèles basés sur des métamodèles formellement définis et à l'utilisation d'un modèle pivot ;
- D'entretenir un couplage faible entre environnement d'analyse et environnement d'implémentation.

Pour démontrer la faisabilité de cette approche, un prototype logiciel a été développé en ce sens, la plateforme SCALP (Solution pour une Cohérence et un Alignement des Processus). Cette plateforme repose sur des technologies logicielles issues de l'open source et ne possédant pas de compatibilité particulière. La mise en œuvre de cette plateforme nous permet d'obtenir une transformation bilatérale entre modèles hétérogènes et une cohérence intermodèle.

## 13.2 BILAN

Pour synthétiser les contributions apportées par nos travaux de recherche, nous reprenons chaque partie du document et résumons les contributions associées.

### 13.2.1 *Cadre méthodologique*

De l'état de l'art abordé dans cette partie, nous avons étudié les relations existantes entre l'entreprise et les processus qu'elle met en œuvre, ainsi que les notions accompagnant la manipulation de ces processus par leurs modèles. Nous avons effectivement démontré que le processus se situe au cœur de la vue fonctionnelle de l'entreprise, cette vue se situant elle-même au centre des vues définissant l'entreprise. Une approche centrée processus permet d'améliorer l'agilité de l'entreprise. La notion de processus n'est pas révolutionnaire en soi, cependant gérer complètement son cycle de vie demeure un exercice difficile. La présence d'acteurs différents intervenant sur les représentations de processus et l'utilisation de modèles hétérogènes et d'abstractions différentes contribuent à la création d'un non-alignement opérationnel entre les domaines métier et SITI, écart se creusant durant le cycle de vie du processus. Pour tenter d'y remédier, des paradigmes et outils portant sur l'utilisation et la manipulation des modèles desdits processus, en particulier l'ingénierie des processus métier et l'utilisation de suites BPM, sont développés et mis en œuvre. Malgré cela, le lien entre l'environnement d'analyse (domaine métier) et l'environnement d'implémentation (domaine SITI) reste difficile à maintenir. Réduire l'écart entre domaines métier et SITI et donc améliorer l'alignement opérationnel est l'objectif de nos travaux de doctorat.

### 13.2.2 *Définition de l'approche*

Des limites d'une démarche BPM « standard », nous identifions trois conditions nécessaires à l'alignement opérationnel, réduisant par la même occasion l'écart métier-SITI : (i) la synchronisation, (ii) l'équivalence sémantique et (iii) la cohérence entre les modèles d'analyse et les modèles d'implémentation correspondants. Nous mettons également en avant des notions nécessaires à une gestion agile des processus d'entreprise, à savoir l'utilisation de règles métier, la volonté d'obtenir un couplage faible entre environnements d'analyse et d'implémentation et l'utilisation systématique de métamodèles pour la transformation de modèles. Essentiellement basée sur la vue fonctionnelle,

l'approche pivot que nous proposons assure un couplage faible entre environnements d'analyse et d'implémentation, tout en maintenant la cohérence entre les différents modèles et métamodèles. D'une généralité que nous avons qualifiée de « relative », l'approche reste indépendante des langages et des environnements utilisés. Sa mise en œuvre nécessite l'intervention d'acteurs divers : l'analyste métier et l'expert SITI. Néanmoins nous avons défini un troisième rôle comme étant un intermédiaire nécessaire au bon déroulement de l'approche : l'architecte de processus.

Basés sur notre étude de la littérature, nous proposons un métamodèle pivot permettant la réalisation de notre approche. Ce métamodèle s'appuie sur les concepts usuels utilisés pour la représentation des processus et respecte la classe de conformité simple des éléments d'XPDL proposée par WfMC. Nous proposons également des outils permettant la création et la détermination des relations sémantiques pouvant exister entre métamodèle pivot, métamodèle issu de l'environnement d'analyse et celui issu de l'environnement d'implémentation. Nous définissons également les fonctions de conformité constructive assurant la transformation d'un modèle vers un autre, chacun respectant au final le métamodèle qui lui est associé.

### 13.2.3 Mise en œuvre

Afin de valider nos différentes propositions exposées au long de ce manuscrit, nous présentons un prototype logiciel mettant en œuvre et validant notre approche, la plateforme Solution pour un ALignement et une Cohérence entre Processus, SCALP, plateforme reposant sur EMF et Kermeta. L'objectif de ce prototype est de montrer l'intérêt de notre approche en transformant un diagramme de processus représenté à l'aide de l'éditeur Intalio Designer en un module exploitable par le progiciel de gestion intégré OpenERP. Ainsi, notre plateforme s'appuie sur des applications logicielles sans compatibilité particulière et fournissant des modèles hétérogènes et dont les environnements d'analyse et d'implémentation ne proposent pas de métamodèles au sens strict.

L'étude de cas réalisée porte sur un processus de réintégration de produits type parfum ayant subi un contrôle qualité. Ce processus a été récemment mis en place par un grand groupe de cosmétique au sein de ses unités de production de parfum. Nous illustrons l'utilisation de la plateforme SCALP à travers un scénario constitué de trois phases. Lors de la première phase, nous réalisons la transformation d'un modèle conceptuel, un diagramme de processus, en un modèle technique, un module ERP. Cette phase montre comment SCALP réalise la transformation tout en préservant les données issues du modèle d'analyse et non-utilisées par le modèle d'implémentation à travers le modèle pivot. Le module obtenu nécessite des ajustements manuels. Lors de la seconde phase, SCALP permet de préserver à leur tour les informations issues du modèle d'implémentation, tout en restituant les informations nécessaires au modèle d'analyse. Nous prouvons également que les modifications effectuées entre ces deux phases sont bien prises en compte. La troisième phase du scénario montre que les modifications apportées au modèle d'analyse, SCALP est en mesure de fournir directement un module OpenERP complet et ne nécessitant pas d'apport d'information, à l'aide du modèle pivot. En synthèse, nous observons que SCALP génère un modèle de sortie prenant directement ou indirectement en compte les modifications subies par le modèle d'entrée. Et à l'aide d'un modèle pivot, l'intégrité des informations est assurée. Ainsi

SCALP permet une synchronisation et une équivalence sémantique des modèles. Nous obtenons alors une cohérence intermodèle permettant l'alignement opérationnel.

Parmi les applications possibles de notre approche, nous abordons la notion de processus-procédé. Nous essayons de déterminer la corrélation existante entre processus et procédé. Pour cela, nous considérons successivement deux points de vue. Tout d'abord un point de vue structurel, où nous établissons les liens existants entre les différents modèles de processus et de procédés utilisés dans l'ingénierie des processus d'entreprise et dans l'ingénierie des procédés industriels. Nous constatons qu'il est envisageable d'inclure un diagramme de procédés préliminaire au sein d'un modèle de processus. Néanmoins, cette approche verticale doit être réalisée avec le support d'ontologies. Nous les abordons en adoptant un point de vue sémantique. A travers cette vue, nous cherchons à définir les liens existants entre ontologies de domaines, et règles métiers. Notre approche permettrait d'assurer un alignement entre différents modèles multi-échelles de l'entreprise. Il sera possible d'appliquer notre méthode pivot de manière à pouvoir générer depuis un diagramme de processus d'entreprise, un fichier de simulation orienté PSE.

### **13.3 SYNTHÈSE DES CONTRIBUTIONS**

Les différentes contributions principales de nos travaux sur l'approche générique pour la modélisation et l'implémentation des processus sont récapitulées dans le Tableau 24.

Suite à ces contributions, nous abordons les limites et perspectives de nos travaux. Les différents points rencontrés peuvent se diviser en deux grandes catégories : les sujets directement liés à nos travaux, assurant une continuation et les sujets s'intéressant aux possibilités offertes à l'issue de ce manuscrit mais dont les recherches dans le domaine associé restent prospectives.

Tableau 24. Synthèse des contributions

	Propriétés	Contributions apportées par l'approche et la plateforme SCALP
P1.	Multi-Domaine	La plateforme intègre le cycle de vie complet du processus métier, depuis l'environnement d'analyse à celui d'implémentation.
P2.	Indépendance des environnements	Le modèle d'analyse ne prend pas en compte les informations nécessaires à la plateforme d'implémentation. Ici, le modèle pivot sert de modèle intermédiaire.
P3.	Prise en considération des modifications	Grâce à l'utilisation de métamodèles et du mapping effectué entre eux, la propagation des modifications se réalise aisément.
P4.	Evolution des outils	L'utilisation d'un pivot permet de faciliter les évolutions des outils. La modification d'un environnement d'analyse (resp. implémentation) n'a aucun impact envers l'environnement d'implémentation (resp. analyse)
P5.	Synchronisation	Le modèle pivot permet de restituer les informations lors des différentes transformations. La propriété P3 est également vérifiée, notre plateforme permet donc d'obtenir une synchronisation entre modèles.
P6.	Equivalence sémantique	L'utilisation de fonctions de conformité constructive (elles-même utilisant Ecore et Kermeta) permet d'obtenir une équivalence sémantique entre modèles.
P7.	Cohérence intermodèle	Les propriétés P5 et P6 étant vérifiées nous obtenons une cohérence intermodèle à l'aide de la plateforme proposée.
P8.	Comportement	Le comportement (avec entre autre les règles métier) de chaque élément d'un métamodèle est représenté au sein du fichier Visiteur associé (intalioVisitor.kmt, pivotVisitor.kmt, openERPVisitor.kmt).
P9.	Couplage faible	P2 et P4 étant vérifiées, la mise en œuvre de cette approche permet d'obtenir un couplage faible entre environnements d'analyse et d'implémentation.
P10.	Transformation	Les transformations sont réalisées à l'aide de l'outil de métamodélisation Kermeta.

## 13.4 LIMITES ET PERSPECTIVES

### 13.4.1 Utilisation et utilité de la plateforme SCALP

La mise en œuvre de notre approche résulte en une plateforme complexe, manipulant divers technologies et outils. Or cette complexité en elle-même démontre l'utilité d'une telle approche. Elle permet de concilier des environnements différents et autonomes, tout en manipulant leurs modèles. Ces modèles sont hétérogènes et d'abstractions différentes. Nous avons montré que plusieurs conditions étaient nécessaires afin d'obtenir une cohérence intermodèle. Néanmoins cette cohérence et ce travail en amont ne sont pas justifiables pour des transformations unilatérales, depuis BPA vers BPI. Ainsi, l'utilisation de notre approche se justifie si et seulement si nous désirons obtenir des transformations bilatérales formellement définies. Tout au long du manuscrit, nous avons montré que cette transformation bilatérale se situait au cœur d'une démarche d'alignement de l'infrastructure métier et des processus d'entreprise et les infrastructures SI.

Il faudrait cependant éprouver notre plateforme en utilisant des processus plus spécifiques issus par exemple de PME/PMI. De même, pour étayer la propriété « couplage faible » de notre plateforme, son utilisation avec de nouveaux outils est envisagée (par exemple Bizagi Process Modeler en amont et un outil bureautique type Microsoft Excel en aval).

#### 13.4.2 Génération de code et orchestration de services

De nombreux travaux proposent des démarches orientées services vers les SI permettant une transformation depuis BPA vers BPI, comme la méthode ADORE développée par (Mosser 2010) et utilisée dans le projet FAROS. L'approche que nous proposons fut développée dans une optique de génération de code. Prendre en considération une optique d'orchestration de services préexistants pourrait compléter notre approche. Pour cela, les mécanismes manipulant la partie pivot de notre approche devront être modifiés et fournir des méthodes de décomposition ou détection des éléments du processus en services déclarés préalablement. La plateforme cible serait des SI à base de composants ou packages standards de services.

#### 13.4.3 Alignement stratégique, alignement opérationnel<sup>27</sup>

A l'aide de notre approche, nous avons cherché à définir les liens existants entre les SI (infrastructures et processus techniques) et les unités métier (infrastructures et processus d'entreprise), et ainsi d'apporter notre contribution à la littérature. Nous avons ainsi réduit nos travaux au domaine interne en nous préoccupant essentiellement de ces infrastructures SITI et des processus d'entreprise. Des travaux futurs pourront être menés sur la relation entre domaine interne et les stratégies employées par l'entreprise. L'étude de cet ajustement stratégique, menée notamment par (Thevenet 2009) et (Avila 2008) permettrait d'élargir les possibilités de notre approche. Nous serions en mesure de conceptualiser l'ensemble des séquences d'alignement telles que définies par le SAM.

La prochaine étape serait alors de permettre l'ingénierie dirigée par les processus, *Process-driven engineering*, étape supportée par notre approche et plateforme SCALP.

#### 13.4.4 Abstraction et interopérabilité

Nous avons sciemment réduit le nombre de pools utilisable dans un BPD. Ainsi, nous manipulons un unique pool, englobant l'ensemble des éléments du BPD et s'identifiant à ce dernier. De futurs travaux permettront d'utiliser n-pools est ainsi d'obtenir des processus interentreprises. Il convient alors de s'interroger sur la place de notre plateforme et de l'environnement pivot qui la définit.

De nombreux travaux se préoccupent des processus collaboratifs, en particulier (Rajsiri 2009) pour le niveau CIM et (Touzi 2007) pour les niveaux PIM-PSM. Nous pouvons également citer les travaux de (Fathallah 2010) qui, dans un cadre d'interopérabilité d'entreprises, cherchent à définir et maintenir la cohérence entre les différents SI d'entreprises selon les typologies de modèles utilisées.

---

<sup>27</sup> Au regard de la section 2.2.1, le lecteur averti pourra apprécier l'épanadiplose !

#### 13.4.5 Evolution de la plateforme SCALP

Le métamodèle pivot se restreint pour le moment à un ensemble de 18 objets. Ceci nous a permis de dé-complexifier notre champ d'étude. Afin de rendre ce métamodèle plus complet et donc le modèle pivot plus expressif, des développements sont en cours afin d'obtenir l'ensemble d'éléments constituant la classe standard de conformité de portabilité de modèle tel que défini par la WfMC. Des travaux futurs sont envisageables quant à la possibilité d'obtenir la classe de conformité complète.

Une autre évolution possible à prendre en compte est la possibilité d'inclure un environnement dédié à la supervision des processus métier, le BAM. Son rôle serait de fournir les outils nécessaires permettant d'obtenir un retour d'informations sur l'utilisation et l'exécution des processus issu de l'environnement SITI. Les informations devront ensuite être transmises à l'environnement métier sous forme de KPIs significatifs.

La méthode présentée dans ce manuscrit se veut générique. Afin d'être le plus « libre » possible, nos choix technologiques se sont orientés vers le domaine du logiciel libre (« open-source »). Il convient dès lors de s'interroger sur la pérennité des outils utilisés pour constituer notre plateforme, notamment avec le langage de métamodélisation Kermeta. Il s'agira de constater si ce langage est maintenu et suit les évolutions de l'Eclipse Modeling Framework.

#### 13.4.6 Extension de l'approche SCALP au domaine Process Systems Engineering

La piste empruntée par notre démarche et décrite chapitre 12 a pour vocation d'être développée plus en détail. Les travaux menés par (Heinz et al. 2011) vont dans ce sens. Ces travaux s'inscrivent dans le projet InBioSynSolv<sup>28</sup> : « Un laboratoire virtuel pour une chimie durable ». Le but est de développer un prototype logiciel de type *Computer Aided Molecular Design* - CAMD utilisant la formulation inverse afin de rechercher des produits chimiques satisfaisant un cahier des charges prédéfini. Dans le but d'être au plus proche des problèmes à traiter, les solutions recherchées sont des mélanges de molécules. Le cahier des charges intègre à la fois des propriétés fonctionnelles et des propriétés évaluant l'impact environnemental et sanitaire... Le développement logiciel associé à ce projet s'inspire des concepts de notre approche assurant l'alignement des différents modèles du système. Cette conception repose sur différents modèles d'analyse (UML, BPMN) et sur les modèles d'implémentation orientés objet dédiés au framework .NET de Microsoft (C#, VB.NET). Ces travaux porteront également sur la détermination et la formalisation des relations sémantiques entre l'ingénierie des processus et l'ingénierie des procédés.

Ces différentes perspectives sont en accord avec notre volonté d'apporter au Laboratoire de Génie Chimique et en particulier au département « Procédés et Systèmes Industriels » nos connaissances relatives au domaine de l'Ingénierie d'Entreprise et des Systèmes d'Information et de l'ingénierie des processus métier. Le but recherché est d'être en mesure d'appliquer nos idées à l'ingénierie des procédés, en particulier dans un contexte de développement durable et de « chimie verte » (« Green Chemistry » et « Green Process Engineering »).

---

<sup>28</sup> Programme ANR 2008 : ANR-09-CP2D-08, Chimie et Procédés pour le Développement Durable





## A

- Aloui, S. 2007. *Contribution à la modélisation et l'analyse du risque dans une organisation de santé au moyen d'une approche système*. Thèse de doctorat, Ecole des Mines de Paris.
- Alter, S. 1999. "A general, yet Useful Theory of Information Systems", in *Association for Information Systems*, vol. 1.
- American Chemical Society, American Institute of Chemical Engineers, Chemical Manufacturers Association, Council for Chemical Research, & Synthetic Organic Chemical Manufacturers Association 1996. *Technology Vision 2020*.
- AMICE 1993. *CIMOSA: Open System Architecture for CIM*, Second extended and revised version ed. Springer-Verlag, Berlin.
- Amit, R. & Schoemaker, P. J. H. 1993, "Strategic Assets and Organizational Rent," In *Strategic Management Journal*, vol. 14 pp. 33-46.
- Anaya, V., Berio, G., Harzallah, M., Heymans, P., Matulevicius, R., Opdahl, A. L., Panetto, H., & Verdecho, M. J. 2008, *The Unified Enterprise Modelling Language – Overview and Further Work*, IFAC Papersonline.
- Avila Cifuentes, O.J. 2009. *Contribution à l'Alignement Complet des Systèmes d'Information Techniques*. Laboratoire de Génie de la Conception (LGeCo) - Thèse de doctorat, Université de Strasbourg.
- Avison, D., Jones, J., Powell, P., & Wilson, D. 2004, "Using and validating the strategic alignment model," In *Journal of Strategic Information Systems*, vol. 13 pp. 223-246.

## B

- Baina, S. 2006. *Interopérabilité Dirigée par les Modèles: Une Approche Orientée Produit pour l'Interopérabilité des Systèmes d'Entreprise*. Thèse de doctorat, Faculté des Sciences et Techniques.
- Baptiste, J.-L. 2009. *Merise - Guide Pratique - Modélisation des données et des traitements, langage SQL*, Eyrolles.
- Bayer, B., Weidenhaupt, K., Jarke, M., & Marquardt, W. 2001, "A Flowsheet-centered architecture for conceptual design," In *European Symposium on Computer Aided Process Engineering*, vol. 9 pp. 345-350.
- Bayer, B. & Marquardt, W. 2003, "A Comparison of Data Models in Chemical Engineering," In *Concurrent Engineering*, SAGE Publications, ed., pp. 129-138.
- Belaud, J.-P. 2002. *Architectures et technologies des systèmes logiciels ouverts - Cape-Open, un standard pour l'interopérabilité et l'intégration des composants logiciels de l'ingénierie des procédés*. Thèse de doctorat, Institut National Polytechnique (INP) de Toulouse.
- Bézivin, J. & Blanc, X. *MDA: Vers un Important Changement de Paradigme en Génie Logiciel*. Développeur Référence . 2002.
- Bézivin, J. 2004, "Sur les principes de base de l'ingénierie des modèles," In *L'Objet*, vol. 10 pp. 147-157.
- Bidan, M. 2004, "Fédération et intégration des applications du Système d'Information de Gestion," In *Revue Systèmes d'Information et Management (SIM) - n° spécial Risques des projets ERP*, vol. 9.
- Blay-Fornarino, M., Dao, M., Lahire, P., & Rivierre, N. 2008, *Transformations depuis les modèles métier*, Livrable F-2.4.
- Bloch, A. & Krob, D. 2005, *L'Entrepreneur Face au Système d'Information: un Enjeu de Formation ?*, Les Echos.
- Boucher, X. 2007. *Vers un pilotage agile de l'évolution des systèmes de production*. Habilité à Dirigée les Recherches, Ecole Nationale Supérieure des Mines de Saint-Etienne.

Bouchiba, A. & Cherkaoui, A. 2007, *Contribution de la modélisation combinée avec l'approche bayésienne dans l'amélioration des performances des processus métiers - Cas de la sûreté ferroviaire au niveau de l'ONCF.*, CPI'2007.

BPMS.Info. BPMN 2.0: ce que la nouvelle version de la notation va apporter (ou pas). Le Journal du Net, [www.journaldunet.com](http://www.journaldunet.com) - Tribunes BPMS.Info . 25-6-2009.

Broadbent, M. & Weill, P. 1993, "Improving business and information strategy alignment: Learning from the banking industry," *In IBM Systems Journal*, vol. 32 pp. 162-179.

### C

Chan, Y. E., Huff, S. L., & Barclay, D. W. 1997, "Business Strategic Orientation, Information Systems Strategic Orientation and Strategic Alignment," *In Information Systems Research*, vol. 8 pp. 125-150.

Combemale, B. Ingénierie Dirigée par les Modèles (IDM) -- État de l'art. <http://hal.archives-ouvertes.fr/hal-00371565/en/>. 2009.

Cuenca, L., Ortiz, A., & Boza, A. 2010, "Business and IS/IT Strategic Alignment Framework," *In Emerging Trends in Technological Innovation*, vol. 314/2010 IFIP Advances in Information and Communication Technology, pp. 24-31.

### D

Darras, F. 2004. *Proposition d'un Cadre de Référence pour la Conception et l'Exploitation d'un Progiciel de Gestion Intégré*. Thèse de doctorat, École des Mines d'Albi-Carmaux.

Dassiti, M. 2006, *Research directions in enterprise modelling for interoperability and integration* DEM 3 - IST-508 011.

David, R. & Alla, H. 1992. *Du GRAFCET aux Réseaux de Pétri* Paris, Hermès.

Debauche, B. & Megard, P. 2004. *BPM Business Process Management : pilotage métier de l'entreprise* Lavoisier.

### E

Earl, M. J. 1992, "Putting IT in its place: a polemic for the nineties," *In Journal of Information Technology*, vol. 7 pp. 100-108.

Edgar, T. F. 2004, "Control and operations: When does controllability equal profitability?," *In Computer & Chemical Engineering*, vol. 29 Elsevier, ed., pp. 41-49.

Etien, A. 2006. *Ingénierie de l'alignement: Concepts, Modèles et Processus - La méthode ACEM pour l'alignement d'un système d'information aux processus d'entreprise*. Thèse de doctorat, Université Paris I - Panthéon - Sorbonne.

### F

Fathallah, A., Stal-Le Cardinal, J., Ermine, J.L., & Bocquet, J.C. 2010, "Enterprise modelling: building a product lifecycle management model as a component of the integrated vision of the enterprise", *in International journal on Interactive Design and Manufacturing (IJDeM)*, vol. 4, n°3, pp.201-209.

Faucher, C., Bertrand, F., & Lafaye, J.-Y. 2008, "Génération d'ontologie à partir d'un modèle métier UML annoté," *In Revue des Technologies de l'Information - RNTI 12: Modélisation des connaissances*, Cépaduès-édition, ed., pp. 65-84.

Favre, J.-M., Estublier, J., & Blay-Fornarino, M. 2006. *L'ingénierie dirigée par les modèles : au-delà du MDA* Paris, Hermès Science Publications.

Ferchichi, A. 2008. *Contribution à l'intégration des processus métier: Application à la mise en place d'un référentiel qualité multi-vues*. Thèse de doctorat, Ecole Centrale de Lille - Ecole Centrale de Paris.

Fingar, P. & Bellini, J. 2004. *The Real-Time Enterprise* New York, Meghan-Kiffer Press.

Fontan, B. 2008. *Méthodologie de conception de systèmes temps réel et distribués en contexte UML/SysML*. Thèse de doctorat, Université Paul Sabatier - Toulouse III.

## G

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. 1999. *Design Patterns - Catalogue de modèles de conception réutilisables* Vuibert.

Garcês, R., Cardoso, J., & Valente, P. 2009, "Open Source Workflow Management Systems: A Consive Survey," *In 2009 BPM & Workflow Handbook - Spotlight on BPM in Government*, L. Fischer, ed., WfMC, pp. 333-346.

Godart, C. & Perrin, O. 2009. *Les processus métiers - Concepts, modèles et systèmes*, 1ère édition ed. Hermès - Lavoisier.

Gordon, J. R. & Gordon, S. R. 2000, "Structuring the interaction between IT and business units: Prototypes for service delivery," *In Information System Management*, vol. 17 pp. 7-16.

Grangel, R., Ben Salem, R., 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 - New Challenges and Approaches*, Springer London, ed., pp. 447-458.

Grossmann, I. E. 2004, "Challenges in the new millennium: product discovery and design, enterprise and supply chain optimization, global life cycle assessment," *In Computers & Chemical Engineering*, vol. 29 Elsevier, ed., pp. 29-39.

Gruber, T. 1993. *A Translation Approach to Portable Ontology Specifications*, Vol.5 ed. Knowledge Acquisition.

Guarino, N. 1998. "Formal Ontology in Information Systems", *in Proceedings of FOIS'98*, Trento, Italie, pp. 3-15.

## H

Havey, M.V. & Havey, M. 2005. *Essential Business Process Modeling*, O'Reilly Media, Incorporated.

Hay, D. & Healy, K. A. 1997, *GUIDE - Business Rule Project Final Report*.

Heinz, J., Gerbaud, V., & Belaud, J.-P. 2011, "Models driven conception of an inverse formulation software tool," en cours de soumission *In 21st European Symposium on Computer Aided Process Engineering - ESCAPE 21*, Elsevier.

Henderson, J. C. & Venkatraman, N. 1993, "Strategic alignment: Leveraging information technology for transforming organizations," *In IBM Systems Journal*, vol. 32 pp. 3-16.

Henderson, J. C. & Venkatraman, N. 1999, "Strategic Alignment: Leveraging information technology for transforming organizations," *In IBM Systems Journal*, vol. 38 n°2 IBM, ed., IBM, pp. 472-484.

Hettel, T. 2010. *Model Round-Trip Engineering*. Faculty of Science and Technology.

Hirao, M., Sugiyama, H., Fischer, U., & Hungerbülher, K. 2008, "IDEFO Activity Modeling for Integrated Process Design Considering Environmental, Health and Safety (EHS) Aspects," *In 18th European Symposium on Computer Aided Process Engineering - ESCAPE18*, B. Braunschweig & X. Joulia, eds., Elsevier, pp. 1065-1070.

Hollingsworth, D. 2004, "The Workflow Reference Model : 10 Years On," Workflow Management Coalition.

## I

IEEE 2000, *Recommended Practice for Architectural Description of Software Intensive* IEEE-std-1471-2000.

IFAC-IFIP Task Force 1997, *GERAM: Generalized Enterprise Reference Architecture and Methodology* (patent).

ISO TC 184/SC 5 2000, *ISO 15704:2000 - Industrial Automation Systems - Requirements for enterprise-reference architectures and methodologies* (patent).

Izza, S. 2006. *Intégration des Systèmes d'Information Industriels - Une Approche Flexible Basée sur les Services Sémantiques*. École Nationale Supérieure des Mines de Saint-Etienne.

### J

Jablonski, S. 2009, "Process Modeling for Holistic Process Management," *In Handbook of Research on Business Process Modeling*, J. Cardoso & W. M. van der Aalst, eds., Information Science, pp. 49-68.

### K

Kadima, H. 2005. *MDA : conception orientée objet guidée par les modèles*, Dunod, Paris.

Klatt, K.-U. & Marquardt, W. 2009, "Perspectives for process systems engineering - Personal views from academia and industry," *In Computers & Chemical Engineering*, vol. 33 Elsevier, ed., pp. 536-550.

Klein, J. & Fleurey, F. Tissage d'aspects comportementaux, *In Langages et Modèles à Objets - LMO'06*, Hermès Lavoisier, pp. 101-116.

### L

LAAS-CNRS 2003, *Rapport final de l'Action Spécifique n°35 - AS PRODLOG "Production et logistique dans l'entreprise étendue: modèles et outils collaboratifs"* INSA Toulouse.

Laleau, R. 2002. *Conception et Développement Formels d'Applications Bases de Données*. Université d'Evry Val-d'Essonne.

Le Moigne, J.-L. 1984. *La Théorie du Système Général*, Edition Presse Universitaire Française.

Leist, S. & Zellner, G. Evaluation of Architecture Frameworks, *In Organizational engineering (OE)*, ACM, pp. 1546-1553.

Lemoigne, J.-L. 1984. *La théorie du système général*, Edition Presse Universitaire Française.

Lequeux, J.-L.W. 2008. *Manager avec les ERP -- Architecture Orientée Services (SOA)*, 3ème édition ed. Editions d'organisation.

List, B. & Korherr, B. An evaluation of conceptual business process modelling languages, *In Symposium on Applied computing'06*, New York, NY, USA: ACM, pp. 1532-1539.

Luftman, J. & Brier, T. 1999, "Achieving and Sustaining Business-IT Alignment," *In California Management Review*, vol. 42 pp. 109-122.

Luftman, J. & Maclean, E. R. 2004, "Key issues for IT executives," *In MIS Quaterly Executive*, vol. 4, n°2 University of Minnesota, ed., University of Minnesota.

### M

Malone, T.W., Crowston, K., & Herman, G.A. 2003. *Organizing Business Knowledge - The MIT Process Handbook*, MIT Press.

Marquardt, W., Morbach, J., & Yang, A. 2010. *OntoCape - A Re-Usable Ontology for Chemical Proces Engineering*.

Marquardt, W., von Wedel, L., & Bayer 1999, Perspectives on lifecycle process modeling, *In FOCAPD '99: 5th international conference on foundations of computer-aided process design*.

Martin, R. A., Robertson, E. L., & Springer, J. A. 2004, *Architectural Principles for Enterprise Frameworks*, Computer Science Department, Indiana University, Bloomington, Indiana.

Mending, J., Newman, J., & Nüttgens, M. 2004. "A Comparaison of XML Interchange Formats for Business Process Modelling", *In EMISA 2004*.

Morley, C., Hugues, J., Leblanc, B., & Hugues, O. 2007. *Processus métiers et S.I. – Evaluation, modélisation, mise en œuvre*, DUNOD.

Mosser, S. 2010. *Behavioral Compositions in Service-Oriented Architecture*. Thèse de doctorat, Université de Nice-Sophia Antipolis.

Mosser, S. & Blay-Fornarino, M. Réflexions autour de la construction dirigée par les modèles d'un atelier de composition d'orchestrations, *In Langages et Modèles à Objets - LMO'09*.

Muller, P.-A., Fleurey, F., & Jézéquel, J.-M. 2005, Weaving Executability into Object-Oriented Meta-Languages, *In The 8th International Conference on Model Driven Engineering Languages and Systems (MODELS/UML'2005)*.

## N

National Institute of Standards and Technology (NIST) 1993, *Integration Definition for Function Modeling - IDEF0*, 183 (patent).

## O

OASIS 2007, *WSBPEL - Web Service Business Process Execution Language* (patent).

OMG 2003, *BPDM - Business Process Definition Metamodel - Request for Proposal* (patent).

OMG 2006, *Meta Object Facility (MOF) Core Specification - Version 2.0* (patent).

OMG 2007a, *UML 2.1.2 Superstructure specification* (patent).

OMG 2007b, *XML Metadata Interchange (XMI) - 2.1.1 - ISO/IEC 19503* (patent).

OMG 2008, *BPMN - Business Process Modeling Notation Specification - OMG Final Adopted Specification* (patent).

Ouyang, C., Van der Aalst, W. M. P., Dumas, M., & Ter Hofstede, A. H. M. 2006, *Translating BPMN to BPEL*.

## P

Palmer, N. Understanding the BPMN-XPDL-BPEL Value Chain. *Business Integration Journal* . 2006.

Panetto, H. 2006. *Méta-modèles et Modèles pour l'Intégration et l'Interopérabilité des Applications d'Entreprises de Production*. Habilité à Dirigée les Recherches, Université Henri Poincaré - Nancy 1.

Panetto, H. 2007. Towards a classification framework for interoperability of enterprise applications. *International Journal of Computer Integrated Manufacturing*, 20, (8) 727-740 available from: <http://portal.acm.org/citation.cfm?id=1392537#>

Panetto, H., Berio, G., Benali, K., Boudjlida, N., & Petit, M. A Unified Enterprise Modelling Language for enhanced interoperability of Enterprise Models, *In IFAC INCOM2004*.

Peppard, J. & Ward, J. 2004, "Beyond Strategic Information Systems: Toward an IS Capability," *In Journal of Strategic Information Systems*, vol. 13 pp. 167-194.

Perez, J. M., Ruiz, F., & Piattini, M. 2006, MDE for BPM: A systematic Review, *In First International Conference of Software and Data Technologies, ICSOFT 2006*, J. Filipe, B. Shiskov, & M. Helfert, eds., Springer.

Pourcel, C. & Gourc, D. 2005. *Modélisation d'entreprise par les processus: activités, organisation et applications* Toulouse, Cépaduès-édition.

Pyke, J. 2006, "BPM in Context: Now and in the Future," *In Workflow Handbook*, L. Fisher, ed., Workflow Management Coalition, pp. 17-28.

## R

Rajsiri, V. 2009. *Knowledge-based system for collaborative process specification*. Thèse de doctorat, Ecole des Mines d'Albi-Carmaux.

Reix, R. 2004. *Systèmes d'Information et Management des Organisations* Edition VUIBERT.

Roboam, M. 1993. *La Méthode GRAI, Principes, Outils, Démarche et Pratique* Toulouse, Teknéa.

Roques, P. & Vallée, F. 2004. *UML2 en action: De l'analyse des besoins à la conception J2EE* Eyrolles.

Roser, S. & Bauer, B. A categorization of collaborative business process modeling techniques, *In Seventh IEEE International Conference 2005*, pp. 43-51.

Ross, R.G. 2003. *Principles of the Business Rule Approach* Boston, Addison-Wesley.

## S

Sadiq, R., Kleiner, Y., & Rajani, B. 2004, "Aggregative risk analysis for water quality failure in distribution network," *In Journal of Water Supply: Research and Technology - AQUA*, vol. 53(4) pp. 241-261.

Scheer, A.-W. 2002. *ARIS : des processus de gestion au système d'intégré d'applications* Springer-Verlag France.

Schneider, R. & Marquardt, W. 2002, "Information technology support in the chemical process design life cycle," *In Chemical Engineering Science*, vol. 57 pp. 1763-1792.

Silver, B. 2008, *BPMS Watch Ratings for Q2 2008*, BPMS Watch.

Silver, B. 2009. *BPMN, Method and Style* Cody-Cassidy press.

Silvius, D. A. J. G. Business & IT Alignment in theory and practice, *In HICSS'07*.

Smith, H. 2003. *BPM and MDA: Competitors, Alternatives or Complementary* Business Process Trends.

Smith, H., Neal, D., Ferrara, L., & Hayden, F. 2002. *The Emergence of Business Process Management* CSC's Research Services.

SPINOV 2006, *Modélisation des Processus Métiers: Etat de l'Art et Conseils Pratiques*, Rapport CITI.

Steel, J. & Jézéquel, J.-M. 2004, "Typing Relationship in MDA," *In Technical Report - University of Kent at Canterbury Computing Laboratory*, University of Kent, pp. 154-159.

Stein, S., Kühne, S., & Ivanov, K. 2008, Business to IT Transformations Revisited, *In BPM 2008*.

Steinberg, D., Budinsky, F., Paternostro, M., & Merks, E. 2009. *EMF, Eclipse Modeling Framework*, 2nd ed. Boston.

Stevens, P. 2008, "A Landscape of Bidirectional Model Transformations," *In Generative and Transformational Techniques in Software Engineering II*, vol. Volume 5235/2008 Springer Berlin / Heidelberg, ed., pp. 408-424.

## I

Takamatsu, T. 1983, "The nature and role of process system engineering," *In Computers & Chemical Engineering*, vol. 7 pp. 203-218.

Tardieu, H., Rochfeld, A., & Rolland, C. 2002. *La Méthode MERISE: Principes et Outils* Editions de l'Organisation.

The Business Rules Group (BRG) 2007, *The Business Motivation Model - Business Governance in a Volatile World*, Release 1.3.

Théroude, F., Braesch C., Haurat A. 2001, "OLYMPIOS: un modèle pour le pilotage des processus", *in MOSIM'01 : conférence francophone de modélisation et simulation*, Troyes, France, 249-253.

Thevenet, L.-H. 2010. *Proposition d'une modélisation conceptuelle d'alignement stratégique : La méthode INSTAL*. Thèse de doctorat, Université Panthéon-Sorbonne.

Touzi, J. 2007. *Aide à la Conception de Système d'Information Collaboratif Support de l'Interopérabilité des Entreprises*. Thèse de doctorat, Ecole des Mines d'Albi Carmaux.

Touzi, J., Bénaben, F., & Pingaud, H. 2008, "Prototype to Support Morphism between BPMN Collaborative Process Model and Collaborative SOA Architecture Model," *In Enterprise Interoperability III New Challenges and Industrial Approaches*, Springer London, pp. 145-157.

Truong Meyer, X.-M. 2009, "Modélisation en génie des procédés," *In Techniques de l'ingénieur - Génie des procédés*, vol. JB1, n°J1021 [Note(s): J1021.1-J1021.18].

## U

Ulmer, J.-S., Belaud, J.-P., & Le Lann, J.-M. 2008, Analyse de l'expressivité des langages de modélisation de processus par les patterns, *In Inforsid'08*, Fontainebleau, France.

Ulmer, J.-S., Belaud, J.-P., & Le Lann, J.-M. 2009, Proposition d'une méthodologie générique pour la formalisation et l'implémentation des processus, *In XXVIIème congrès INFORSID*, Toulouse, France, pp. 43-58.

Ulmer, J.-S., Belaud, J.-P., & Le Lann, J.-M. 2010a, "Proposition d'une approche générique pour la formalisation et l'implémentation des processus," *In Networking and Information Systems - Revue des sciences et technologies de l'information*, vol. 15, n°4/2010.

Ulmer, J.-S., Belaud, J.-P., & Le Lann, J.-M. 2010b, Towards a pivotal-based approach for Business process alignment, *In 8<sup>th</sup> International Conference of Modeling and Simulation - MOSIM'10 - "Evaluation and optimization of innovative production systems of goods and services"*.

Uschold, M. & Gruninger, M. 1996. Ontologies: Principles, Methods and Applications. *Knowledge Engineering Review*, Cambridge University Press, vol.11-02, pp.93-136

## V

Van der Aalst, W. M. P. 2004, "Business Process Management Demystified: A Tutorial on Models, systems and Standards for Workflow Management," *In Lectures in Concurrency and Petri Nets*, Springer Berlin Heidelberg, ed., pp. 21-58.

Various IIBA & Brennan, K. 2008. *A Guide to the Business Analysis Body of Knowledge (BABOK)*, v.1.6. ed. Canada, International Institute of Business Analysis.

Varma, V. A., Reklaitis, G. V., Blau, G. E., & Pekny, J. F. 2007, "Enterprise-wide modeling & optimization—An overview of emerging research challenges and opportunities," *In Computers & Chemical Engineering*, vol. 31 pp. 692-711.

Vernadat, F.B. 1999. *Techniques de Modélisation en Entreprise: Applications aux Processus Opérationnels* Paris, Economica.

Vernadat, F. & Hamaidi, L. 1998. *La modélisation en entreprise: Méthodes descriptives des processus opérationnels* Paris, Economica.

Vernadat, F. B. 2007, "Interoperable enterprise systems: Principles, concepts, and methods," *In Annual Reviews in Control*, vol. 31 Elsevier, ed., pp. 137-145.

von Halle, B. 2002. *Business Rules Applied*, John Wiley & Sons, New York.

### W

Wagner, H.-T., Beimborn, D., Franke, J., & Weitzel, T. IT Business Alignment and IT Usage in Operational Processes: A Retail Banking Case, *In 39th Hawaii International Conference on Systems Science*, IEEE Computer Society.

Ward, J. & Peppard, J. 2002. *Strategic Planning for Information Systems* John Wiley & Sons, Ltd, 3rd Edition.

Weidlich, M., Decker, G., Großkopf, A., & Weske, M. BPEL to BPMN: The Myth of a Straight-Forward Mapping, *In OTM 2008*, Berlin: Springer-Verlag, pp. 265-282.

WfMC 1999, *Workflow Management Coalition - Terminology and Glossary - WfMC-TC-1011* (patent).

### Y

Yang, A., Braunschweig, B., Fraga, E. S., Guessoum, Z., Marquardt, W., Nadjemi, O., Paen, D., Piñol, D., Roux, P., Sama, S., Serra, M., & Stalker, I. 2008, "A multi-agent system to facilitate component-based process modeling and design," *In Computers & Chemical Engineering*, vol. 32 Elsevier, ed., pp. 2290-2305.

### Z

Zur Muehlen, M. 2004. *Workflow-based Process Controlling Foundation, Design, and Application of Workflow-driven Process Information Systems* Berlin, Logos Verlag.

Zur Muehlen, M. *Business Process Management Standards - An Overview*. 2008.

Zur Muehlen, M. & Recker, J. 2008, How Much Language is Enough? Theoretical and Practical Use of the Business Process Modeling Notation, *In 20th International Conference on Advanced Information Systems Engineering (CAISE 2008)*, Springer, ed..





## 15.1 GLOSSAIRE

<u>A</u>		
Agilité	Définit la capacité d'une entreprise à s'adapter rapidement à son environnement, à procéder à des changements harmonieux et continus de l'organisation.	..... 27
Ajustement stratégique	Voir Domaines externes, domaines internes	..... 25
Alignement opérationnel	Maintenir un alignement opérationnel revient à conserver une cohérence sémantique et structurelle entre modèles hétérogènes et d'abstraction/granularité différentes. Elle est souvent perçue comme l'intégration fonctionnelle.	..... 27
Alignement stratégique	Amélioration de la performance organisationnelle à travers divers mécanismes comme les processus de pilotage, les ressources humaines et les capacités technologiques, où ces capacités peuvent être interprétées essentiellement comme la capacité à déployer des ressources.	..... 25
Analyse des processus métier	Ou <i>Business Process Analysis</i> – BPA. Etapes de conception et de modélisation des processus réalisées par l'analyste métier.	..... 62
Analyste métier	Acteur recherchant de nouvelles façons d'améliorer l'efficacité « métier » de son entreprise. Cette amélioration peut être réalisée en modifiant la manière de travailler collectivement, en changeant d'outils ou de processus	..... 49
Architecte des processus métier	Concilie les deux niveaux d'abstractions propres à l'analyste métier et à l'expert SITI, l'acteur « pivot » permet de « fluidifier » les échanges et d'obtenir une meilleure cohérence entre différents modèles (d'analyse et technique). Il possède les connaissances fonctionnelles et technologiques nécessaires pour définir un « style architectural » BPM à appliquer au sein de l'organisation.	..... 92
Architecture Dirigée par les Modèles	Ou <i>Model-Driven Architecture</i> - MDA. Approche mettant en avant l'utilisation systématique de modèles comme support à la conception et au développement de différents types de systèmes. MDA utilise trois types de modèles situés à niveaux d'abstraction différents : CIM, PIM et PSM.	..... 55
<u>C</u>		
Cadre de modélisation	Approche, incluant un ensemble de composants - modèles et définitions - formant un squelette adaptable et/ou modifiable au domaine d'application de l'entreprise et ce dans le but de développer et documenter les descriptions d'architecture.	..... 40
Chorégraphie	Permet de composer une collaboration entre plusieurs entités autonomes. Le comportement global émerge en fonction de l'interaction et du résultat de chaque processus (entités).	..... 114
Cohérence intermodèle	Il existe une cohérence dite « intermodèle » entre deux modèles si nous observons une équivalence sémantique et s'ils sont synchronisés entre eux.	..... 80
<i>Computation Independent Model - CIM</i>	Modèles abstraits associés aux exigences d'un système et/ou à un domaine métier. Le CIM décrit usuellement l'environnement, les processus métier et objets ainsi que les exigences spécifiques du système. Il permet de définir les règles et le vocabulaire métier et représente l'aspect organisationnel du système.	..... 55

Couplage faible	Définit une interaction forte où les modèles demeurent « autonomes » et les environnements associés restent modifiables.	.....	72
Cycle de vie	Représentation des différents niveaux de dérivation d'un modèle. Il décrit le modèle depuis « l'énoncé des prescriptions pour arriver à un modèle traitable par informatique ».	.....	40
<u>D</u>			
Démarche en Y	Ou <i>2 Tracks Unified Process</i> . Processus de développement logiciel pour la description de l'architecture logicielle en dissociant les aspects fonctionnels (approche par les fonctionnalités) des aspects technique (étude de la mise en œuvre).	.....	55
Domaines externes	Le domaine externe porte sur la formulation des stratégies reflétant l'environnement dans lequel s'inscrit l'entreprise. Le domaine interne se préoccupe de l'infrastructure SITI et des processus d'entreprise.	.....	25
Domaines internes	L'ajustement stratégique décrit la relation verticale existant entre un domaine externe et le domaine interne qui lui est associé. L'intégration fonctionnelle représente la relation horizontale existant entre deux domaines.		
<u>E</u>			
Entreprise	Système sociotechnique ouvert et dynamique motivé par un ou plusieurs buts et objectifs.	.....	35
Equivalence sémantique	Les modèles i et j sont sémantiquement équivalents si : Pour chaque élément appartenant à $E_i$ , un élément (ou un ensemble d'éléments) appartenant à $E_j$ peut lui être associé ; Chacun de ces éléments associés suit la même orchestration.	.....	79
Expert SITI	Acteur prenant en charge les besoins métier définis par l'analyste métier et les transforme selon des considérations techniques.	.....	49
<u>F</u>			
Flexibilité	Est semblable à la notion de réactivité industrielle. Elle traduit la capacité d'une entreprise à répondre rapidement aux besoins de ses clients par une utilisation différente de ses ressources.	.....	27
<u>G</u>			
Généricité relative	Généricité limitée à un domaine métier (services bancaires, procédés physico-chimiques...), à un contexte d'étude et au niveau d'abstraction recherché.	.....	92
<u>I</u>			
Implémentation des processus métier	Ou <i>Business Process Implementation</i> – BPI. Etapes d'implémentation et d'exécution des processus réalisées par l'expert SITI.	.....	64
Indicateur clé de performance	Ou <i>Key Performance Indicator</i> – KPI. Métriques calculés sur la base des mesures obtenues depuis le BPI, ils permettent de mesurer la performance et l'atteinte des buts organisationnels fixés.	.....	66
Ingénierie des processus métier	Ou <i>Business Process Management</i> – BPM. Permet de modéliser, déployer, exécuter et optimiser de manière continue les différents types de processus et ainsi d'améliorer l'agilité d'une organisation à l'aide des technologies de l'information	.....	49

Ingénierie Dirigée par les Modèles - IDM	Ou <i>Model-Driven Engineering, MDE</i> . Approche intégrative générale mettant à disposition des outils, concepts et langages pour créer et transformer des modèles.	.....	53
Intégration fonctionnelle	Voir Domaines externes, domaines internes	.....	25
<u>M</u>			
<i>Meta-Object Facility</i> - MOF	Socle sur lequel les langages de modélisation se basent à travers la couche d'abstraction : méta-métamodèle.	.....	57
Méthode de Modélisation d'Entreprise - MME	Décrit les aspects comportementaux (notamment la gestion des évènements), fonctionnels et dynamiques, ainsi que la formalisation des savoir-faire d'une entreprise.	.....	38
Métier	Ensemble d'activités d'un domaine donné nécessitant des compétences et savoir-faire des acteurs de l'entreprise.	.....	34
Modèle d'analyse	Modèle conceptuel représentant le processus métier selon un point de vue « monde réel », s'exprimant à l'aide de termes métier exprimés généralement dans un langage naturel.	.....	81
Modèle d'implémentation	Modèle technique adoptant un point de vue « informatique » et utilisant des notions et de termes propres aux SITI.	.....	66
Modélisation d'entreprise	A pour objet la construction du modèle de tout ou partie de l'entreprise. L'entreprise est alors vue comme un système et sa modélisation doit en expliquer la structure, l'organisation et le fonctionnement.	.....	38
<u>O</u>			
Orchestration	Définit l'ensemble des étapes internes à un processus incluant conditions et exceptions.	.....	34
<u>P</u>			
<i>Platform Independent Models</i> - PIM	Décrivent tout ou partie des fonctionnalités du système modélisé sans se soucier des détails techniques. Modèle conceptuel indépendant de toutes considérations liées à la plateforme cible, à son langage ou à la technologie usitée.	.....	55
<i>Platform Specific Models</i> - PSM	Modèle associé à une plateforme spécifique basée sur une technologie bien définie.	.....	55
Processus d'entreprise	Ensemble d'activités, entreprises dans un objectif déterminé.	.....	23
Processus de mesure	Fournissent les métriques nécessaires à l'évaluation des processus et à leur amélioration continue.	.....	41
Processus de pilotage	Ou processus de management. Ils ont pour but d'organiser les objectifs stratégiques de l'entreprise.	.....	25
Processus de support	Ou processus de soutien. Périphériques au métier de l'entreprise, ils ne participent qu'indirectement à l'accomplissement d'un objectif métier.	.....	44
Processus métier	Orchestration d'activités, incluant une interaction entre différents acteurs sous la forme d'échange d'informations, réalisant des objectifs métier.	.....	34
Processus opérationnel	a pour fonction d'accomplir une mission dans un domaine donné et utilise plusieurs fonctions de l'entreprise.	.....	44

Progiciel de Gestion Intégré – PGI	Ou <i>Enterprise Resource Planning</i> – ERP. Solution logiciel gérant l'ensemble des processus opérationnels par l'intégration de l'ensemble des fonctions du processus considéré.	.....	44
<u>R</u>			
Règle métier	Formulation qui permet de structurer, de contrôler ou d'influencer le comportement du métier.	.....	83
<u>S</u>			
Suite BPM	Ou <i>BPM Suite</i> – BPMS. Plateforme intégrée d'un éditeur unique rassemblant tous les composants nécessaires au développement et à l'exécution d'une solution BPM : modélisation et analyse, orchestration automatisée, tâches humaines, intégration des applications, règles métier, et supervision.	.....	68
Supervision des processus métier	Ou <i>Business Activity Monitoring</i> – BAM. Fournit un accès en temps réel à un ensemble d'indicateurs d'analyse de performance, <i>Key Performance Indicators</i> (KPI) et procure un contrôle et une amélioration constants des processus métier.	.....	66
Synchronisation	Deux modèles, i et j, sont dits synchronisés si et seulement si des changements significatifs effectués sur le modèle i peuvent être répercutés sur le modèle j. Est considéré comme significatif tout changement modifiant la structure et/ou le comportement d'un modèle.	.....	28
Système d'Information – SI	Représente la partie réelle constituée d'informations organisées et d'acteurs qui agissent sur ces informations ou à partir de ces informations, selon des processus visant une finalité de gestion et utilisant des technologies de l'information.	.....	35
Système de Gestion des règles métier	Ou Business Rules Management Suite – BRMS. Fournit une suite de fonctionnalités typiques pour la gestion des règles métier.	.....	83
Système workflow	Définit, crée et gère l'exécution de workflows à l'aide d'un logiciel capable d'interpréter la définition du processus, interagir avec les participants du workflow et est capable d'utiliser des outils SITI.	.....	62
<u>V</u>			
Vue	Représentation de tout un système selon la perspective d'un ensemble d'intérêts liés.	.....	41
Vue comportementale	Ou <i>control-flow</i> . Définit les dépendances entre processus et donc l'ordre dans lequel ces processus sont exécutés. Est parfois assimilé à la vue fonctionnelle.	.....	41
Vue des ressources	ou vue opérationnelle. Elle spécifie les outils ou les systèmes permettant la réalisation du processus en décrivant les moyens humains et matériels nécessaires, ainsi que le mode de gestion de ces ressources.	.....	41
Vue fonctionnelle	Description des processus et de leurs structures. Elle identifie les noms, buts et actions des processus.	.....	41
Vue informationnelle	ou <i>data-flow</i> . Cette vue indique quels sont les documents et données utilisés à chaque étape d'un processus. Elle décrit les objets du système, leurs relations et leurs différents états possibles.	.....	41
Vue organisationnelle	Définit les agents responsables et aptes à exécuter le processus.	.....	41

W

Workflow	Automatisation totale ou en partie des processus métier, ou les documents, informations ou tâches sont distribués d'un participant à un autre selon un ensemble de règles procédurales.	.....	62
----------	---	-------	----

**15.2 TRADUCTION DES TERMES ANGLOPHONES**

La liste suivante présente une traduction ou du moins une équivalence en français des différents termes anglophones rencontrés tout au long du manuscrit.

#

<i>2 Tracks Unified Process</i>	.....	Démarche en Y
---------------------------------	-------	---------------

A

<i>As-is</i>	.....	Processus existant
<i>As-wish</i>	.....	Processus désiré
<i>Atlas Transformation Language</i>	.....	Langage de transformation ATLAS

B

<i>Back-office</i>	.....	Arrière-boutique
<i>Block-Flow Diagram</i>	.....	Diagramme de procédés préliminaire
<i>Bloc-oriented</i>	.....	Orienté en bloc
<i>Bottom-up</i>	.....	Approche ascendante
<i>BPMN Common Core</i>	.....	Noyau commun de BPMN
<i>BPMN Extended Core</i>	.....	Noyau étendu de BPMN
<i>Business Activity Monitoring</i>	.....	Supervision des processus métier
<i>Business Process Analysis</i>	.....	Analyse des processus métier
<i>Business Process Diagram</i>	.....	Diagramme de processus métier
<i>Business Process Implementation</i>	.....	Implémentation des processus métier
<i>Business Process Management</i>	.....	Ingénierie des processus métier
<i>Business Process Management Suite</i>	.....	Suite d'outils dédiée à l'ingénierie des processus métier
<i>Business Process Modelling</i>	.....	Modélisation des processus métier
<i>Business Rules Management</i>	.....	Gestion des règles métier
<i>Business Rules Management System</i>	.....	Système de gestion des règles métier

C

<i>Computation Independent Model</i>	.....	Modèle métier indépendant de l'informatisation
<i>Computer-Aided Modelling and Process Design</i>	.....	Conception et modélisation des processus assistées par ordinateur
<i>Computer Aided Molecular Design</i>	.....	Conception moléculaire assistée par ordinateur
<i>Control-flow</i>	.....	Flux de contrôle

D

<i>Data flow</i>	.....	Flux de données
<i>Diagnosis</i>	.....	Diagnostic
<i>Document Type Definition</i>	.....	Définition de type de document
<i>Domain-Specific Modeling Language</i>	.....	Langage de modélisation dédié à un domaine spécifique

	<u>E</u>	
<i>Enterprise Application Integration</i>	.....	Intégration d'Application d'Entreprise
	<u>F</u>	
<i>Framework</i>	.....	Cadre
	<u>H</u>	
<i>Human-centric</i>	.....	Basé sur l'humain
	<u>I</u>	
<i>Integration-centric</i>	.....	Basé sur intégration
	<u>K</u>	
<i>Key Performance Indicators</i>	.....	Indicateur clé de performance
	<u>L</u>	
<i>Legacy software</i>	.....	Logiciel hérité
	<u>M</u>	
<i>Management</i>	.....	Gestion ou ingénierie
<i>Model Driven Architecture</i>	.....	Architecture dirigée par les modèles
<i>Model portability conformance class</i>	.....	Classe de conformité de la portabilité du modèle
<i>Model-Driven Engineering</i>	.....	Ingénierie dirigée par les modèles
<i>Model-Driven Interoperability</i>	.....	Interopérabilité dirigée par les modèles
	<u>P</u>	
<i>Pattern</i>	.....	Patron (au sens de « modèle réutilisable »)
<i>Piping and Instrumentation Diagram</i>	.....	Schéma tuyauterie et instrumentation
<i>Platform Independent Model</i>	.....	Modèle d'un système métier indépendant de la plateforme technologique
<i>Platform Model</i>	.....	Modèle générique de la plateforme technologique
<i>Platform Specific Model</i>	.....	Modèle spécifique de la plateforme technologique
<i>Process Design</i>	.....	Conception des processus
<i>Process-driven engineering</i>	.....	Ingénierie dirigé par les processus
<i>Process Enactment</i>	.....	Promulgation des processus (au sens application/exécution)
<i>Process Systems Engineering</i>	.....	Ingénierie des systèmes de procédés
<i>Process-Flow Sheet</i>	.....	Schéma de principe de procédé
	<u>S</u>	
<i>Sequence-flow</i>	.....	Flux séquentiel
<i>Strategic Alignment Model</i>	.....	Modèle d'alignement stratégique
<i>System Configuration</i>	.....	Configuration du système
	<u>T</u>	
<i>To-be</i>	.....	Processus cible
<i>Top-down</i>	.....	Approche descendante

U

<i>Unified Enterprise Modeling Language</i>	.....	Langage unifié de modélisation d'entreprise
<i>Unit operation</i>	.....	Opération unitaire

W

<i>Workflow</i>	.....	Workflow, automatisation des processus
<i>Workflow Management Coalition</i>	.....	Coalition de gestion de workflow
<i>Workflow Management System</i>	.....	Système de workflow

**15.3 ACRONYMES**#

2TUP	.....	2 Track Unified Process
------	-------	-------------------------

A

ARIS	.....	Architecture of Integrated Information Systems
AS	.....	Action Spécifique

B

BAM	.....	Business Activity Monitoring
BFD	.....	Block-Flow Diagram
BPA	.....	Business Process Analysis
BPD	.....	Business Process Diagram
BPEL	.....	Business Process Execution Language
BPI	.....	Business Process Implementation
BPM	.....	Business Process Management
BPMN	.....	Business Process Modeling Notation
BPMS	.....	Business Process Management Suite
BPMS-HC	.....	Business Process Management Suite – Human Centric
BPMS-IC	.....	Business Process Management Suite – Integration Centric

C

CAMD	.....	Computer Aided Molecular Design
CAPE	.....	Computer-Aided Process Engineering
CIM	.....	Computational Independant Model

D

DSML	.....	Domain-Specific Modeling Language
DTD	.....	Document Type Definition

E

EAI	.....	Enterprise Application Integration
EEML	.....	Extended Enterprise Modeling Language
EMF	.....	Eclipse Modeling Framework
EMOF	.....	Eclipse Meta-Object Facility
ERP	.....	Enterprise Resource Planning

G

GDR	.....	Groupe De Recherche
-----	-------	---------------------

I

I3	.....	Information-Interaction-Intelligence
IDM	.....	Ingénierie Dirigée par les Modèles
IEM	.....	Integrated Enterprise Modeling

IESI	.....	Ingénierie d'Entreprise et des Systèmes d'Information
ISO	.....	International Organization for Standardization
<u>J</u>		
J2EE	.....	Java Enterprise Edition
<u>K</u>		
KPI	.....	Key Performance Indicator
<u>M</u>		
MACS	.....	Modélisation, Analyse et Conduite des Systèmes Dynamiques
MDA	.....	Model-Driven Architecture
MDI	.....	Model-Driven Interoperability
MME	.....	Méthode de Modélisation d'Entreprise
MOF	.....	Meta-Object Facility
<u>O</u>		
OCL	.....	Object Constraint Language
OMG	.....	Object Management Group
<u>P</u>		
PFS	.....	Process-Flow Sheet
PIM	.....	Platform-Independent Model
PM	.....	Platform Model
PSE	.....	Process Systems Engineering
PSM	.....	Platform Specific Model
<u>S</u>		
SAM	.....	Strategic Alignment Model
SCALP	.....	Solution pour la Cohérence et l'Alignement des Processus
SGBD	.....	Système de Gestion de Base de Données
SI	.....	Système d'Information
SIC	.....	Système d'Information Collaboratif
SITI	.....	Système et Technologies d'Information
<u>T</u>		
TI	.....	Technologie d'Information
<u>U</u>		
UEML	.....	Unified Enterprise Modelling Language
UML	.....	Unified Modeling Language
<u>W</u>		
WfMC	.....	Workflow Management Coalition
WFMS	.....	Workflow Management System
<u>X</u>		
XMI	.....	XML Metadata Interchange
XML	.....	eXtensible Markup Language
XPDL	.....	XML Process Definition Language
XSD	.....	XML Schema
XSLT	.....	eXtensible Stylesheet Language Transformation

## 15.4 DEMONSTRATION : EXTRAITS DE FICHIERS

### 15.4.1 Extrait du fichier *m<sub>BPA</sub>*

```

<?xml version="1.0" encoding="UTF-8"?>
<bpmn:BpmnDiagram xmlns:bpmn="http://ensiacet.org/lgc-psi/bpmn" xmlns:xmi="http://www.omg.org/XMI"
xmlns:ecore="http://www.eclipse.org/emf/2002/Ecore" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmi:version="2.0" xmi:id="_VbJHkO_0Ed-zNqI9Y599ag" iD="_VblggO_0Ed-zNqI9Y599ag"
xsi:schemaLocation="http://ensiacet.org/lgc-psi/bpmn ../metamodels/BPA/intalio.ecore">
  <pools xmi:type="bpmn:Pool" xmi:id="_VcnHMe_0Ed-zNqI9Y599ag" iD="_VcnHMO_0Ed-zNqI9Y599ag"
name="Département Beauté et Industrie">
    <eAnnotations xmi:type="ecore:EAnnotation" xmi:id="_wx7yMO_3Ed-zNqI9Y599ag" source="executablepool">
      <details xmi:type="ecore:EStringToStringMapEntry" xmi:id="_wx7yMe_3Ed-zNqI9Y599ag" key="poolsExecutable"
value="false" />
    </eAnnotations>
    <lanes xmi:type="bpmn:Lane" xmi:id="_acjL8e_0Ed-zNqI9Y599ag" iD="_acjL8O_0Ed-zNqI9Y599ag" name="Equipe
Service Qualité" activities="_VdM9EO_0Ed-zNqI9Y599ag_jSob0e_0Ed-zNqI9Y599ag_3Bkroe_0Ed-zNqI9Y599ag
_erHose_1Ed-zNqI9Y599ag_qgbCwe_1Ed-zNqI9Y599ag">
      <vertices xmi:type="bpmn:Activity" xmi:id="_VdM9EO_0Ed-zNqI9Y599ag" iD="_VdMWAO_0Ed-zNqI9Y599ag"
outgoingEdges="_3BqLMe_0Ed-zNqI9Y599ag" incomingEdges="_j5NToO_0Ed-zNqI9Y599ag" name="Réaliser
prélèvement" lanes="_acjL8e_0Ed-zNqI9Y599ag" activityType="Task">
        <m_BPA x="118" y="18" height="61" width="107" xmi_id="_VdM9EO_0Ed-zNqI9Y599ag" />
      </vertices>
      <vertices xmi:type="bpmn:Activity" xmi:id="_jSob0e_0Ed-zNqI9Y599ag" iD="_jSob0O_0Ed-
zNqI9Y599ag" outgoingEdges="_j5NToO_0Ed-zNqI9Y599ag" lanes="_acjL8e_0Ed-zNqI9Y599ag"
activityType="EventStartEmpty">
        <m_BPA x="38" y="32" height="null" width="null" xmi_id="_jSob0e_0Ed-zNqI9Y599ag" />
      </vertices>
      <vertices xmi:type="bpmn:Activity" xmi:id="_3Bkroe_0Ed-zNqI9Y599ag" iD="_3BkroO_0Ed-zNqI9Y599ag"
outgoingEdges="_erW5Qe_1Ed-zNqI9Y599ag" incomingEdges="_3BqLMe_0Ed-zNqI9Y599ag" name="Contrôler
prélèvement" lanes="_acjL8e_0Ed-zNqI9Y599ag" activityType="Task">
        <m_BPA x="274" y="18" height="null" width="106" xmi_id="_3Bkroe_0Ed-zNqI9Y599ag" />
      </vertices>
      <vertices xmi:type="bpmn:Activity" xmi:id="_erHose_1Ed-zNqI9Y599ag" iD="_erHosO_1Ed-zNqI9Y599ag"
outgoingEdges="_qggiUe_1Ed-zNqI9Y599ag_4Hv0Ee_1Ed-zNqI9Y599ag" incomingEdges="_erW5Qe_1Ed-
zNqI9Y599ag" name="Etat du lot ?" lanes="_acjL8e_0Ed-zNqI9Y599ag_a5b8Qe_0Ed-zNqI9Y599ag"
activityType="GatewayDataBasedExclusive">
        <m_BPA x="420" y="22" height="null" width="75" xmi_id="_erHose_1Ed-zNqI9Y599ag" />
      </vertices>
      <vertices xmi:type="bpmn:Activity" xmi:id="_qgbCwe_1Ed-zNqI9Y599ag" iD="_qgbCwO_1Ed-zNqI9Y599ag"
outgoingEdges="_n9IUse_3Ed-zNqI9Y599ag" incomingEdges="_qggiUe_1Ed-zNqI9Y599ag" name="Passer en suivi de
lots bloqués" lanes="_acjL8e_0Ed-zNqI9Y599ag" activityType="Task">
        <m_BPA x="538" y="18" height="null" width="114" xmi_id="_qgbCwe_1Ed-zNqI9Y599ag" />
      </vertices>
      <m_BPA x="null" y="5" height="132" width="20" xmi_id="_acjL8e_0Ed-zNqI9Y599ag" />
    </lanes>
    <lanes xmi:type="bpmn:Lane" xmi:id="_a5b8Qe_0Ed-zNqI9Y599ag" iD="_a5b8QO_0Ed-zNqI9Y599ag"
name="Equipe Conditionnement" activities="_4Hq7ke_1Ed-zNqI9Y599ag_B0jQke_2Ed-zNqI9Y599ag_GPvpge_2Ed-
zNqI9Y599ag_MW3Ywe_2Ed-zNqI9Y599ag_PX79Ee_2Ed-zNqI9Y599ag_n5SL8e_2Ed-zNqI9Y599ag_rD5Gwe_2Ed-
zNqI9Y599ag_tjtYle_2Ed-zNqI9Y599ag_8TN5ge_2Ed-zNqI9Y599ag_S9BEe_2Ed-zNqI9Y599ag_DqK48e_3Ed-
zNqI9Y599ag_MQZu8e_3Ed-zNqI9Y599ag_Rh7S8e_3Ed-zNqI9Y599ag_Z_EYO_3Ed-zNqI9Y599ag_iMRJUe_3Ed-
zNqI9Y599ag_mLc7se_3Ed-zNqI9Y599ag_qKVMEo_3Ed-zNqI9Y599ag_erHose_1Ed-zNqI9Y599ag">
      <vertices xmi:type="bpmn:Activity" xmi:id="_4Hq7ke_1Ed-zNqI9Y599ag" iD="_4Hq7kO_1Ed-zNqI9Y599ag"
outgoingEdges="_B0owle_2Ed-zNqI9Y599ag" incomingEdges="_4Hv0Ee_1Ed-zNqI9Y599ag" name="Réceptionner le
lot" lanes="_a5b8Qe_0Ed-zNqI9Y599ag" activityType="Task">
        <m_BPA x="489" y="258" height="null" width="null" xmi_id="_4Hq7ke_1Ed-zNqI9Y599ag" />
      </vertices>

```

Figure 92. Extrait du fichier *m<sub>BPA</sub>*

### 15.4.2 Extrait du fichier mBPI

```

<?xml version="1.0" encoding="UTF-8"?>
<openerp:Data xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:openerp="http://openerp/1.0" xsi:schemaLocation="http://openerp/1.0
../metamodels/BPI/openERP.ecore" name="ReintegProcess" version="v1.0" author="ULMER Jean-Stéphane">
  <activities id="_jSob0O_0Ed-zNqI9Y599ag" wkf_id="wkf0" name="Start" flow_start="true"/>
  <activities id="_VdMWA0_0Ed-zNqI9Y599ag" wkf_id="wkf0" name="Réaliser Prélèvement" kind="function"
action="true"/>
  <activities id="_3BkroO_0Ed-zNqI9Y599ag" wkf_id="wkf0" name="Contrôler Prélèvement" kind="function"/>
  <activities id="_erHosO_1Ed-zNqI9Y599ag" wkf_id="wkf0" name="Etat du lot?" split_mode="XOR"/>
  <activities id="_qgbCwO_1Ed-zNqI9Y599ag" wkf_id="wkf0" name="Passer en suivi de lots bloqués"/>
  <activities id="_4Hq7kO_1Ed-zNqI9Y599ag" wkf_id="wkf0" name="Réceptionner le lot" kind="function"/>
  <activities id="_B0jQkO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Type?" split_mode="XOR"/>
  <activities id="_GPvpgO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Traiter produits semi-finis" kind="function"/>
  <activities id="_MW3YwO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Traiter en interne" kind="function"/>
  <activities id="_8TN5gO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="default" join_mode="XOR"/>
  <activities id="_qKUIAO_3Ed-zNqI9Y599ag" wkf_id="wkf0" name="End" flow_stop="true"/>
  <activities id="_Z_dUO_3Ed-zNqI9Y599ag" wkf_id="wkf0" name="Traiter produits testeurs" kind="function"/>
  <activities id="_iMRJUO_3Ed-zNqI9Y599ag" wkf_id="wkf0" name="default" join_mode="XOR"/>
  <activities id="_S9BEO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Mettre en conformité" kind="function"/>
  <activities id="_mLc7sO_3Ed-zNqI9Y599ag" wkf_id="wkf0" name="Disponibilité palette?" split_mode="XOR"/>
  <activities id="_Rh7S8O_3Ed-zNqI9Y599ag" wkf_id="wkf0" name="Charger palette" kind="function"/>
  <activities id="_MQZu8O_3Ed-zNqI9Y599ag" wkf_id="wkf0" name="Attendre palette disponible"/>
  <activities id="_PX79EO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Traiter produits ventes" kind="function"/>
  <activities id="_n5SL8O_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Format sur ligne?" split_mode="XOR"/>
  <activities id="_rD5GwO_2Ed-zNqI9Y599ag" wkf_id="wkf0" name="Recellophaner produits" kind="function"/>
  <activities wkf_id="wkf0" name="Attendre ligne disponible" kind="function"/>
  <transitions id="_j5MskO_0Ed-zNqI9Y599ag" act_from="_jSob0O_0Ed-zNqI9Y599ag" act_to="_VdMWA0_0Ed-
zNqI9Y599ag" role_id="&quot;Equipe Service Qualité&quot;" condition="true"/>
  <transitions id="_3BqLMO_0Ed-zNqI9Y599ag" act_from="_VdMWA0_0Ed-zNqI9Y599ag" act_to="_3BkroO_0Ed-
zNqI9Y599ag" role_id="&quot;Equipe Service Qualité&quot;" condition=""/>
  <transitions id="_erW5QO_1Ed-zNqI9Y599ag" act_from="_3BkroO_0Ed-zNqI9Y599ag" act_to="_erHosO_1Ed-
zNqI9Y599ag" role_id="&quot;Equipe Service Qualité&quot;" />
  <transitions id="_qggjUO_1Ed-zNqI9Y599ag" act_from="_erHosO_1Ed-zNqI9Y599ag" act_to="_qgbCwO_1Ed-
zNqI9Y599ag" role_id="&quot;Equipe Service Qualité&quot;" condition="(lot bloqué) == true"/>
  <transitions id="_4Hv0EO_1Ed-zNqI9Y599ag" act_from="_qgbCwO_1Ed-zNqI9Y599ag" act_to="_8TN5gO_2Ed-
zNqI9Y599ag" role_id="&quot;Equipe Service Qualité&quot;" />
  <transitions id="_B0owJO_2Ed-zNqI9Y599ag" act_from="_erHosO_1Ed-zNqI9Y599ag" act_to="_4Hq7kO_1Ed-
zNqI9Y599ag" role_id="&quot;Equipe Service Qualité&quot;" condition="(lot bloqué) != true"/>
  <transitions id="_GP1JEO_2Ed-zNqI9Y599ag" act_from="_B0jQkO_2Ed-zNqI9Y599ag" act_to="_GPvpgO_2Ed-
zNqI9Y599ag" role_id="&quot;Equipe Conditionnement&quot;" condition="(default)"/>
  <transitions id="_MW9fYO_2Ed-zNqI9Y599ag" act_from="_GPvpgO_2Ed-zNqI9Y599ag" act_to="_MW3YwO_2Ed-
zNqI9Y599ag" role_id="&quot;Equipe Conditionnement&quot;" />
  <transitions id="_PYA1kO_2Ed-zNqI9Y599ag" act_from="_MW3YwO_2Ed-zNqI9Y599ag" act_to="_8TN5gO_2Ed-
zNqI9Y599ag" role_id="&quot;Equipe Conditionnement&quot;" />
  <transitions id="_n5XrgO_2Ed-zNqI9Y599ag" act_from="_B0jQkO_2Ed-zNqI9Y599ag" act_to="_Z_dUO_3Ed-
zNqI9Y599ag" role_id="&quot;Equipe Conditionnement&quot;" />
  <transitions id="_rD-mUO_2Ed-zNqI9Y599ag" act_from="_Z_dUO_3Ed-zNqI9Y599ag" act_to="_8TN5gO_2Ed-
zNqI9Y599ag" role_id="&quot;Equipe Conditionnement&quot;" condition=""/>
  <transitions id="_tjy3sO_2Ed-zNqI9Y599ag" act_from="_8TN5gO_2Ed-zNqI9Y599ag" act_to="_S9BEO_2Ed-
zNqI9Y599ag" role_id="&quot;Equipe Conditionnement&quot;" />
  <transitions id="_2sloMO_2Ed-zNqI9Y599ag" act_from="_S9BEO_2Ed-zNqI9Y599ag" act_to="_mLc7sO_3Ed-

```

Figure 93.Extrait du fichier m<sub>BPI</sub>

### 15.4.3 Fichiers constituant le module OpenERP

Les différents fichiers composant le module OpenERP traité dans la partie D. Mise en œuvre sont présentés dans cette section. A titre de rappel, leurs rôles détaillés dans le Tableau 25 ci-dessous :

Tableau 25. Noms et rôles des fichiers du module OpenERP

Nom du fichier	Rôle
<code>__init__.py</code>	Fichier de démarrage indiquant les <i>imports</i> du module
<code>__terp__.py</code>	Fichier indiquant la nomenclature du module
<code>ReintegProcess.py</code>	Description fonctionnelle (classes et fonctions)
<code>ReintegProcess_view.xml</code>	Description de l'interface
<code>ReintegProcess_workflow.xml</code>	Description du séquençement d'activités

```
import ReintegProcess
```

Figure 94. Fichier `__init__.py`

```
{
  'name' : 'ReintegProcess',
  'version' : 'v1.0',
  'author' : 'ULMER Jean-Stéphane',
  'depends' : ['base'],
  'update_xml' : [
    'ReintegProcess_workflow.xml', 'ReintegProcess_view.xml' ],
  'installable': True,
  'active': False,
}
```

Figure 95. Fichier `__terp__.py`

```

from osv import fields, osv
class ReintegProcess_ReintegProcess (osv.osv):
    _name = "ReintegProcess.ReintegProcess"
    _columns = {
        'user_id': fields.many2one('res.users', 'Utilisateur', required=True,
readonly=True),
        'description': fields.text('Description', required=True, help='Remarques ou
commentaires'),
        'state': fields.selection([('_jSob00_0Ed-
zNq19Y599ag', 'Start'), ('_VdMWA0_0Ed-zNq19Y599ag', 'Réaliser Prélèvement'), ('_3Bkro0_0Ed-
zNq19Y599ag', 'Contrôler Prélèvement'), ('_erHos0_1Ed-zNq19Y599ag', 'Etat du
lot?'), ('_qgbCw0_1Ed-zNq19Y599ag', 'Passer en suivi de lots bloqués'), ('_4Hq7k0_1Ed-
zNq19Y599ag', 'Réceptionner le lot'), ('_B0jQk0_2Ed-zNq19Y599ag', 'Type?'), ('_GPvpg0_2Ed-
zNq19Y599ag', 'Traiter produits semi-finis'), ('_MW3Yw0_2Ed-zNq19Y599ag', 'Traiter en
interne'), ('_8TN5g0_2Ed-zNq19Y599ag', 'default'), ('_qKULAO_3Ed-zNq19Y599ag', 'End'), ('_Z_-
dUO_3Ed-zNq19Y599ag', 'Traiter produits testeurs'), ('_iMRJUO_3Ed-
zNq19Y599ag', 'default'), ('_S9BEO_2Ed-zNq19Y599ag', 'Mettre en conformité'), ('_mLc7s0_3Ed-
zNq19Y599ag', 'Disponibilité palette?'), ('_Rh7S80_3Ed-zNq19Y599ag', 'Charger
palette'), ('_MQZu80_3Ed-zNq19Y599ag', 'Attendre palette disponible'), ('_PX79EO_2Ed-
zNq19Y599ag', 'Traiter produits ventes'), ('_n5SL80_2Ed-zNq19Y599ag', 'Format sur
ligne?'), ('_rD5Gw0_2Ed-zNq19Y599ag', 'Recellophaner produits'), ('_null', 'Attendre ligne
disponible'),], 'Etat du module', readonly=True),
    }
    _defaults = {
        'user_id': lambda self, cr, uid, context: uid,
        'state': lambda *a: ('_jSob00_0Ed-zNq19Y599ag', 'Start'),
    }

def ReintegProcess__jSob00_0Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_jSob00_0Ed-zNq19Y599ag' })
    return True
def ReintegProcess__VdMWA0_0Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_VdMWA0_0Ed-zNq19Y599ag' })
    return True
def ReintegProcess__3Bkro0_0Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_3Bkro0_0Ed-zNq19Y599ag' })
    return True
def ReintegProcess__erHos0_1Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_erHos0_1Ed-zNq19Y599ag' })
    return True
def ReintegProcess__qgbCw0_1Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_qgbCw0_1Ed-zNq19Y599ag' })
    return True
def ReintegProcess__4Hq7k0_1Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_4Hq7k0_1Ed-zNq19Y599ag' })
    return True
def ReintegProcess__B0jQk0_2Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_B0jQk0_2Ed-zNq19Y599ag' })
    return True
def ReintegProcess__GPvpg0_2Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_GPvpg0_2Ed-zNq19Y599ag' })
    return True
def ReintegProcess__MW3Yw0_2Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_MW3Yw0_2Ed-zNq19Y599ag' })
    return True
def ReintegProcess__8TN5g0_2Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_8TN5g0_2Ed-zNq19Y599ag' })
    return True
def ReintegProcess__qKULAO_3Ed-zNq19Y599ag(self, cr, uid, ids):
    self.write(cr, uid, ids, { 'state' : '_qKULAO_3Ed-zNq19Y599ag' })
    return True

```

Figure 96.Extrait du fichier ReintegProcess.py

```

<?xml version="1.0" encoding="UTF-8"?>
<openerp>
  <data>
    <record model="workflow" id="wkf0_ReintegProcess">
      <field name="name">ReintegProcess.wkf0</field>
      <field name="osv">ReintegProcess.ReintegProcess</field>
      <field name="on_create">True</field>
    </record>
    <record model="workflow.activity" id="_jSob00_0Ed-zNq19Y599ag">
      <field name="wkf_id" ref="wkf0_ReintegProcess" />
      <field name="name">Start</field>
      <field name="flow_start">>true</field>
    </record>
    <record model="workflow.activity" id="_VdMWA0_0Ed-zNq19Y599ag">
      <field name="wkf_id" ref="wkf0_ReintegProcess" />
      <field name="name">Réaliser Prélèvement</field>
      <field name="kind">function</field>
      <field name="action">>true</field>
    </record>
    <record model="workflow.activity" id="_3Bkro0_0Ed-zNq19Y599ag">
      <field name="wkf_id" ref="wkf0_ReintegProcess" />
      <field name="name">Contrôler Prélèvement</field>
      <field name="kind">function</field>
    </record>
    <record model="workflow.activity" id="_erHos0_1Ed-zNq19Y599ag">
      <field name="wkf_id" ref="wkf0_ReintegProcess" />
      <field name="name">Etat du lot?</field>
      <field name="split_mode">XOR</field>
    </record>
    <record model="workflow.activity" id="_qgbCw0_1Ed-zNq19Y599ag">
      <field name="wkf_id" ref="wkf0_ReintegProcess" />
      <field name="name">Passer en suivi de lots bloqués</field>
    </record>
    <record model="workflow.activity" id="_4Hq7k0_1Ed-zNq19Y599ag">
      <field name="wkf_id" ref="wkf0_ReintegProcess" />
      <field name="name">Réceptionner le lot</field>
      <field name="kind">function</field>
    </record>
  </data>
</openerp>

```

Figure 97. Extrait du fichier ReintegProcess\_workflow.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<openerp>
  <data>
    <menuitem name="Modules SCALP" id="menu_scalp" icon="STOCK_PREFERENCES" />
    <record model="ir.ui.view" id="ReintegProcess_form_view">
      <field name="name">ReintegProcess.form</field>
      <field name="model">ReintegProcess.ReintegProcess</field>
      <field name="type">form</field>
      <field name="arch" type="xml">
        <form string="ReintegProcess">
          <field name="user_id" select="1" />
          <field name="description" colspan="4" select="2" widget="text_wiki" />
          <field name="state" select="2" />
          <group colspan="2" col="20">
            <button name="_VdMWAO_0Ed-zNql9Y599ag" string="réaliser prélèvement"
states="Start" />
            <button name="_3BkroO_0Ed-zNql9Y599ag" string="contrôler prélèvement"
states="Réaliser Prélèvement" />
            <button name="_erHosO_1Ed-zNql9Y599ag" string="etat du lot?"
states="Contrôler Prélèvement" />
            <button name="_qgbCwO_1Ed-zNql9Y599ag" string="passer en suivi de lots
bloqués" states="Etat du lot?" />
            <button name="_4Hq7kO_1Ed-zNql9Y599ag" string="réceptionner le lot"
states="Passer en suivi de lots bloqués" />
            <button name="_B0jQkO_2Ed-zNql9Y599ag" string="type?" states="Réceptionner le
lot" />
            <button name="_GPvpgO_2Ed-zNql9Y599ag" string="traiter produits semi-finis"
states="Type?" />
            <button name="_MW3YwO_2Ed-zNql9Y599ag" string="traiter en interne"
states="Traiter produits semi-finis" />
            <button name="_8TN5gO_2Ed-zNql9Y599ag" string="default" states="Traiter en
interne" />
            <button name="_qKULAO_3Ed-zNql9Y599ag" string="end" states="default" />
            <button name="_Z_-dUO_3Ed-zNql9Y599ag" string="traiter produits testeurs"
states="End" />
            <button name="_iMRJUO_3Ed-zNql9Y599ag" string="default" states="Traiter
produits testeurs" />
            <button name="__S9BEO_2Ed-zNql9Y599ag" string="mettre en conformité"
states="default" />
            <button name="_mLc7sO_3Ed-zNql9Y599ag" string="disponibilité palette?"
states="Mettre en conformité" />
            <button name="_Rh7S8O_3Ed-zNql9Y599ag" string="charger palette"
states="Disponibilité palette?" />
            <button name="_MQZu8O_3Ed-zNql9Y599ag" string="attendre palette disponible"
states="Charger palette" />
            <button name="_PX79EO_2Ed-zNql9Y599ag" string="traiter produits ventes"
states="Attendre palette disponible" />
            <button name="_n5SL8O_2Ed-zNql9Y599ag" string="format sur ligne?"
states="Traiter produits ventes" />
            <button name="_rD5GwO_2Ed-zNql9Y599ag" string="recellophaner produits"
states="Format sur ligne?" />
            <button name="default" string="attendre ligne disponible"
states="Recellophaner produits" />
          </group>
        </form>
      </field>
    </record>
    <record model="ir.ui.view" id="ReintegProcess_tree_view">
      <field name="name">ReintegProcess.tree</field>
      <field name="model">ReintegProcess.ReintegProcess</field>

```

Figure 98. Extrait du fichier ReintegProcess\_view.xml

15.4.4 Diagramme Intalio obtenu en fin de seconde phase

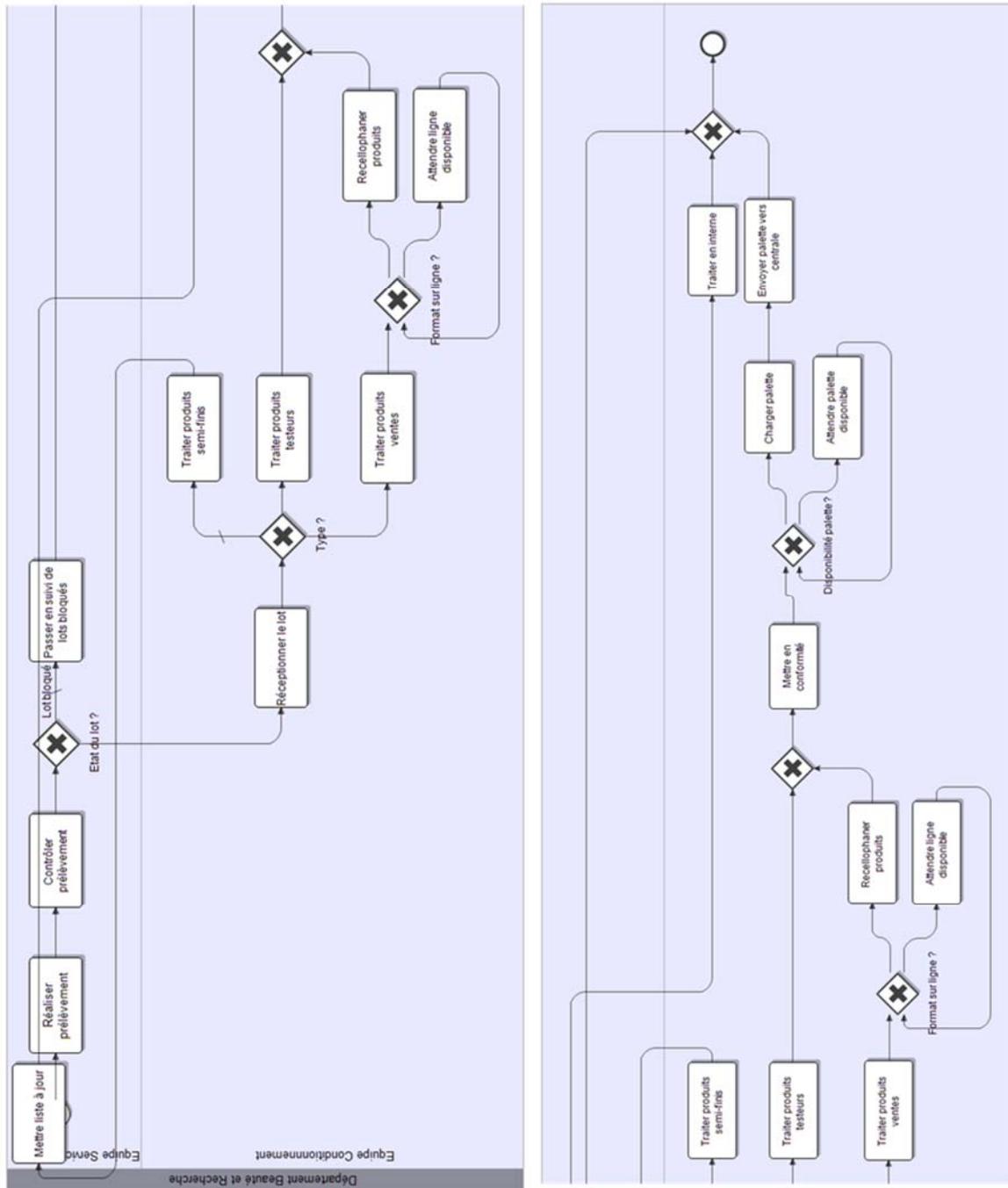


Figure 99. Diagramme Intalio obtenue en fin de seconde phase

15.4.5 Diagramme Intalio modifié au début de la troisième phase

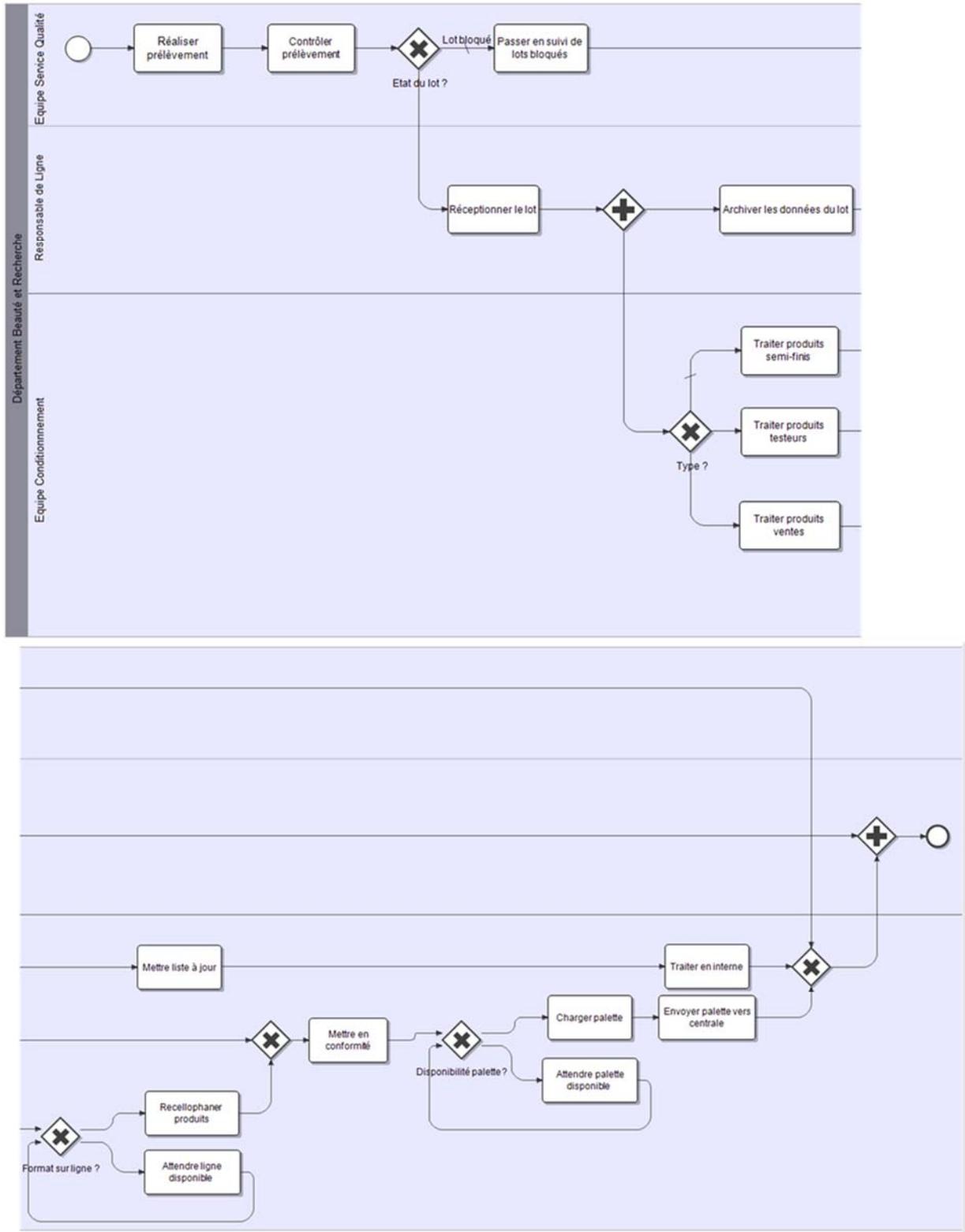


Figure 100. Diagramme Intalio modifié au début de la troisième phase



## 15.5 SOMMAIRE DETAILLE

<b>A. PREAMBULE</b> .....	<b>15</b>
<b>1. INTRODUCTION</b> .....	<b>17</b>
1.1 CADRE DES TRAVAUX DE RECHERCHE .....	18
1.2 PRESENTATION DU MANUSCRIT .....	19
1.2.1 Démarche proposée.....	19
1.2.2 Typologie adoptée .....	20
<b>2. CONTEXTE ET PROBLEMATIQUE</b> .....	<b>23</b>
2.1 CONTEXTE .....	24
2.1.1 Naissance et importance des stratégies SITI et métier.....	24
2.1.2 Besoin d'un alignement entre stratégie métier et stratégie SITI .....	24
2.1.3 Alignement stratégique, alignement opérationnel.....	25
2.1.4 Emergence d'un nouveau domaine de l'ingénierie des SI.....	26
2.2 PROBLEMATIQUE .....	27
2.2.1 Alignement opérationnel, intégration fonctionnelle, cohérence, .....	27
2.2.2 Cycles de vie d'applications et de technologies asynchrones .....	27
2.2.3 De la difficulté à maintenir l'alignement.....	28
2.3 OBJECTIFS DE RECHERCHE .....	29
2.3.1 Contributions.....	29
2.3.2 Résultats principaux.....	29
<b>B. CADRE METHODOLOGIQUE</b> .....	<b>31</b>
<b>3. DE L'ENTREPRISE AU PROCESSUS</b> .....	<b>33</b>
3.1 ENTREPRISE .....	34
3.1.1 Définition.....	34
3.1.2 Entreprise, un système de systèmes.....	35
3.2 SYSTEME D'INFORMATION.....	36
3.3 MODELISATION D'ENTREPRISE .....	38
3.4 CADRE DE MODELISATION D'ENTREPRISE .....	39
3.4.1 Cycle de vie.....	40
3.4.2 Structure et comportement.....	41
3.4.3 Degré de particularisation.....	41
3.5 CONCLUSION.....	42
<b>4. NOTIONS AUTOUR DU TERME « PROCESSUS »</b> .....	<b>43</b>

4.1	PROCESSUS.....	44
4.2	PROCESSUS MÉTIER.....	45
4.2.1	Processus métier et SI.....	47
4.2.2	Processus métier, processus fonctionnel, procédure.....	48
4.3	CYCLE DE VIE ET ACTEURS.....	49
4.3.1	Cycle de vie.....	49
4.3.2	Acteurs.....	49
4.4	MODELISATION PAR LES PROCESSUS : IMPORTANCE DE LA VUE FONCTIONNELLE.....	50
4.5	CONCLUSION.....	51
<b>5.</b>	<b>INGENIERIE ET ARCHITECTURE DIRIGÉES PAR LES MODELES.....</b>	<b>53</b>
5.1	INGENIERIE DIRIGÉE PAR LES MODELES.....	54
5.1.1	Système réel, modèle, métamodèle.....	54
5.1.2	Processus de tissage de conception.....	55
5.2	ARCHITECTURE DIRIGÉE PAR LES MODELES.....	55
5.2.1	Rôles des CIM, PIM et PSM.....	56
5.2.2	MOF, UML, XMI.....	57
5.2.3	Limite du MDA.....	58
5.3	CONCLUSION.....	59
<b>6.</b>	<b>INGENIERIE DES PROCESSUS METIER.....</b>	<b>61</b>
6.1	DE LA GESTION DU WORKFLOW A LA GESTION DES PROCESSUS METIER.....	62
6.2	LE CYCLE DE VIE DU PROCESSUS SELON L'APPROCHE BPM.....	64
6.2.1	Business Process Analysis, BPA.....	65
6.2.2	Business Process Implementation, BPI.....	65
6.2.3	Business Activity Monitoring, BAM.....	66
6.3	Du MDA AU BPM.....	67
6.4	SUITES BPM.....	68
6.4.1	Une évolution nécessaire.....	68
6.4.2	Définition.....	68
6.4.3	Structure.....	70
6.4.4	Classification.....	70
6.4.5	Conclusion sur les BPMS.....	71
6.5	LES LIMITES DU BPM.....	72
6.6	CONCLUSION.....	72
<b>C.</b>	<b>DEFINITION DE L'APPROCHE.....</b>	<b>75</b>
<b>7.</b>	<b>DE LA NECESSITE DE L'APPROCHE.....</b>	<b>77</b>

7.1	<i>VERS UN ALIGNEMENT OPERATIONNEL</i> .....	78
7.1.1	<i>Synchronisation</i> .....	79
7.1.2	<i>Equivalence sémantique</i> .....	79
7.1.3	<i>Cohérence intermodèle</i> .....	80
7.2	<i>HETEROGENEITE ET DIFFERENTES ABSTRACTIONS</i> .....	81
7.2.1	<i>Modèle d'analyse</i> .....	81
7.2.2	<i>Modèle d'implémentation</i> .....	81
7.2.3	<i>Deux domaines, deux visions : prémisses de l'écart métier-SITI</i> .....	82
7.3	<i>CONCEPTS ET APPROCHES POUR UNE GESTION AGILE</i> .....	82
7.3.1	<i>Utilisation de règles métier</i> .....	83
7.3.2	<i>Facilité l'évolution des plateformes</i> .....	83
7.3.3	<i>Transformation entre modèles : la nécessité du métamodèle</i> .....	85
7.4	<i>CONCLUSION</i> .....	86
<b>8.</b>	<b><i>CARACTERISATION</i></b> .....	<b>89</b>
8.1	<i>VERS UNE APPROCHE CENTREE PIVOT</i> .....	90
8.2	<i>VUES</i> .....	91
8.3	<i>GENERICITE</i> .....	92
8.4	<i>ACTEURS</i> .....	92
8.5	<i>DONNEES</i> .....	92
8.6	<i>CONCLUSION</i> .....	96
<b>9.</b>	<b><i>CONCEPTION</i></b> .....	<b>97</b>
9.1	<i>CONFORMITE ENTRE MODELE ET METAMODELE</i> .....	98
9.1.1	<i>Modèle et « ReprésentationDe (<math>\mu</math>) »</i> .....	98
9.1.2	<i>Métamodèle et « EstConformeA (<math>\chi</math>) »</i> .....	98
9.2	<i>DE LA TRANSFORMATION DE MODELES BIDIRECTIONNELLES A LA NOTION DE PIVOT</i> .....	100
9.2.1	<i>Relation bidirectionnelle</i> .....	100
9.2.2	<i>Fonctions de conformité constructive</i> .....	101
9.3	<i>TRANSFORMATIONS HORIZONTALES, TRANSFORMATIONS VERTICALES ET INTEROPERABILITE</i> .....	103
9.4	<i>METAMODELE ET METAMODELES PIVOT</i> .....	104
9.5	<i>DEFINITION DU METAMODELE PIVOT</i> .....	107
9.5.1	<i>Éléments de modélisation des processus métier</i> .....	107
9.5.2	<i>Famille d'éléments : Special</i> .....	111
9.5.3	<i>Groupe Node</i> .....	112
9.5.4	<i>Groupe Edge</i> .....	113
9.5.5	<i>Groupe Swimlane</i> .....	114
9.5.6	<i>Groupe Artefact</i> .....	114
9.6	<i>CARACTERISATION DES LIENS SEMANTIQUES</i> .....	115

9.6.1	Typologie des liens sémantiques.....	115
9.6.2	Exemple d'équivalence sémantique.....	115
9.7	FORMATION DES METAMODELES BPA ET BPI.....	116
9.8	CONCLUSION.....	117

<b>D. MISE EN ŒUVRE.....</b>	<b>119</b>
------------------------------	------------

<b>10. PLATEFORME SOLUTION POUR UNE COHERENCE ET UN ALIGNEMENT DES PROCESSUS - SCALP .....</b>	<b>121</b>
10.1 ARCHITECTURE GENERALE DE LA PLATEFORME SCALP .....	122
10.2 ENVIRONNEMENT PIVOT .....	122
10.2.1 Eclipse Modeling Framework.....	122
10.2.2 Kermeta.....	124
10.3 ENVIRONNEMENT DE MODELISATION.....	125
10.4 ENVIRONNEMENT D'IMPLEMENTATION .....	126
10.4.1 SI et ERP .....	126
10.4.2 Côté technique.....	126
10.5 IMPLEMENTATION DES MAPPINGS ET TRANSFORMATIONS .....	129
10.5.1 Métamodèles BPA et BPI.....	129
10.5.2 Pattern Visiteur.....	131
10.5.3 Mappings et transformations.....	134
10.6 CONCLUSION.....	134
<b>11. DEMONSTRATION .....</b>	<b>137</b>
11.1 PRESENTATION DU PROCESSUS ETUDIE .....	139
11.2 SCENARIO DE VALIDATION.....	139
11.3 PREMIERE PHASE : DU DIAGRAMME DE PROCESSUS AU MODULE ERP.....	143
11.3.1 Du diagramme de processus Intalio vers un modèle d'analyse $m_{BPA}$ .....	143
11.3.2 Du modèle d'analyse vers le modèle pivot .....	145
11.3.3 Du modèle pivot vers le modèle d'implémentation.....	147
11.3.4 Du modèle d'implémentation vers le module OpenERP .....	148
11.4 DEUXIEME PHASE : DU MODULE ERP AU DIAGRAMME DE PROCESSUS .....	149
11.4.1 Modifications apportées .....	149
11.4.2 Du module OpenERP vers le modèle d'implémentation.....	150
11.4.3 Du modèle d'implémentation au diagramme de processus .....	151
11.5 TROISIEME PHASE : DU DIAGRAMME DE PROCESSUS AU MODULE ERP .....	154
11.6 CONCLUSION.....	155
<b>12. INGENIERIE DES PROCESSUS AU SERVICE DE L'INGENIERIE DES PROCEDES.....</b>	<b>157</b>
12.1 NOTIONS AUTOUR DE L'INGENIERIE DES PROCEDES .....	158

12.2	VERS UNE APPROCHE PROCESSUS-PROCEDE .....	159
12.3	CONCLUSION.....	163

<b>E. EPILOGUE.....</b>	<b>165</b>
-------------------------	------------

<b>13. CONCLUSION ET PERSPECTIVES.....</b>	<b>167</b>
13.1 CONCLUSION.....	167
13.2 BILAN.....	168
13.2.1 Cadre méthodologique.....	168
13.2.2 Définition de l'approche.....	168
13.2.3 Mise en œuvre.....	169
13.3 SYNTHÈSE DES CONTRIBUTIONS .....	170
13.4 LIMITES ET PERSPECTIVES.....	171
13.4.1 Utilisation et utilité de la plateforme SCALP.....	171
13.4.2 Génération de code et orchestration de services.....	172
13.4.3 Alignement stratégique, alignement opérationnel.....	172
13.4.4 Abstraction et interopérabilité .....	172
13.4.5 Evolution de la plateforme SCALP.....	173
13.4.6 Extension de l'approche SCALP au domaine Process Systems Engineering .....	173
<b>14. BIBLIOGRAPHIE .....</b>	<b>175</b>
<b>15. ANNEXES .....</b>	<b>185</b>
15.1 GLOSSAIRE .....	186
15.2 TRADUCTION DES TERMES ANGLOPHONES.....	190
15.3 ACRONYMES .....	192
15.4 DEMONSTRATION : EXTRAITS DE FICHIERS.....	194
15.4.1 Extrait du fichier m <sub>BPA</sub> .....	194
15.4.2 Extrait du fichier mBPI.....	195
15.4.3 Fichiers constituant le module OpenERP.....	196
15.4.4 Diagramme Intalio obtenu en fin de seconde phase.....	200
15.4.5 Diagramme Intalio modifié au début de la troisième phase.....	201
15.5 SOMMAIRE DÉTAILLÉ.....	203