



HAL
open science

Proposition d'une méthodologie de modélisation et de réorganisation du circuit du médicament dans les pharmacies hospitalières

Johan Royer

► **To cite this version:**

Johan Royer. Proposition d'une méthodologie de modélisation et de réorganisation du circuit du médicament dans les pharmacies hospitalières. Autre [cs.OH]. Université Blaise Pascal - Clermont-Ferrand II, 2014. Français. NNT : 2014CLF22533 . tel-01163428

HAL Id: tel-01163428

<https://theses.hal.science/tel-01163428v1>

Submitted on 12 Jun 2015

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

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

N° d'ordre : D.U. : 2533
EDSPIC : 680

Université Blaise Pascal – Clermont II

École Doctorale Sciences Pour L'Ingénieur de Clermont-Ferrand

THÈSE

présentée pour obtenir le grade de Docteur d'Université,
spécialité « Informatique »

par

Johan Royer

PROPOSITION D'UNE METHODOLOGIE DE MODELISATION ET DE REORGANISATION DU CIRCUIT DU MEDICAMENT DANS LES PHARMACIES HOSPITALIERES

Thèse soutenue le 17/12/2014 devant le jury composé de

M.	VINCENT CHAPURLAT	Professeur des Universités, Ecole des Mines d'Alès	(Examineur)
M ^{me}	MICHELLE CHABROL	Maître de Conférences, ISIMA	(Co-encadrant)
M.	JOËL COLLOC	Professeur des Universités, Université du HAVRE	(Rapporteur)
M.	MICHEL GOURGAND	Professeur des Universités, ISIMA	(Examineur)
M.	ALAIN GUINET	Professeur des Universités, INSA de LYON	(Rapporteur)
M.	JEAN-LUC PARIS	Professeur des Universités, IFMA	(Directeur)
M.	PHILIPPE RAMADOUR	Maître de Conférences, Université Paul Cézanne Aix-Marseille	(Co-encadrant)
M	VINCENT PINON	Pharmacien, CHU de Clermont-Ferrand	(Examineur)

Réalisé en collaboration entre le Laboratoire d'Informatique, de Modélisation et d'Optimisation des systèmes (LIMOS) et le Centre Hospitalier Universitaire (CHU) de Clermont-Ferrand

REMERCIEMENTS

Partie la plus courte mais pas nécessairement la plus simple de la thèse pour autant. De nombreuses personnes sont à remercier pour leur participation, leur soutien, leur aide et que sais-je encore, c'est pourquoi je remercie par avance tous ceux qui n'apparaîtraient pas sur cette page.

Michel Gourgand, Bernard Aleksy et Jean Chopineau pour m'avoir permis de réaliser ce travail de longue haleine et m'avoir ouvert les portes d'un milieu qui m'était jusqu'alors inconnu.

Bernard, et oui encore, Michèle Chabrol et Jean-Luc Paris, pour m'avoir encadré, soutenu durant ces trois années et surtout pour leur bonne humeur toujours au rendez-vous et leur disponibilité quel que soit le moment.

Michèle et Jean-Luc merci aussi pour vos relectures assidues de mes articles, ce n'était pas gagné, et surtout pour la relecture de cette thèse qui vous aura fortement mis à contribution (mentale et physique !).

Merci aussi à Vincent Pinon pour avoir toujours été disponible lorsque j'avais des questions et pour son aide plus que précieuse dans mon travail.

Merci à JFD pour ces trois années passées dans le même bureau, pour son aide et surtout pour les nombreuses discussions (de travail bien entendu) que nous avons eu.

Aux internes et tout particulièrement Bertrand, Lucie et Matthieu, merci de m'avoir accepté parmi vous, j'ai passé des soirées mémorables à vos côtés (et j'espère en passer encore).

A Béa, pour sa bonne humeur, sans qui cela ne se serait fait (elle comprendra).

A Jojo, Aline, Marie, Pascal, Soumia et Paulette, pour les nombreux gâteaux, pour leur soutien et pour la bonne ambiance dans laquelle j'ai pu travailler pendant ces trois ans.

Un mot aussi pour les rapporteurs, examinateurs et membres du jury de cette thèse, Alain Guinet et Joël Colloc qui ont accepté d'évaluer ce travail, qui j'espère les intéressera, à Vincent Chapurlat et Vincent Pinon pour avoir accepté de participer à mon jury, et enfin à Philippe Ramadour, présent malgré les difficultés.

A Thomas, j'arrive bientôt en Alsacie !!

Enfin merci à mes parents et ma sœur qui ont toujours tout fait pour que je puisse être là où je suis aujourd'hui.

Et pour terminer, merci à toi Salomé, pour ton soutien, ton aide, ta patience, ta relecture méticuleuse (c'est pas pertinent !!!) et pour tout le reste, qui comme tu le sais, ne compte que trop pour moi.

TABLE DES MATIÈRES

TABLE DES MATIERES	vi
LISTE DES FIGURES	xii
LISTE DES TABLEAUX	xviii
LISTE DES ILLUSTRATIONS	xxi
LISTE DES ACRONYMES	xxii
INTRODUCTION GENERALE	1
1 CONTEXTE ET PROBLEMATIQUES	5
1.1 INTRODUCTION	7
1.2 QU'EST CE QUE LE CIRCUIT DU MEDICAMENT ?	7
1.2.1 Définitions	7
1.2.2 Les étapes du CDM	9
1.2.3 Un cadre réglementaire strict	11
1.2.4 Les différents modes de délivrance existants	12
1.3 LE CDM : CONTRAINT ET SOUS PRESSION	18
1.3.1 Introduction	18
1.3.2 Les contraintes du CDM	19
1.3.3 Les problèmes constatés	22
1.3.4 Erreurs et Evènements Indésirables	23
1.3.5 Classification des EIM et étude ENEIS	31
1.4 COMMENT SECURISER LE CDM ?	34
1.4.1 Les recommandations ministérielles et les lois	35
1.4.2 Que faire à chaque étape du CDM d'après ces lois ?	37
1.4.3 Quelles voies de sécurisation ?	41
1.5 CONCLUSION	50
2 DE L'EXISTANT A SA COMPREHENSION	51
2.1 INTRODUCTION	53

2.2	COMMENT CARACTERISER LE CDM ?	53
2.2.1	Pourquoi parle-t-on de processus dans le soin ?	53
2.2.2	Modélisation et modèle !	54
2.2.3	Qu'est-ce qu'un processus ?	55
2.2.4	L'approche par processus en quelques mots	57
2.2.5	Modélisation et granularité des processus	57
2.2.6	Les différents niveaux des processus.	59
2.2.7	Les différentes classes de processus	60
2.3	QUELS LANGAGES POUR LA CARTOGRAPHIE ?	61
2.3.1	Les langages choisis	62
2.3.2	Evaluation des langages	65
2.4	METHODES & METHODOLOGIES.....	67
2.4.1	Les méthodologies de modélisation.....	68
2.4.2	Les méthodes de modélisation	71
2.4.3	Conclusion au niveau des méthodes	76
2.5	LA SIMULATION : POURQUOI ET COMMENT ?	76
2.5.1	Qu'est-ce que la simulation ?	76
2.5.2	Doit-elle être utilisée à chaque fois ?	77
2.5.3	Quels avantages et inconvénients ?	78
2.5.4	Les différentes classes de simulation	79
2.5.5	Les niveaux d'abstraction	80
2.6	LES CODES DE LA GENERICITE	82
2.6.1	Le spectre de la généricité.....	83
2.6.2	Générique versus Spécifique	84
2.6.3	La réutilisabilité	86
2.7	CONCLUSION	89
3	OU EN EST LA SIMULATION ?	91
3.1	INTRODUCTION	92
3.2	SITUATION DE LA SIMULATION HOSPITALIERE	92
3.2.1	Une évolution marquée.....	93
3.2.2	Quelles tendances dans la simulation ?	94
3.2.3	Qu'en est-il du CDM ?.....	100
3.3	CONCLUSION	103

4	ASCI & BPMN	105
4.1	INTRODUCTION	107
4.2	ANALYSE, SPECIFICATION, CONCEPTION ET IMPLEMENTATION	107
4.2.1	Etape 1 : Construction du modèle générique de connaissance	110
4.2.2	Etape 2 : Construction du modèle d'action pour un système	113
4.2.3	Etape 3 : Mise en œuvre du modèle d'action	114
4.2.4	Etape 4 : Action sur le système	114
4.2.5	Conclusion	114
4.3	BUSINESS PROCESS MODEL AND NOTATION	114
4.3.1	Le Business Process Management.....	115
4.3.2	BPMN, une histoire récente	115
4.3.3	Les éléments du langage	116
4.3.4	Les activités	117
4.3.5	Les évènements.....	122
4.3.6	Les Portes	125
4.3.7	« Pool » et « Ligne »	129
4.3.8	Les données.....	130
4.3.9	Les connecteurs.....	132
4.3.10	BPMN : son utilisation	133
4.4	CONCLUSION	136
5	ASCI POUR LES PHARMACIENS	139
5.1	INTRODUCTION	140
5.2	CONSTRUCTION DE NOTRE DEMARCHE.....	140
5.2.1	Les adaptations.....	140
5.2.2	Spécification des phases.....	142
5.2.3	Agencement des phases.....	143
5.2.4	Bilan outils & Comparaison ASCI/Démarche.....	144
5.3	CONCLUSION	148
6	CONCEPTION DU MODELE GENERIQUE DE CONNAISSANCE	149
6.1	INTRODUCTION	151
6.2	MODELE GENERIQUE BPMN DES PROCESSUS	151
6.2.1	Les acteurs et entités du CDM.....	153
6.2.2	Les activités	155

6.2.3	Les points clés du processus.....	157
6.2.4	L'impact des points clés sur les actions des acteurs	159
6.2.5	Dernier pas vers un modèle final	161
6.2.6	Le modèle final	163
6.2.7	Les simplifications et hypothèses faites	171
6.2.8	Proposition d'évolution pour BPMN	172
6.3	DIAGRAMME GENERIQUE DE CLASSES UML.....	176
6.3.1	Le diagramme SSPDL de [RODIER 10]	176
6.3.2	Le diagramme SSPDL de [HUET 11]	176
6.3.3	Notre diagramme des classes.....	180
6.4	CONCLUSION	182
7	BASE DE LA CONCEPTION ET DU DEVELOPPEMENT DU MODELE DE SIMULATION	185
7.1	INTRODUCTION	187
7.2	CRITERES DE PERFORMANCE ET CURSEURS	187
7.2.1	Evaluation de la performance	187
7.2.2	Leviers de simulation.....	189
7.3	SIMIO.....	190
7.3.1	Qu'est-ce que SIMIO ?.....	190
7.3.2	Les objets.....	197
7.3.3	Les processus dans SIMIO	209
7.3.4	Un exemple simple d'application de SIMIO	212
7.4	NOTRE APPROCHE POUR PASSER DE BPMN A SIMIO	213
7.4.1	Comment traduire une activité.....	214
7.4.2	Comment traduire les évènements.....	217
7.4.3	Comment traduire les portes décisionnelles.....	220
7.4.4	Comment traduire les acteurs.....	221
7.5	CONCLUSION	222
8	CONCEPTION DU MODELE DE SIMULATION	225
8.1	INTRODUCTION	227
8.2	LE MODELE DE SIMULATION DANS SA GLOBALITE	228
8.2.1	Vue globale.....	228
8.2.2	Les éléments du modèle.....	228
8.2.3	Les parcours possibles.....	229
8.3	LES ELEMENTS COMPOSANT LE MODELE	230

8.3.1	Les travailleurs.....	230
8.3.2	Les demandes.....	231
8.3.3	Le Service source.....	233
8.3.4	Les demandes urgentes et leur génération.....	237
8.3.5	La séparation des prescriptions d'une demande.....	238
8.3.6	La décomposition des prescriptions.....	241
8.3.7	La validation des prescriptions.....	245
8.3.8	La distribution des prescriptions.....	249
8.3.9	La préparation robotisée.....	255
8.3.10	Edition et contrôle des demandes.....	258
8.3.11	La décomposition des demandes.....	262
8.3.12	La préparation des demandes.....	265
8.3.13	Réunion, transport et envoi.....	266
8.3.14	Le réapprovisionnement des stocks.....	269
8.4	LES TABLES DE DONNEES ET LES ETATS.....	272
8.4.1	Les états.....	273
8.4.2	Pourquoi de telles tables ?.....	273
8.4.3	Les tables.....	274
8.5	Interface de saisie des données.....	280
8.6	CONCLUSION.....	283
9	EVALUATION DES OUTILS ET APPLICATION DE NOTRE DEMARCHE	285
9.1	INTRODUCTION.....	287
9.2	GENERICITE ET REUTILISABILITE DES OUTILS.....	287
9.2.1	La généricité.....	287
9.2.2	La réutilisabilité.....	292
9.3	CADRE GENERAL.....	294
9.4	SCENARIO 1 : IMPACT DE LA DJINM.....	296
9.4.1	Contexte.....	296
9.4.2	Utilisation des outils et de la démarche.....	296
9.4.3	Résultats de la simulation.....	300
9.4.4	Conclusion.....	302
9.5	SCENARIO 2 : IMPACT DE LA DJINA.....	302
9.5.1	Contexte.....	302
9.5.2	Utilisation des outils et de la démarche.....	303

9.5.3	Résultats de la simulation	305
9.5.4	Conclusion	309
9.6	SCENARIO 3 : COMMENT REPARTIR SA VALIDATION ?	309
9.6.1	Contexte	309
9.6.2	Utilisation des outils et de la démarche	310
9.6.3	Résultats de la simulation	313
9.6.4	Conclusion	318
9.7	CONCLUSION	318
CONCLUSION GENERALE		321
BIBLIOGRAPHIE		329
ANNEXES		337

LISTE DES FIGURES

Figure 1.1 : Chronologie des étapes du CDM.....	10
Figure 1.2 : Cheminement DgM	14
Figure 1.3 : Cheminement DGM.....	15
Figure 1.4 : Cheminement DNMP.....	16
Figure 1.5 : Cheminement DJIN.....	17
Figure 1.6 : Classification de Glouberman et Mintzberg.....	20
Figure 1.7 : Répartition des erreurs du GEM.....	25
Figure 1.8 : Les 4 étapes qui ont généré le plus d’erreurs pour GEM.....	25
Figure 1.9 : Répartition des erreurs dans l’étude MERVEIL	26
Figure 1.10 : Les 4 étapes générant le plus d’erreurs pour l’étude MERVEIL	26
Figure 2.1 : Modélisation du processus selon la définition ministérielle.....	56
Figure 2.2 : Modélisation de notre définition d’un processus	57
Figure 2.3 : Exemple d’un processus commun à deux organisations d’activités différentes	61
Figure 2.4 : Axe chronologique de l’apparition des langages	62
Figure 2.5 : Exemple de Réseaux de Pétri	62
Figure 2.6 : Exemple de diagramme EPC.....	63
Figure 2.7 : Exemple de diagramme UML	64
Figure 2.8 : Exemple de diagramme BPMN.....	64
Figure 2.9 : (a) Vue globale LAESH ; (b) Vue détaillée LAESH.....	65
Figure 2.10 : Couverture de modélisation de la méthodologie GRAI-GIM	69
Figure 2.11 : Vue synthétique basé sur GRAI-GIM.....	69
Figure 2.12 : La méthodologie ASCI.....	70
Figure 2.13 : Déroulement de la méthode GRAI	72
Figure 2.14 : Exemple de diagramme IDEF 3.....	73
Figure 2.15 : Exemple de diagramme IDEF 0.....	74
Figure 2.16 : Niveaux et étapes de la méthode MERISE	75
Figure 2.17 : Modèle abstrait de la méthode OSSAD.....	75
Figure 2.18 : Modèle d’opérations de la méthode OSSAD.....	76
Figure 2.19 : The Range of Modeling Options and the Building Blocks Used in Each Case	80
Figure 2.20 : Niveaux de modélisation pour la formalisation de la connaissance	81
Figure 2.21 : Spectre de la généralité.....	83
Figure 2.22 : Spectre revu de la généralité.....	84
Figure 2.23 : Spectre de réutilisabilité par [ROBINSON 04].....	87
Figure 2.24 : Evolution de la difficulté de réutilisabilité.....	89
Figure 3.1 : Nombre de publications par période	93
Figure 3.2 : Evolution du nombre de publications pour [Lent 12] après 2000	94
Figure 3.3 : Modèle conceptuelle du réseau « Urgence » de la ville de Nottingham	95
Figure 3.4 : Généalogie des thématiques d’études sur les hôpitaux	99
Figure 3.5 : (a) Modèle du circuit actuel (AS-IF) ; (b) Modèle futur du circuit (TO-BE).....	101
Figure 3.6 : Modèle en réseau de Pétri coloré du CDM	102

Figure 3.7 : Modèle AUTOMOD de la pharmacie par Huet (2011)	102
Figure 4.1 : Vue schématique des domaines et sous-domaines	108
Figure 4.2 : La méthodologie ASCI d'un point de vue domaine (gauche) et du point de vue d'un système du domaine (droite)	109
Figure 4.3 : Vue en cycle d'ASCI pour l'étude d'un système du domaine.....	110
Figure 4.4 : Interactions entre les sous-systèmes d'ASCI.....	112
Figure 4.5 : Représentation basique d'une Activité	117
Figure 4.6 : Exemple d'une Activité de compensation liée à son activité de base	118
Figure 4.7 : Illustration des marqueurs sur une Tâche pour les trois comportements.....	118
Figure 4.8 : « Activité appellante » pour une Tâche (a) ou un processus (b).....	119
Figure 4.9 : Sous-processus réduit (a) ou étendu (b)	119
Figure 4.10 : Sous processus non valide (en haut) et valide (en bas)	119
Figure 4.11 : Ensemble des comportements pour un Sous-processus	120
Figure 4.12 : Symbole utilisé pour la Transaction	120
Figure 4.13 : Sous-processus événementiel seul (a) et incorporé dans un autre sous-processus (b)	120
Figure 4.14 : Illustration d'une Tâche de type « Service »	121
Figure 4.15 : Illustration d'une Tâche de type « Envoi »	121
Figure 4.16 : Illustration d'une Tâche de type « Réception »	121
Figure 4.17 : Illustration d'une Tâche de type « Utilisateur »	121
Figure 4.18 : Illustration d'une Tâche de type « Manuelle »	122
Figure 4.19 : Symbole d'un Evènement de Début.....	122
Figure 4.20 : Evènement de Début : (a) Message, (b) Timer, (c) Conditionnel, (d) Signal	123
Figure 4.21 : Evènement de début Interruptible et Non-Interruptible.....	123
Figure 4.22 : Evènement Intermédiaire de base	123
Figure 4.23 : Evènement Intermédiaire de type « Lien » (a) ou « Multiple Parallèle » (b).....	124
Figure 4.24 : Evènement Intermédiaire de type « Timer » Interruptible et Non-Interruptible.....	124
Figure 4.25 : Symbole de base d'un Evènement de Fin	125
Figure 4.26 : Evènement de fin de type « Erreur » (a), « Multiple » (b) et « Arrêt » (c).....	125
Figure 4.27 : Symbole de base d'une Porte décisionnelle	125
Figure 4.28 : Représentaion d'une porte Exclusive divergente avec les deux symboliques.....	126
Figure 4.29 : Représentation d'une porte exclusive convergente avec la marque.....	126
Figure 4.30 : Représentation d'une porte Inclusive divergente.....	127
Figure 4.31 : Représentation d'une porte Parallèle divergente (à gauche) et convergente (à droite).....	127
Figure 4.32 : Représentation d'une porte Complexe convergente (a) et divergente (b).....	128
Figure 4.33 : Représentation d'une porte divergente basée sur un Evènement Intermédiaire « Multiple »	129
Figure 4.34 : Illustration d'un « Pool »	129
Figure 4.35 : Illustration d'un « Pool » contenant deux « Lignes »	130
Figure 4.36 : Représentation d'un objet de donnée (a) et d'une collection de données (b).....	130
Figure 4.37 : Représentation d'un Entrepôt de données.....	130
Figure 4.38 : Représentation d'une Donnée d'entrée (a) et d'un Donnée de sortie (b).....	131
Figure 4.39 : Représentation d'une Association suivant la situation	131
Figure 4.40 : Flux de séquence standard (a), conditionnel (b) ou par défaut (c).....	132
Figure 4.41 : Représentation d'un Flux de message	132
Figure 4.42 : Degré d'utilisation des éléments de BPMN.....	134

Figure 4.43 : Degré d'utilisation des éléments graphiques de BPMN par groupe	134
Figure 4.44 : Domaine d'utilisation de BPMN	135
Figure 5.1 : Représentation numérique de l'équivalent d'ASCI	141
Figure 5.2 : Etapes d'adaptation d'ASCI en vue de l'obtention de notre démarche.....	141
Figure 5.3 : Contenu des phases 1 (a), 2 (b), 3 (c) et 4 (d) de notre démarche.....	143
Figure 5.4 : Vue graphique de notre démarche sans (a) et avec (b) les liens hérités d'ASCI	144
Figure 5.5 : Vue graphique complète de notre démarche	145
Figure 5.6 : Comparaison graphique entre ASCI et notre démarche	147
Figure 6.1 : Macro-processus du CDM issu de [HAS 11]	152
Figure 6.2 : Liste des activités réalisées par les différents acteurs du CDM [HAS 11]	153
Figure 6.3 : Ebauche de la structure des processus composant le CDM : (a) unité de soins, (b) pharmacie.....	160
Figure 6.4 : Vue mésoscopique du processus du CDM	163
Figure 6.5 : Détails des activités sur le segment E1 – D2	164
Figure 6.6 : Détails des activités sur les segment D2 – E2/D4.....	164
Figure 6.7 : Détails des activités sur la partie « Contrôle » des segment E2/D4 – E9.....	165
Figure 6.8 : Détails des activités sur la partie « Commande » des segment E2/D4 – E9	165
Figure 6.9 : Détails des activités sur le segment E9 – E10.....	166
Figure 6.10 : Détails des activités sur le segment E10 – D10	166
Figure 6.11 : Détails des activités sur le segment D10 – Fin	167
Figure 6.12 : Détails des activités sur le segment E21 – D6	167
Figure 6.13 : Détails des activités sur le segment D6 – D8.....	167
Figure 6.14 : Détails des activités sur le segment D8 – Fin	168
Figure 6.15 : Détails des activités sur le segment Demande réappro – Fin	168
Figure 6.16 : Détails des activités sur le segment D6 – E6	169
Figure 6.17 : Détails des activités sur le segment D7 – D8 (Sous bloc (a)).....	169
Figure 6.18 : Détails des activités sur le segment D8 – D5 (Sous bloc (b)).....	169
Figure 6.19 : Détails des activités sur le segment D5 – Fin	170
Figure 6.20 : Détails des activités sur le segment E18/E19 – Fin	170
Figure 6.21 : Nouveau formalisme pour les messages.....	172
Figure 6.22 : Nouvelles portes proposées pour compléter BPMN : disponibilité d'une ressource humaine (a) et non humaine (b)	172
Figure 6.23 : Portes « mathématiques » proposées pour compléter BPMN	173
Figure 6.24 : Exemple de diagramme de chorégraphie avec le rôle des différents messages	173
Figure 6.25 : Première proposition pour lier plusieurs acteurs à une même activité	174
Figure 6.26 : Seconde proposition d'amélioration.....	174
Figure 6.27 : Formalisme macroscopique proposé et sa traduction en microscopique.....	175
Figure 6.28 : Exemple d'application du formalisme macroscopique	175
Figure 6.29 : Diagramme des classes conçu par [RODIER 10].....	177
Figure 6.30 : Sous-système Décisionnel conçu par [HUET 11].....	178
Figure 6.31 : Sous-système Logique conçu par [HUET 11].....	178
Figure 6.32 : Sous-système Physique conçu par [HUET 11].....	179
Figure 6.33 : Diagramme des classes générique conçu pour notre démarche	181
Figure 7.1 : Interface « Facility » positionnée sur l'option « Run ».....	191
Figure 7.2 : Aperçu de l'interface « Processes »	193

Figure 7.3 : Aperçu de l'interface « Définitions ».....	194
Figure 7.4 : Aperçu de l'interface « Data » avec une table de séquence.....	196
Figure 7.5 : Aperçu de l'interface « Experiment ».....	196
Figure 7.6 : Modèle SIMIO de la Source	199
Figure 7.7 : Modèle SIMIO du Puits.....	200
Figure 7.8 : Modèle SIMIO du Serveur	200
Figure 7.9 : Modèle SIMIO de la Station de travail	202
Figure 7.10 : Modèle SIMIO de la Ressource	203
Figure 7.11 : Modèle SIMIO du Travailleur	204
Figure 7.12 : Modèle SIMIO du Combineur (a) et du Séparateur (b).....	205
Figure 7.13 : Modèle SIMIO d'un Véhicule.....	206
Figure 7.14 : Représentation d'un nœud basique et d'un nœud de transfert.....	208
Figure 7.15 : Vue de l'un des modèles SIMIO pour le Convoyeur.....	209
Figure 7.16 : Modèle SIMIO du processus de départ (a) et d'un processus avec étapes (b).....	209
Figure 7.17 : Processus de départ (a) et avec la fonction test « Poids ? » (b).....	210
Figure 7.18 : Aperçu des propriétés du « Decide » « Poids ? ».....	211
Figure 7.19 : Vue du processus complet avec assignation des valeurs pour Num_MO	211
Figure 7.20 : Aperçu des propriétés des « Assign » utilisés	211
Figure 7.21 : Vue de l'ébauche du modèle SIMIO	212
Figure 7.22 : Aperçu des propriétés de « Porte_entree » (a) et « Bureau_des_entrees » (b)	213
Figure 7.23 : Vue du modèle SIMIO en fonctionnement	213
Figure 24 : Activités utiles avant réduction.....	214
Figure 7.25 : Activités utiles avant après réduction	215
Figure 7.26 : Vue par bloc de la pharmacie.....	215
Figure 7.27 : Vue affinée par bloc de la pharmacie.....	215
Figure 7.28 : Matrice cubique de sélection du type d'acteur sous SIMIO	221
Figure 7.29 : Choix associés à la matrice cubique de la Figure 7.28	222
Figure 8.1 : Vue globale avec les blocs du modèle BPMN.....	228
Figure 8.2 : Vue globale avec la décomposition des blocs 3, 4 et 5	229
Figure 8.3 : Cheminement possible des demandes dans le modèle de simulation	229
Figure 8.4 : Représentation des valideurs et préparateurs (a), et des agents logistiques (b)	230
Figure 8.5 : Représentation sous SIMIO de la « Prescriptions » (a) et de la « Demande » (b)	231
Figure 8.6 : Méthode de décomposition d'une demande en sous-demandes	233
Figure 8.7 : Source sous SIMIO des demandes (hors urgences et réapprovisionnement)	233
Figure 8.8 : Vue de la table temporelle (à droite) et de la table des facteurs (à gauche).....	234
Figure 8.9 : Vue du processus « Choix_planning » de la Source	236
Figure 8.10 : Processus définissant le temps de transport d'une demande	237
Figure 8.11 : Vue de la source urgente, de la demande urgente et du nœud de sortie de la source	237
Figure 8.12 : Séparateur prescription/demande (en bas à gauche) et du nœud d'orientation (en haut à droite).....	238
Figure 8.13 : Vue du processus « Choix_apres_creation »	241
Figure 8.14 : Méthode de décomposition d'une prescription en sous-prescriptions.....	241
Figure 8.15 : Vue des trois séparateurs (en bas) permettant de décomposer les prescriptions.....	242
Figure 8.16 : Vue du processus définissant le temps de validation	244
Figure 8.17 : Vue du processus de gestion de la disponibilité des valideurs	245

Figure 8.18 : Vue de la zone validation	245
Figure 8.19 Vue du processus « Chargementdonneesvalideur ».....	247
Figure 8.20 : Vue du processus « Capacite_validation ».....	248
Figure 8.21 : Vue du processus « Selectionvalideur ».....	248
Figure 8.22 : Vue du processus « liberation_valideur ».....	249
Figure 8.23 : Vue du bloc de répartition des prescriptions dans le modèle	249
Figure 8.24 : Vue du processus « Choix_mode_prepa ».....	251
Figure 8.25 : Vue du processus « Creerpreparateur »	253
Figure 8.26 : Vue du processus de sélection des préparateurs « Selection_prepa »	254
Figure 8.27 : Vue des serveurs représentant chacun un cycle de préparation.....	255
Figure 8.28 : Vue du processus « Choix_apres_preparation ».....	257
Figure 8.29 : Vue du bloc « Edition/Contrôle ».....	258
Figure 8.30 : Vue du processus « Capacite_Edition »	260
Figure 8.31 : Vue du processus d'attente d'un préparateur « Attente_select_prepa ».....	261
Figure 8.32 : Vue du processus « Choix_apres_edition »	261
Figure 8.33 : Vue des séparateurs utilisés pour décomposer les demandes en sous-demandes.....	262
Figure 8.34 : Vue du processus « Choix_apres_assignment_duplication »	264
Figure 8.35 : Vue du processus « Rech_prepa_associe ».....	265
Figure 8.36 : Vue des serveurs de picking manuel (a), de stockeur mécanique (b) et de la chambre froide (c).....	265
Figure 8.37 : Vue du processus de libération des préparateurs « Liberation_prepa ».....	266
Figure 8.38 : Vue des trois éléments réunissant les demandes et les expédiant	266
Figure 8.39 : Vue du processus recherchant le temps de transport des demandes.....	268
Figure 8.40 : Vue du processus d'enregistrement des données « Enregistrer_donnees ».....	269
Figure 8.41 : Vue de la chaîne de reconditionnement unitaire des produits	269
Figure 8.42 : Vue du processus surveillant la consommation de produits	271
Figure 8.43 : Vue du processus mettant à jour la capacité de production	272
Figure 8.44 : Vue du processus gérant la capacité de production du serveur « Reconditionnement »	272
Figure 8.45 : Découpage de chaque journée sous SIMIO	275
Figure 8.46 : Matrice Jour/Créneau représentant la présence des préparateurs sur la semaine	276
Figure 8.47 : Exemple de remplissage du pourcentage d'apparition des demandes sur les créneaux	277
Figure 8.48 : Exemple de remplissage de la « Table_transport »	280
Figure 8.49 : Interface « Data » avec l'option « Bind to » sélectionnée	281
Figure 8.50 : Vue de la fenêtre de sélection du fichier EXCEL.....	281
Figure 8.51 : Page d'accueil de l'interface.....	281
Figure 8.52 : Vue de la page de saisie des informations sur les demandes	282
Figure 8.53 : Ecran de saisie des informations sur les sous-demandes et sous-prescriptions	282
Figure 8.54 : Vue du planning de présence préparateur	283
Figure 9.1 : Processus avant (en rouge) et après évolution (en vert)	297
Figure 8.2 : Vue du SSPDL utilisé dans de le cas de la DJINM (parties non grisées).....	297
Figure 9.3 : Vue des classes « Demande nominative » et « Demande de service »	298
Figure 9.4 : Charge de travail des préparateurs.....	301
Figure 9.5 : Nombre de demandes préparées par chaque préparateur	302

Figure 9.6 : Vue du processus actuel (en rouge) et de la DJINA (en bleu)	303
Figure 9.7 : Caractéristiques principales du robot de dispensation.....	304
Figure 9.8 : Vue du SSPDL utilisé dans de le cas de la DJINA (parties non grisées).....	304
Figure 9.9 : Charges de travail des préparateurs	307
Figure 9.10 : Nombre de demandes préparées par le robot	308
Figure 9.11 : Nombre de demandes préparées par chaque préparateur	308
Figure 9.12 : Vue du processus de validation (en orange)	310
Figure 9.13 : Informations liées à la classe Valideur	311
Figure 9.14 : Vue du SSPDL utilisé dans de le cas de la DJINA avec validation (parties non grisées) .	311
Figure 9.15 : Nombre de prescriptions validées suivant le nombre de valideurs à 10h	314
Figure 9.16 : Résultats pour la Situation 1-Configuration 2	314
Figure 9.17 : Résultats pour la Situation 2-Configuration 2	315
Figure 9.18 : Résultats pour la Situation 2-Configuration 3	316
Figure 9.19 : Résultats pour la Situation 3	317
Figure 9.20 : Résultats pour la Situation 4 – Configuration 3.....	318
Figure A.1 : Répartition des erreurs par type entre 2005 et 2008	338
Figure A.2 : Répartition des causes d’erreurs entre 2005 et 2008.....	339
Figure A.3 : Vue hiérarchique du taux d’interception des erreurs à chaque étape du CDM.....	341
Figure A.4 : Critères clés de la généricité durant l’initiation du projet	342
Figure A.5 : Critères clés de la généricité durant la conception et le développement	342
Figure A.6: Critères clés de la généricité pour les données	343
Figure A.7 : Critères clés de la généricité durant la validation.....	343
Figure A.8 : Critères clés de la généricité durant l’implémentation	344
Figure A.9 : Ensemble des évènements existants dans BPMN 2.0	345

LISTE DES TABLEAUX

Tableau 1.1 : Liste des étapes, acteurs et actions du CDM.....	9
Tableau 1.2 : Lois et articles principaux régissant chaque étape du CDM.....	11
Tableau 1.3 : Critères sous-jacents à chaque type de dispensation / délivrance	13
Tableau 1.4 : Nombre d'erreurs à chaque étape déclarées au GEM de l'Afssaps	24
Tableau 1.5 : Part moyenne des erreurs issues des grandes étapes du CDM	27
Tableau 1.6 : Répartition des causes de l'iatrogénie médicamenteuse.....	27
Tableau 1.7 : Localisation des EIG	32
Tableau 1.8 : Répartition des EIG par année par type d'incident	33
Tableau 1.9 : Répartition des EIG par année et combinaison d'incident.....	33
Tableau 1.10 : Répartition des EIG par type et année pour 1000 jours d'hospitalisation	33
Tableau 1.11 : Répartition des EIG par combinaison d'incident et année pour 1000 jours d'hospitalisation	34
Tableau 2.1 : Lien entre documents gouvernementales et processus	54
Tableau 2.2 : Matrice Objectifs / Granularité [Rodier 10].....	58
Tableau 2.3 : Matrice Objectifs / Granularité	59
Tableau 2.4 : Les différents niveaux de processus.....	59
Tableau 2.5 : Exemple des différents niveaux de processus.....	60
Tableau 2.6 : Notation des différents langages.....	67
Tableau 2.7 : Note obtenue pour chaque langage.....	67
Tableau 2.8 : Facteurs clés dans la conception et la construction d'un modèle générique ou spécifique	85
Tableau 2.9 : Exemple de critères pour les quatre autres thématiques.....	86
Tableau 3.1 : Articles passant en revue la simulation hospitalière.....	93
Tableau 5.1 : Bilan des besoins pour chaque phase de notre démarche.....	145
Tableau 5.2 : Lieu d'utilisation de nos outils dans ASCI et notre démarche	146
Tableau 6.1 : Répartition des acteurs par « Pool »	154
Tableau 6.2 : Evènements rythmant l'avancée du processus.....	157
Tableau 6.3 : Décisions orientant l'évolution du processus.....	158
Tableau 6.4 : Actions réalisées sur chaque segment du processus de l'unité de soins	161
Tableau 6.5 : Actions réalisées sur chaque segment du processus de la pharmacie.....	162
Tableau 7.1 : Liste des informations présentes dans l'interface de sortie.....	188
Tableau 7.2 : Cycles de préparation et impact.....	189
Tableau 7.3 : Récapitulatif des leviers de simulation.....	190
Tableau 7.4 : Situation d'exécution des processus	192
Tableau 7.5 : Principaux types d' « Elements ».....	194
Tableau 7.6 : Objets appartenant à la librairie standard de SIMIO.....	198
Tableau 7.7 : Aperçu des propriétés basiques d'une Source	200
Tableau 7.8 : Aperçu des propriétés basiques d'un Serveur.....	201
Tableau 7.9 : Aperçu des propriétés basiques de la Station de travail	202
Tableau 7.10 : Aperçu des propriétés basiques de la Ressource	203

Tableau 7.11 : Aperçu des propriétés du Travailleur	204
Tableau 7.12 : Aperçu des propriétés basiques du Combineur	205
Tableau 7.13 : Aperçu des propriétés basiques du Séparateur	205
Tableau 7.14 : Aperçu des propriétés basiques du Véhicule	206
Tableau 7.15 : Aperçu des propriétés basique d'un Nœud de transfert	207
Tableau 7.16 : Aperçu des propriétés basiques d'un Chemin.....	208
Tableau 7.17 : Eléments composant le système réel et le modèle SIMIO	212
Tableau 7.18 : Question et réponses possibles durant le processus de préparation	220
Tableau 7.19 : Question et réponses possibles durant le processus de réapprovisionnement	221
Tableau 7.20 : Règles de passage principales de BPMN à SIMIO	224
Tableau 8.1 : Echantillon des propriétés de l'élément « Prescriptions »/« Demande ».....	231
Tableau 8.2 : Propriétés principales de la source du Service	234
Tableau 8.3 : Echantillon des caractéristiques de la « Demande » fournies par le « Service »	235
Tableau 8.4 : Propriétés des nœuds « Demande_ou_prescription » et « Hybridation_possible »	239
Tableau 8.5 : Poids appliqués aux chemins ayant pour origine l'un des nœuds présents.....	242
Tableau 8.6 : Données nécessaires dans le cadre de la validation	246
Tableau 8.7 : Données nécessaires pour la distribution des prescriptions.....	250
Tableau 8.8 : Données utilisées en sortie des serveurs de préparation robotisée	256
Tableau 8.9 : Données nécessaires pour le serveur « EditionControle ».....	259
Tableau 8.10 : Poids des chemins utilisés pour l'orientation des demandes	263
Tableau 8.11 : Propriétés associées aux éléments de regroupement, de transport et d'expédition.	267
Tableau 8.12 : Données associées aux éléments de regroupement, de transport et d'expédition ...	267
Tableau 8.13 : Propriétés liées au reconditionnement	270
Tableau 8.14 : Données associées au reconditionnement.....	271
Tableau 8.15 : Echantillon des états du modèle	273
Tableau 8.16 : Données comprises dans la table « Donneessimulation »	274
Tableau 8.17 : Données liées à la préparation automatisée.....	275
Tableau 8.18 : Données caractérisant les demandes parcourant notre modèle	277
Tableau 8.19 : Données relatives à la création des sous-demandes/sous-prescriptions	278
Tableau 8.20 : Données concernant le retour produits des services.....	279
Tableau 8.21 : Données concernant le reconditionnement de produits	279
Tableau 8.22 : Données concernant le transport des demandes	280
Tableau 9.1 : Réponses aux facteurs clés de [FLETCHER 09a] pour l'initiation d'un projet	288
Tableau 9.2 : Réponses aux facteurs clés de [FLETCHER 09a] pour le design et la conception.....	289
Tableau 9.3 : Réponses aux facteurs clés de [FLETCHER 09a] pour les données.....	290
Tableau 9.4 : Réponses aux facteurs clés de [FLETCHER 09a] pour la validation	290
Tableau 9.5 : Réponses aux facteurs clés de [FLETCHER 09a] pour l'implémentation	291
Tableau 9.6 : Variation du niveau de réutilisabilité suivant l'objectif poursuivi pour Robinson	292
Tableau 9.7 : Variation du niveau de réutilisabilité suivi l'objectif poursuivi pour Taylor.....	293
Tableau 9.8 : Répartition des préparateurs sur une journée de travail.....	294
Tableau 9.9 : Paramètres des demandes globalisées	295
Tableau 9.10 : Données nécessaires pour une simulation en DJIN manuelle.....	298
Tableau 9.11 : Charge des préparateurs pour 10% des lits en DJINM	301
Tableau 9.12 : Données liées au robot pour la production automatisée	305
Tableau 9.13 : Nombre de demandes nominatives et d'UF traitées par la pharmacie	306

Tableau 9.14 : Charge des préparateurs et du robot en DJINA	307
Tableau 9.15 : Données liées à l'analyse et la validation d'une prescription	312
Tableau 9.16 : Données liées aux valideurs	312
Tableau 9.17 : Récapitulation des différentes situations.....	313
Tableau 9.18 : Résultats pour la Situation 1-Configuration 1	313
Tableau 9.19 : Résultats pour la Situation 1-Configuration 2	314
Tableau 9.20 : Résultats pour la Situation 2-Configuration 2	315
Tableau 9.21 : Résultats pour la Situation 2-Configuration 3	316
Tableau 9.22 : Résultats pour la Situation 3.....	317
Tableau 9.23 : Résultats pour la Situation 4 – Configuration 3.....	317
Tableau A.1 : Nombre d'erreurs par type entre 2005 et 2008	338
Tableau A.2 : Origine des erreurs entre 2005 et 2008	339
Tableau A.3 : Nature des causes des EM.....	340
Tableau A.4 : Vue détaillée des causes des EM.....	340
Tableau A.5 : Ensemble des propriétés de l'élément « Prescriptions »/ « Demande ».....	346
Tableau A.6 : Assignation des états des demandes/prescriptions par la source « Service ».....	347

LISTE DES ILLUSTRATIONS

Image 1.1 : (a) Stockeur rotatif vertical (b) stockeur avec tapis.....	47
Image 1.2 : Robot de dispensation de McKesson.....	48
Image 1.3 : Robot de dispensation d'EURAF	48
Image 1.4 : (a) PACMED de McKesson (b) Autoprint de EURAF	48
Image 1.5 : Armoires et chariots informatisés et sécurisés	49

LISTE DES ACRONYMES

AFSSAPS	Agence Française de Sécurité Sanitaire des Produits de Santé
ANAP	Agence Nationale d'Appui à la Performance
ANSM	Agence Nationale de Sécurité du Médicament et des produits de santé
ATU	Autorisation Temporaire d'Utilisation
CDM	Circuit Du Médicament
CHIC	Centre Hospitalier Intercommunale de Cornouaille
CMP	Centre Médico-Psychiatrique
CSP	Code de la Santé Publique
CTM	Conciliation des Traitements Médicamenteux
DCI	Dénomination Commune Internationale
DGA	Dispensation Globalisée Automatisée
DgM	Distribution globale Manuelle
DGM	Dispensation Globalisée Manuelle
DGOS	Direction Générale de l'Offre de Soins
DJIN	Dispensation Journalière Individuelle Nominative
Dms	Durée moyenne de séjour
DM	Dispositifs Médicaux
DNMP	Dispensation Nominative Manuelle Partielle
EI(G)	Evènement Indésirable (Grave)
EIM	Evènement Indésirable Médicamenteux
ErM	Erreur médicamenteuse
ES	Etablissement de Santé
FFI	Faisant Fonction d'Interne
GEM	Guichet Erreurs Médicamenteuses
GHS	Groupement Homogène de Séjour
GM	Gabriel Montpied
HAS	Haute Autorité de Santé
HPST	Hôpital, Patient, Santé et Territoire
MDS	Médicaments Dérivés du Sang
MEAH	Mission nationale d'Expertise et d'Audit Hospitalier
MGC	Modèle Générique de Connaissance
MT2A	Mission de Tarification à l'Activité
OMEDIT	Observatoire des Médicaments, des Dispositifs Médicaux et des Innovations thérapeutiques
PUI	Pharmacie à Usage Intérieur
SI	Système d'information
US	Unité de Soins

INTRODUCTION GENERALE

Désormais faire évoluer le fonctionnement d'une organisation ne se révèle pas chose facile du fait des nombreuses possibilités et évolutions – organisationnelles, structurelles, technologiques – qui s'offrent aux décideurs mais aussi du fait du nombre croissant de contraintes – notamment économiques – auxquels ils doivent faire face durant cette démarche. Le contexte se complexifie encore quand ces décideurs ne sont pas coutumiers de ces différentes possibilités et ne peuvent donc prendre efficacement une décision sans risquer l'erreur. Cette problématique se retrouve tout particulièrement dans le milieu hospitalier où nombre de praticiens se doivent de choisir l'orientation de leurs services sans être pour autant au courant de tous les tenants et aboutissants que ce choix va impliquer pour la vie future de leur unité. Ces difficultés prennent tout leur sens quand les pharmaciens se doivent de fournir un plan d'évolution de la pharmacie, notamment dans le domaine d'application de cette thèse : le circuit du médicament.

Dès lors, avoir à disposition les méthodes et outils permettant de comprendre ce que chaque décision impliquera serait et devrait être un impératif, ce qui n'est malheureusement que très peu le cas, l'absence de tels dispositifs étant des plus flagrantes.

Ainsi, dans le but de faciliter ces choix et de permettre cette réorganisation, nous proposerons un outil d'aide à la décision à destination des pharmaciens, qui sera conçu grâce à une méthodologie de modélisation analysant le système tel qu'il est et tel qu'il devra être. Plusieurs problématiques viennent donc s'opposer à nous dans le cadre de ces recherches.

Tout d'abord, il va être nécessaire de comprendre comment fonctionne le circuit du médicament mais aussi l'origine des nombreux problèmes qu'il subit, afin de pouvoir y apporter des solutions. Nous nous appuyerons alors sur des rapports, tels que [DHOS 09], qui fournissent déjà de nombreuses pistes d'améliorations parmi lesquelles nous retrouvons l'informatisation du circuit, sa robotisation, de nouvelles méthodes de gestions des stocks, etc. tout cela dans le but d'améliorer la performance et la sécurité de ce circuit sensible.

La deuxième problématique sera pour nous de voir comment nous pouvons représenter le circuit du médicament afin de l'étudier et de l'améliorer. En effet, même en ayant identifié les problèmes de ce circuit, il nous sera impossible de travailler dessus si nous ne pouvons comprendre avec précision sa structure, mais aussi si nous ne disposons pas de la méthodologie adéquate pour réaliser cette analyse. Nous montrerons donc par l'étude et l'examen de divers langages, méthodes et méthodologies, comment la représentation de ce circuit est possible.

Dès lors, une nouvelle question apparaît. Comment valider sans risque les nombreux changements que les pharmaciens désireront appliquer au circuit du médicament ? Plusieurs techniques (modèles mathématiques, numériques, etc.) ou méthodes (tests réels, simulation, etc.) existent, mais il sera cependant nécessaire d'en choisir une, cette dernière devant être la plus adéquate et performante possible.

Ce choix fait, nous porterons notre attention sur la connaissance acquise jusqu'à ce jour, afin de valider ou non cette décision, puis nous construirons les divers éléments permettant d'atteindre l'objectif cité précédemment, à savoir proposer, a minima, un outil d'aide à la décision.

Les travaux présentés dans ce mémoire de thèse apporteront donc des réponses à ces nombreuses questions, qui visent à fournir aux pharmaciens un environnement de prise de décision le plus adéquat possible à leur situation. Comme nous le verrons, cet environnement prendra la forme d'une démarche, structurée en différentes étapes bien identifiées et qui sera basée sur un certain nombre de notions et d'outils déjà préexistants. Le but est donc de fournir une démarche complète comprenant les outils nécessaires (modèle, outil d'aide à la décision, etc.) pour que les pharmaciens n'aient plus à s'aventurer sur des chemins hors de leurs sphères de compétences, et qui pourra au final être vue comme une « **démarche d'aide à la décision** ».

La structure de ce mémoire de thèse sera la suivante :

Le **premier chapitre** présentera le contexte et les problématiques générales qui sont en lien avec le circuit du médicament. Une analyse de la composition et des grands principes de fonctionnement de ce dernier sera donc menée, avant de mettre en lumière les contraintes et problèmes majeurs que ce circuit rencontrent. S'en suivra alors une étude de l'origine et de l'importance des erreurs découlant des lacunes du circuit avant de terminer par une présentation des nombreuses recommandations et solutions qu'il est possible de mettre en œuvre pour apporter plus de sûreté et de performance à la dispensation et distribution des médicaments dans les structures hospitalières.

Le **second chapitre** sera quant à lui consacré à la prospection et à l'étude de nombreux langages, méthodes et méthodologies pouvant apporter une réponse à la première problématique que nous rencontrons, à savoir comment représenter le circuit du médicament. Cette partie sera précédée par une présentation de diverses notions – processus, modèle, etc. – qui nous suivront tout du long de ce mémoire de thèse et qu'il est donc impératif de clarifier d'entrée. Ces deux premières parties seront suivies par une discussion sur la simulation, qui reste pour nous, la méthode la plus adéquate pour apporter une réponse à la seconde problématique à laquelle nous ferons face, à savoir comment mettre efficacement en œuvre les évolutions du circuit choisies sans risque pour l'hôpital. Enfin, nous terminerons ce chapitre en abordant deux notions primordiales dans ces travaux, la généricité et la réutilisabilité.

Le **troisième chapitre**, faisant office d'état de l'art, détaillera ce qui a été réalisé jusqu'à aujourd'hui dans le domaine de la simulation hospitalière afin de mettre en avant les atouts de cette technique mais aussi afin de montrer qu'il y a un véritable manque de travaux poussés sur le domaine de la pharmacie hospitalière et tout particulièrement sur le circuit du médicament.

A travers le **quatrième chapitre**, nous commencerons à aborder la partie technique de nos travaux. Ce chapitre présentera donc le langage BPMN ainsi que la méthodologie ASCI, conséquence directe des choix faits dans le second chapitre. Le pourquoi de la méthodologie ASCI ainsi que chacune de ces étapes seront décrites avec précision, avant de détailler le langage BPMN en explicitant le rôle et l'utilisation de chaque item composant ce langage, pour terminer par un bilan rapide sur l'utilisation de BPMN dans le monde industriel et hospitalier.

Le **cinquième chapitre** concernera la construction de notre démarche dédiée aux pharmaciens, où il sera expliqué comment à partir de la méthodologie ASCI nous parvenons à obtenir cette nouvelle démarche.

Une fois la démarche et les outils nécessaires à son exécution connus, il est possible de commencer à concevoir ces derniers. Ce sera l'objet de ce **sixième chapitre** qui détaillera comment le modèle générique des processus a été construit et à partir de quels éléments le diagramme générique des classes UML a été généré. Nous présenterons quels choix ont été faits au niveau des acteurs, des activités, des événements, etc. pour représenter le plus fidèlement possible le CDM avec le langage BPMN et quelles ont été les évolutions menant à la création de notre diagramme des classes, celui-ci étant basé sur les travaux de [[RODIER 10](#)] et [[HUET 11](#)].

Le **septième chapitre** abordera les aspects techniques en vue de construire le troisième et dernier outil, à savoir le modèle générique de simulation. Diverses parties composent ce chapitre conséquent. Tout d'abord après une présentation des leviers de simulation, désirés par le CHU de Clermont-Ferrand, nous expliquerons dans sa globalité le logiciel SIMIO, en détaillant ses nombreuses fonctions mais aussi en présentant les divers éléments qui permettent de construire un modèle de simulation. Dans la dernière partie de ce chapitre, nous décrirons quelles ont été les grandes règles justifiant le passage de notre modèle générique sous BPMN à notre modèle générique de simulation sous SIMIO.

Le **huitième chapitre** portera par conséquent sur la conception même du modèle générique de simulation en présentant comment les divers éléments du circuit ont été traduits dans ce logiciel, quels choix ou simplifications ont été faits, mais aussi en détaillant les nombreux processus et formulations mathématiques qui régissent le fonctionnement des diverses entités dans le modèle.

Le **neuvième et ultime chapitre** traitera de l'évaluation de nos outils et de l'application de notre démarche. Cette évaluation portera sur la généricité et la réutilisabilité des dits outils en se référant aux travaux de [[FLETCHER 09a](#), [FLETCHER 09b](#)]. Cette partie sera suivie de l'application de notre démarche sur trois scénarios, portant chacun sur des problématiques différentes du circuit du médicament. Ainsi, pour chaque scénario l'utilisation et l'application de nos outils seront illustrées, avant de présenter et d'analyser les divers résultats obtenus par le biais de la simulation. L'objet de ce chapitre sera donc d'évaluer l'efficacité et la validité de notre modèle, vis-à-vis de la réalité, mais aussi de comprendre l'impact de certaines évolutions sur une pharmacie hospitalière.

Une conclusion, ponctuera la présentation de ces travaux, en dressant un bilan de la contribution de cette thèse et en présentant les perspectives futures de ce travail.

I

CONTEXTE ET PROBLÉMATIQUES

RESUME DU CHAPITRE 1

Dans ce chapitre, nous présentons le circuit du médicament et les différents acteurs qui y sont impliqués. Pour cela, nous détaillons les différents types, les différentes structures de circuit possibles ainsi que les nombreuses lois régissant son fonctionnement et encadrant rigoureusement les tâches de chaque acteur. Viendront ensuite le détail et la description des nombreuses contraintes et problèmes constatés au sein de ce circuit avant de décrire minutieusement les erreurs qui en découlent et leur impact sur le plan humain.

Il est donc constaté que le monde hospitalier possède encore de nos jours, un retard certain (mais moins important qu'au début des années 2000) dans le domaine de la gestion et de l'anticipation des risques en comparaison avec l'industrie notamment. Pour diminuer de manière conséquente les erreurs ayant lieu au sein du circuit, il est nécessaire de sécuriser le fonctionnement de celui-ci et cela sur de nombreux plans. La sécurisation passe donc par le développement de systèmes permettant d'épauler les acteurs dans leur travail de tous les jours mais aussi par le développement de standards nécessaires à l'évolution des métiers gravitant autour de ce domaine. Dès lors celle-ci peut prendre de nombreuses formes : robotisation des tâches, informatisation du circuit, nouvelle politique de gestion, etc.

SOMMAIRE DU CHAPITRE

1.1	INTRODUCTION	7
1.2	QU'EST CE QUE LE CIRCUIT DU MEDICAMENT ?	7
1.2.1	Définitions	7
1.2.2	Les étapes du CDM	9
1.2.3	Un cadre réglementaire strict	11
1.2.4	Les différents modes de délivrance existants	12
1.3	LE CDM : CONTRAINTS ET SOUS PRESSION	18
1.3.1	Introduction.....	18
1.3.2	Les contraintes du CDM	19
1.3.3	Les problèmes constatés	22
1.3.4	Erreurs et Evènements Indésirables.....	23
1.3.5	Classification des EIM et étude ENEIS.	31
1.4	COMMENT SECURISER LE CDM ?	34
1.4.1	Les recommandations ministérielles et les lois.....	35
1.4.2	Que faire à chaque étape du CDM d'après ces lois?.....	37
1.4.3	Quelles voies de sécurisation ?	41
1.5	CONCLUSION	50

1.1 INTRODUCTION

Ce chapitre a pour objectif de décrire dans quel contexte ce travail de thèse a eu lieu en se recentrant tout particulièrement sur le circuit du médicament (noté **CDM**) et sa structure, la sécurisation de celui-ci étant une priorité de santé publique. En effet, une connaissance accrue des tenants et des aboutissants du fonctionnement de ce circuit est primordiale pour pouvoir en extraire toutes les subtilités et les contraintes avec lesquelles il se doit de composer. Dans un premier temps, le CDM est présenté de manière globale avant d'en visualiser les grandes étapes, le cadre réglementaire et les différents types d'organisations existants. S'en suit une identification des contraintes majeures auxquelles le CDM est soumis pour pouvoir fonctionner, puis une analyse des erreurs pouvant apparaître à chaque étape de ce circuit. Pour terminer, sont présentées les grandes lignes de sécurisation préconisées par de nombreux rapports et études menés par diverses institutions d'état sur le CDM.

1.2 QU'EST CE QUE LE CIRCUIT DU MEDICAMENT ?

1.2.1 Définitions

Tout comme son nom le laisse penser, le CDM peut être représenté par un ensemble de flux (médicaments, dispositifs médicaux, etc.) qui transitent par des *Pharmacies à Usage Intérieur* (noté **PUI**), existantes dans chaque hôpital, à destination de nombreuses unités de soins (noté **US**) qui les composent. Ces premières, gestionnaires et responsables des stocks de ces produits, sont sous le contrôle et sous la responsabilité directe des pharmaciens hospitaliers.

Du fait de sa pluridisciplinarité et de sa transversalité, la définition même de CDM pose certains problèmes suivant la personne et le niveau auquel on se place. Les deux définitions du CDM suivantes permettent de se faire une idée de cette difficulté et des ambiguïtés qui peuvent apparaître dans une tâche qui semble pourtant simple.

Définition 1.1 : « En pratique, le circuit du médicament dans les hôpitaux et les cliniques est un macro-processus de soins complexe, transversal, jalonné d'étapes mobilisant des professionnels différents : la prescription est un acte médical, la dispensation, un acte pharmaceutique et l'administration, un acte infirmier ou médical. » [[CAUTERMAN 07](#)]

Définition 1.2 : « Le circuit du médicament est défini comme étant le processus de la prise en charge thérapeutique médicamenteuse d'un patient, hospitalisé dans un établissement de santé, résident d'un établissement social ou en soins ambulatoires. Ce processus interdisciplinaire est constitué par les étapes de prescription, de dispensation, d'administration et de suivi thérapeutique, et comprend le traitement de l'information. » [[SCHMITT 06](#)]

Ces deux définitions se différencient notamment par la présence dans la seconde de la notion de « traitement de l'information » qui est désormais primordiale dans la gestion qu'un hôpital peut avoir du parcours patient en son sein.

De ces définitions, il est alors possible de mettre en lumière certains items et caractéristiques du CDM :

- ↪ Une mission aux règles strictes : le CDM a pour obligation de répondre à des exigences fortes pour garantir la sécurité du patient, notamment durant l'étape de dispensation, où il se doit de donner le bon médicament au bon patient, sous la bonne forme et au bon moment quel que soit l'environnement dans lequel il se situe. Le cadre réglementaire du CDM sera présenté de manière plus approfondi dans la **partie 1.2.3**.
- ↪ Dispensation et délivrance : ces deux notions portent malheureusement souvent à confusion. En se référant à l'article R.4235-48¹ du code de la santé publique, il est possible de mieux cerner celles-ci :

Définition 1.3 : « Le pharmacien doit assurer dans son intégralité l'acte de dispensation du médicament, associant à sa délivrance :

- ✓ L'analyse pharmaceutique de l'ordonnance médicale si elle existe ;
- ✓ La préparation éventuelle des doses à administrer ;
- ✓ La mise à disposition des informations et les conseils nécessaires au bon usage du médicament.

Il a un devoir particulier de conseil lorsqu'il est amené à délivrer un médicament qui ne requiert pas une prescription médicale.

Il doit, par des conseils appropriés et dans le domaine de ses compétences, participer au soutien apporté au patient. »

La délivrance consistera en l'acte de remettre un produit pharmaceutique à une US ou tout autre entité ayant le droit de faire une telle demande. Ainsi en complément de cette délivrance, l'acte de dispensation inclut donc des actions telles que l'analyse pharmaceutique, le suivi pharmaceutique, etc.

- ↪ Transversal et pluridisciplinaire : Le CDM est considéré comme l'un des procédés les plus transversaux, critiques et structurants avec lequel un établissement de santé doit composer. En effet, celui-ci est initié au niveau des services de soins par la prescription (médecin) et les soins (infirmiers), permet par le biais des demandes, d'approvisionner, de délivrer les médicaments (pharmacien et préparateur) aux US avant leur administration au patient (infirmier).

La multiplicité de ces acteurs (médecin, infirmier, pharmacien, préparateur, aide-soignant, patient, etc.) qui possèdent chacun un point de vue, un niveau de responsabilité et des exigences par rapport au fonctionnement et à l'organisation existante, ne fait que rendre plus complexe cette transversalité et donc la bonne marche du CDM.

En conclusion, le CDM est un circuit soumis à de fortes contraintes inhérentes à son fonctionnement et aux entités qui le constituent (voir **partie 1.3.2** et **1.3.3**), qui se doit de composer et de limiter au maximum les erreurs qui ont lieu en son sein (voir **partie 1.3.4** et **1.3.5**), et qui pour

¹ Code de la santé publique,

<http://www.legifrance.gouv.fr/affichCodeArticle.do?idArticle=LEGIARTI000006913703&cidTexte=LEGITEXT000006072665&dateTexte=20081224>

cela, est soumis à des objectifs de sécurisation portant autant sur sa structure organisationnelle, qu'humaine et décisionnelle (voir [partie 1.4](#)).

1.2.2 Les étapes du CDM

A partir des éléments précédemment cités, et en se basant sur les nombreux articles et publications ministérielles existants, il est dès lors possible de schématiser le déroulement de la prise en charge thérapeutique médicamenteuse d'un patient ayant pour but l'utilisation sécurisée, appropriée et efficiente d'un médicament chez celui-ci. En se basant notamment sur le document [[DGOS 12](#)], guide sur la qualité de la prise en charge médicamenteuse, les étapes suivantes sont identifiables :

- ❖ La prescription (en prenant en compte la gestion du traitement personnel du patient dès son admission, ainsi que les vérifications des prescriptions lors des transferts de patient, la prescription de sortie) ;
- ❖ La préparation ;
- ❖ La dispensation ;
- ❖ L'approvisionnement ;
- ❖ Le transport ;
- ❖ La détention et le stockage ;
- ❖ L'administration ;
- ❖ L'information du patient ;
- ❖ La surveillance du traitement.

Les étapes majeures identifiées, il est alors possible d'associer à chacune d'entre elles, un (des) acteur(s) et une (des) action(s) (Tableau 1.1).

Tableau 1.1 : Liste des étapes, acteurs et actions du CDM

ETAPES	ACTEUR(S)	ACTIONS(S)
Prescription	Médecin(s) & Pharmacien(s)	Traitement de l'ordonnance médicale
Préparation	Préparateur(s)	Préparation des doses, colis, etc.
Dispensation	Infirmière(s) & Pharmacien(s)	Gérer la prescription et préparer le traitement
Approvisionnement	Pharmacien(s)	Approvisionner les stocks en produits pharmaceutiques
Transport	Manutentionnaire(s)	Transférer médicaments
Détention et stockage	Pharmacien(s) & Préparateur(s)	Gestion des stocks
Administration	Infirmière(s)	Traiter le patient
Information du patient	Pharmacien(s)	Education thérapeutique
Surveillance du traitement	Infirmière(s)	Suivi et contrôle du patient

La Figure 1.1 donne une représentation graphique du circuit.

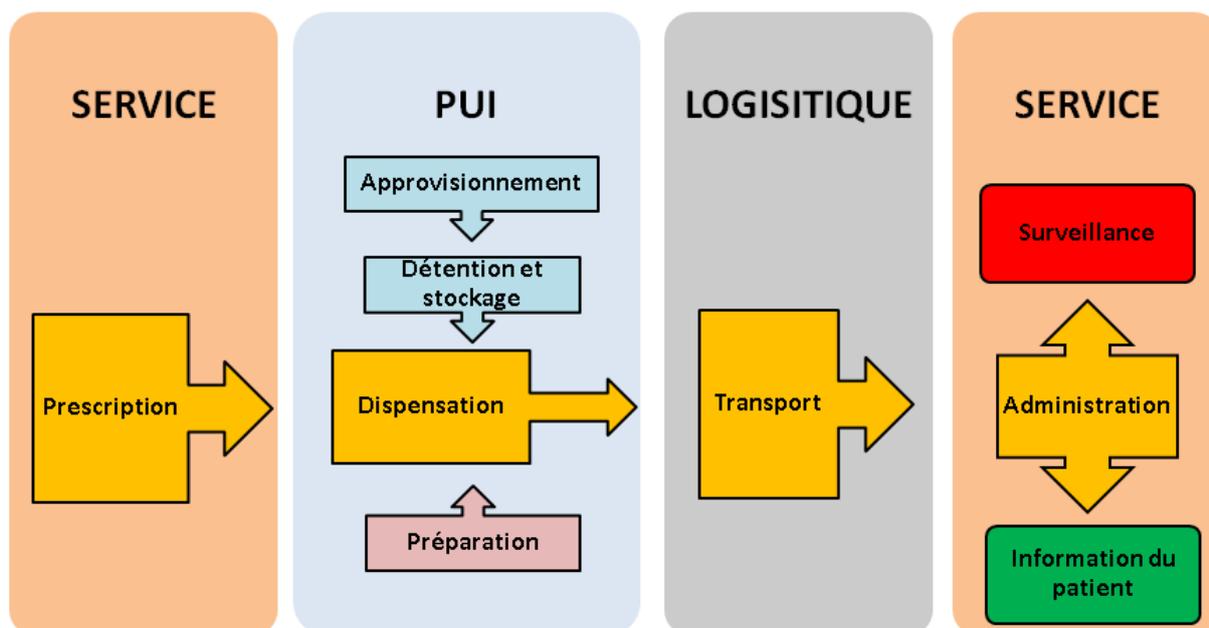


Figure 1.1 : Chronologie des étapes du CDM

Le CDM se concentre donc logiquement sur les services de soins et sur la PUI. Le pôle logistique bien que n'assurant qu'un rôle de transport, se retrouve quand même soumis à de nombreuses contraintes qui seront notamment présentées dans le cadre réglementaire entourant le circuit.

En complément de cette vision globale et rapide du CDM, il faut aussi prendre en compte la dimension très diversifiée des produits qu'il doit supporter. En effet, chaque type de produits se trouve être soumis à de fortes contraintes tant sur ses méthodes de stockage que de transport. Parmi les produits identifiables il y a :

- ❖ Les dispositifs médicaux, communément notés **DM** ;
- ❖ Les gaz médicaux ;
- ❖ Les stupéfiants ;
- ❖ Les antibiotiques ;
- ❖ Les préparations magistrales ;
- ❖ Les préparations hospitalières ;
- ❖ Les cytostatiques ;
- ❖ Les médicaments dérivés du sang (noté **MDS**) ;
- ❖ Les essais cliniques ;
- ❖ Les produits rétrocédables ;
- ❖ Les produits avec « Autorisation Temporaire d'Utilisation » (notée **ATU**) ;
- ❖ Et les médicaments normaux.

Clairement, la complexité déjà élevée du CDM ne s'en trouve que renforcée avec cette multiplication de produits et les contraintes réglementaires qui en découlent.

1.2.3 Un cadre réglementaire strict

Du fait du grand nombre de produits à risque manipulés et gérés, le cadre réglementaire entourant le CDM est l'un des plus stricts existants, afin de garantir une qualité et une sécurité irréprochables dans la prise en charge thérapeutique médicamenteuse des patients. Ce maillon essentiel sur lequel se repose l'hôpital est constitué entre autres des réglementations suivantes :

❖ Réglementation d'ordre générale :

- ✓ *Arrêté du 31 Mars 1999* : relatif à la prescription, à la dispensation et à l'administration des médicaments ;
- ✓ *Arrêté du 6 Avril 2011* : relatif au management de la qualité de la prise en charge thérapeutique médicamenteuse et aux médicaments dans les établissements de santé.

❖ Réglementation spécifique :

Tableau 1.2 : Lois et articles principaux régissant chaque étape du CDM

Etape	Texte / Article	Descriptif
Prescription	Art R. 5194 Code de santé publique (CSP)	Critère de validité d'une prescription
	Art R. 4127-8 CSP	Liberté de prescription du médecin
Préparation	Art L. 5121-1 CSP	Définition des différentes catégories de préparations.
	Art L. 5121-5 CSP	Bonne pratique de préparation.
Dispensation	Art L. 4241-1 à 18 CSP	Rôle et responsabilités des préparateurs en pharmacie.
	Art R. 4235-48 CSP	Définition de la dispensation.
Administration	Art R.4311-1 à 15 CSP	Rôle et missions des infirmières.
	Art R. 4312-29	<i>« L'infirmier ou l'infirmière applique et respecte la prescription médicale écrite, datée et signée par le médecin prescripteur, ainsi que les protocoles thérapeutiques et de soins d'urgence que celui-ci a déterminés... »</i>
Détention et stockage	Art R. 5126-1 à 115 CSP	Rôle et droits des PUI.
Information du patient	Art L. 1111-2 à 7 CSP	Droit d'information du patient.
	Art L.1142-4 CSP	Droit d'information lié aux dommages imputés à un traitement.
	Art 1161-1 à 6 CSP	Education thérapeutique du patient.

Cette liste non exhaustive des articles et lois régissant chaque étape du CDM permet de voir que chacune d'entre elles est parfaitement encadrée et remises régulièrement en question par de nouveaux amendements, ce qui n'empêche malheureusement pas des erreurs et problèmes à chaque niveau, comme il sera possible de le voir dans la [partie 1.3.4](#).

1.2.4 Les différents modes de délivrance existants

Maintenant que les grandes étapes du CDM et son cadre réglementaire ont été définis, s'intéresser aux différents types de dispensation qui sont en vigueur dans les hôpitaux est primordial étant donné que ceux-ci peuvent définir à eux seuls la structure que le CDM devra prendre. Que dit la législation à ce sujet? En se référant à l'article R. 4235-48 du Code de la santé publique, cité dans la « **Définition 1.3** » ci-dessus, il est constaté que la dispensation est définie comme un ensemble d'actions incorporant entre autres l'acte d'analyse des prescriptions. Cependant, cette tâche, dévolue au pharmacien, n'est malheureusement encore que trop peu appliquée, et il n'est donc pas possible de parler de « dispensation » dans chaque cas au sens où la législation l'entend. Le terme « distribution » sera alors usité, « délivrance » faisant déjà référence à une action définie, ce terme se prêtant bien à un contexte où le contrôle pharmaceutique des prescriptions n'a pas lieu, la PUI n'étant alors qu'un distributeur.

De plus, bien que structurantes, les trois grandes étapes composant le CDM (prescription, dispensation, administration), sont tributaires du contexte historique des établissements et des moyens humains, techniques et économiques qui y sont mis en œuvre.

Ainsi comme critères d'hétérogénéité qui apparaissent du fait de la grande diversité des organisations et fonctionnements existants, il y a :

- ❖ La taille et les missions de chaque hôpital ;
- ❖ Le type de séjour (court, moyen, long) impactant le rythme des prescriptions ;
- ❖ Le niveau de dialogue et d'interopérabilité entre les trois principales catégories d'acteurs: infirmières, médecins prescripteurs et équipe pharmaceutique ;
- ❖ L'organisation des flux physiques et des flux d'informations ;
- ❖ La configuration monosite ou multisites ;
- ❖ Les moyens humains disponibles ;
- ❖ La présence ou l'absence d'une plateforme logistique ;
- ❖ Les moyens techniques disponibles : matériels de rangement, de stockage, de transport, de protection, d'identification, de logiciels ou d'automates, etc.

Les différents types de dispensations/délivrances s'articuleront donc autour de cinq items :

- ❖ L'échelle de temps : Journalière / Pluri journalière ;
- ❖ Le traitement de la prescription : Papier / Informatique ;
- ❖ Le caractère nominatif de la prescription ;
- ❖ L'automatisation du circuit ;
- ❖ La présence d'une analyse pharmaceutique des prescriptions.

Pour identifier chaque type existant, intéressons-nous alors au Tableau 1.3 :

Tableau 1.3 : Critères sous-jacents à chaque type de dispensation / délivrance

	Type de gestion	DgM	DGM	DNMP	DJINA	DJINM
Critère						
Temporel	Journalier			Les deux sont possibles	X	X
	Pluri-journalier	X	X			
Prescription / Demande	Papier	X		X		
	Informatisée		X		X	X
Prescription nominative	Oui			X	X	
	Non	X	X			X
Analyse pharmaceutique	Oui			X	X	X
	Non	X	X			
Préparation automatisée	Oui				X	
	Non	X	X	X		X

Avec :

- ❖ DgM : Distribution globale Manuelle ;
- ❖ DGM : Dispensation Globalisée Manuelle ;
- ❖ DNMP : Dispensation Nominative Manuelle Partielle ;
- ❖ DJINA : Dispensation Journalière Individuelle Nominative Automatisée ;
- ❖ DJINM : Dispensation Journalière Individuelle Nominative Manuelle.

Détaillons maintenant avec plus de précision ces différentes gestions :

↳ La Distribution globale Manuelle :

Modèle traditionnel de fonctionnement hérité de l'époque où l'informatisation et la robotique étaient encore inexistantes dans le domaine hospitalier, ce mode de gestion a le défaut majeur de relayer la pharmacie au rang de simple grossiste répartiteur de médicaments pour la structure hospitalière. En effet, les différents services des hôpitaux transmettent leurs demandes sous forme papier, la pharmacie n'ayant plus alors qu'à préparer celles-ci, perdant ainsi son rôle majeur, le conseil et l'analyse pharmaceutique, et sans pouvoir contrôler le devenir des médicaments une fois livrés aux services.

Le fonctionnement est alors ici entièrement manuel. Mais, bien qu'ayant l'avantage de ne nécessiter aucun moyen ou matériel onéreux, il a le défaut d'être extrêmement gourmand en main d'œuvre et sous la menace permanente de l'apparition d'erreurs à tous ses niveaux de fonctionnement dû à la multiplicité et à la répétitivité des tâches à accomplir.

Cependant, ce mode de gestion globale ne devrait plus être en vigueur depuis 1999, la réglementation pharmaceutique obligeant en théorie les prescriptions à être nominatives. Néanmoins, certains services du fait de leur fonctionnement ne pourraient se passer de ce mode, ce qui est notamment le cas des services de réanimations, des urgences et des blocs opératoires.

Le cheminement de ce mode de gestion peut être résumé de la manière suivante :

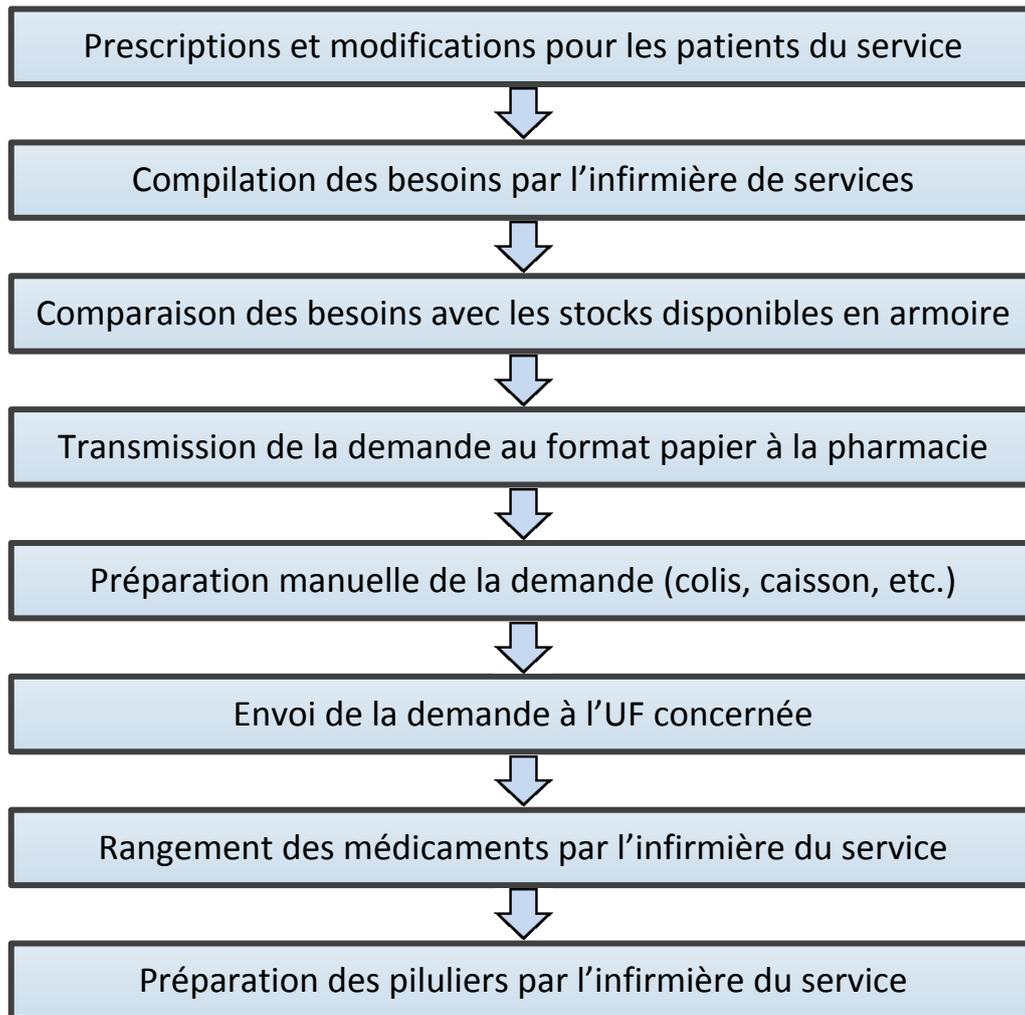


Figure 1.2 : Cheminement DgM

↳ La Dispensation Globalisée Manuelle:

Contrairement au cas précédent, les demandes de services sont ici informatisées. En général, les prescriptions individuelles de ces services sont elles aussi informatisées et permettent donc aux pharmaciens de compléter la préparation des demandes par une validation pharmaceutique de chaque prescription. Celle-ci n'a cependant pas une incidence directe sur les médicaments délivrés, la validation n'ayant ici qu'un rôle consultatif. Deux cas de figures sont à envisager. Pour le premier, les services réalisent eux-mêmes leurs demandes, c'est-à-dire que chaque prescription est compilée pour estimer les besoins du service avant de transmettre les besoins à la pharmacie. Dans le second cas, la pharmacie réalise cette compilation, situation plus sécurisée, celle-ci n'envoyant que le nécessaire contrairement au premier cas où les besoins peuvent être gonflés pour diverses raisons.

Ce mode nécessite plus de moyens pour la partie informatisation, mais simplifie une grande partie du travail au niveau des prescriptions en sécurisant le CDM par la même occasion.

Cette fois-ci le cheminement peut être interprété comme suit :

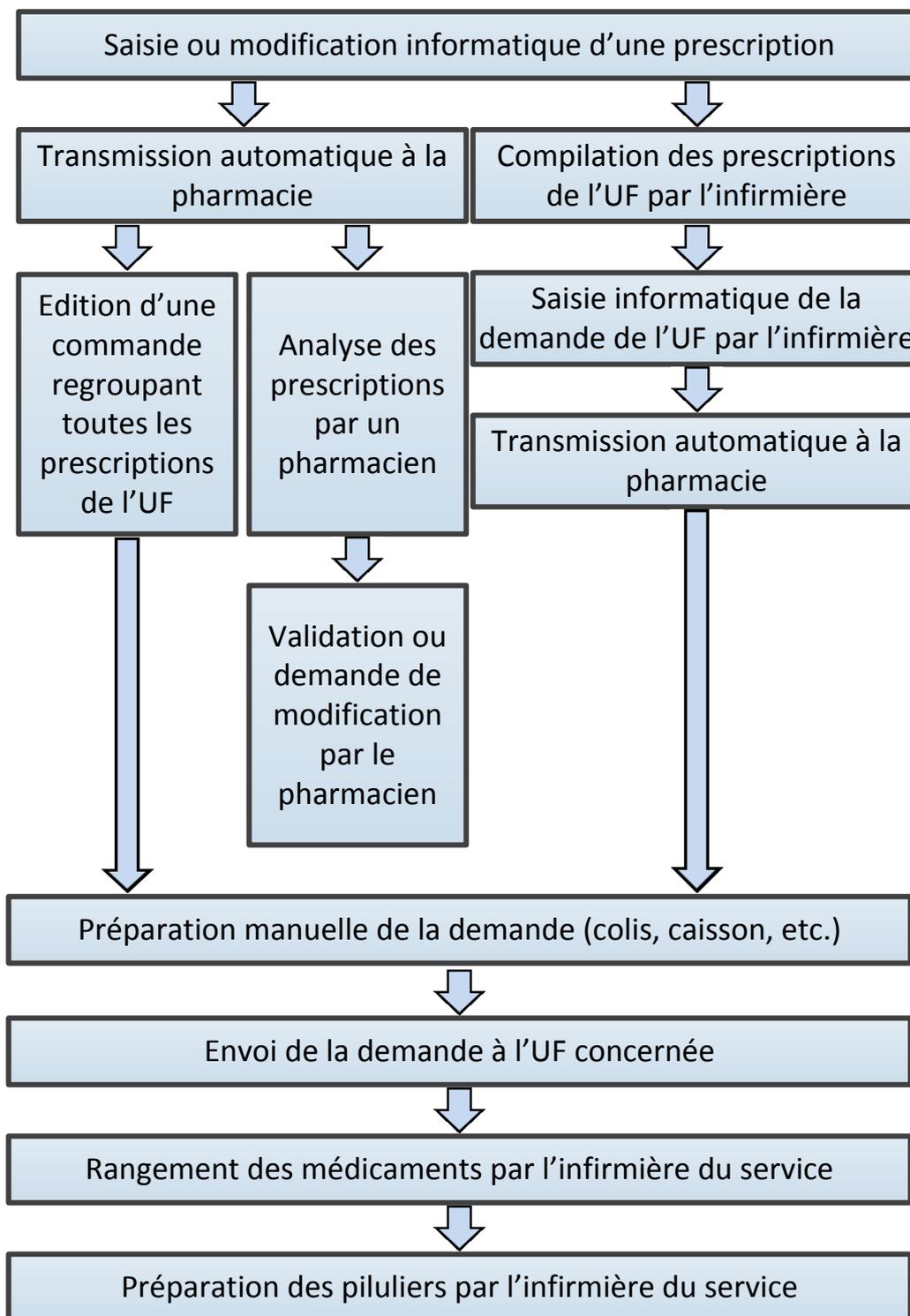


Figure 1.3 : Cheminement DGM

Cette méthode de dispensation est encore utilisée dans la majorité des services du CHU de Clermont-Ferrand.

↳ La Dispensation Nominative Manuelle Partielle :

Cette dispensation est un cas unique. Elle intervient en parallèle de la DgM ou DGM. En effet, certains médicaments étant soumis à un contrôle strict avant délivrance, la prescription doit donc être impérativement transmise au pharmacien et validée par ce dernier.

Le déroulement est ici :

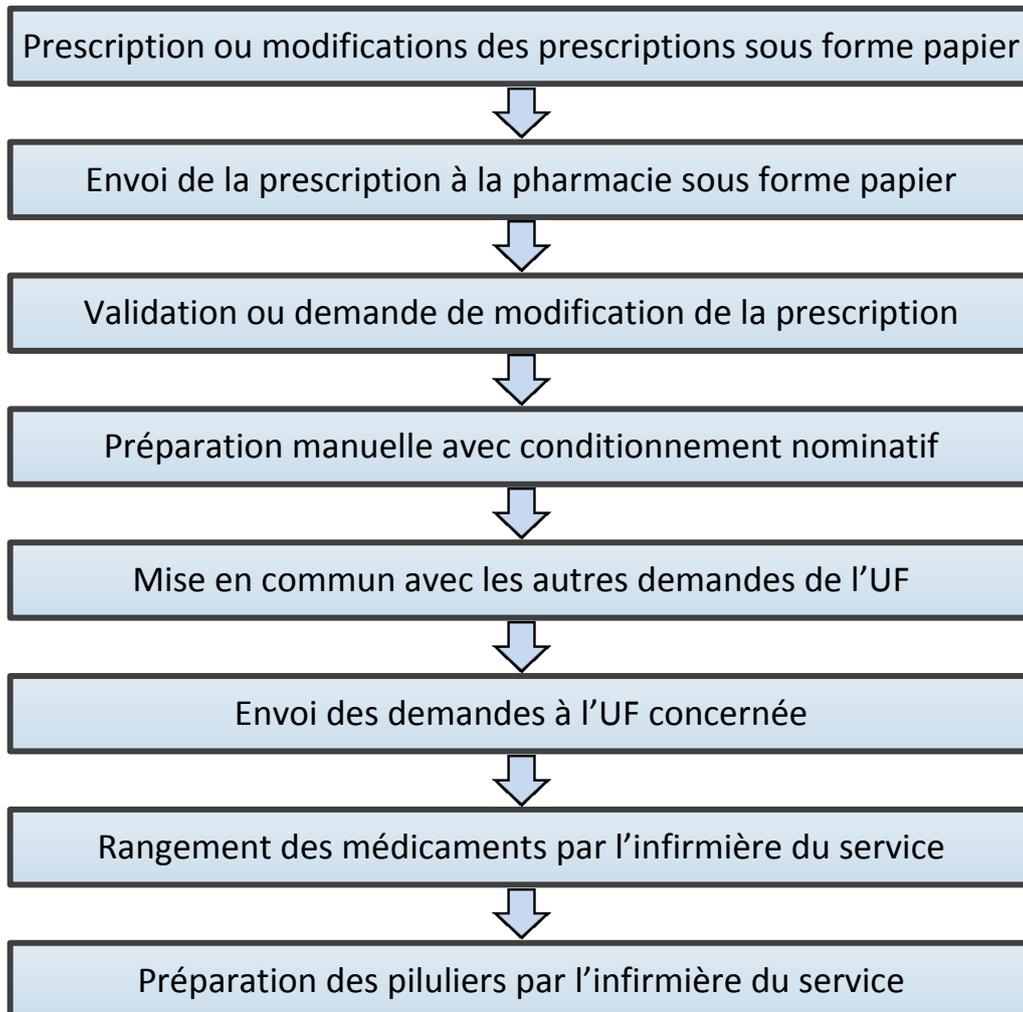


Figure 1.4 : Cheminement DNMP

↳ La Dispensation Journalière Individuelle Nominative Automatisée ou Manuelle :

Pour terminer, ce mode de dispensation consiste en la délivrance de doses nominatives pour une journée (traitement journalier ou trié par avance en matin/midi/soir). Le gros avantage de ce type de dispensation est son côté hautement sécuritaire. Cependant pour y parvenir, de forts investissements financiers et humains sont obligatoires. Il est notamment nécessaire d'avoir des prescriptions individuelles informatisées, des doses unitaires, des automates de distribution (pour le mode automatisé) et du personnel formé à un tel fonctionnement.

De nombreuses réflexions sont menées où l'ont été afin d'obtenir le meilleur mode de fonctionnement possible pour la DJIN. Des versions avec « antennes de pharmacie » dans chaque service ou par pôle ont donc vu le jour. La DJIN n'est cependant pas réalisable dans le cas des

services de réanimations, des urgences et dans les blocs opératoires, du fait de la grande volubilité des demandes et de leur rythme irrégulier.

A noter qu'il existe une variante de ce fonctionnement pour des services avec peu de changements dans les prescriptions tels que la psychiatrie par exemple. La dispensation se fait donc pour plusieurs jours (voire une semaine) et non plus de manière journalière. Les avantages et le fonctionnement restent cependant les mêmes. Le terme utilisé sera alors DIN.

Le déroulement de ce mode de gestion est le suivant :

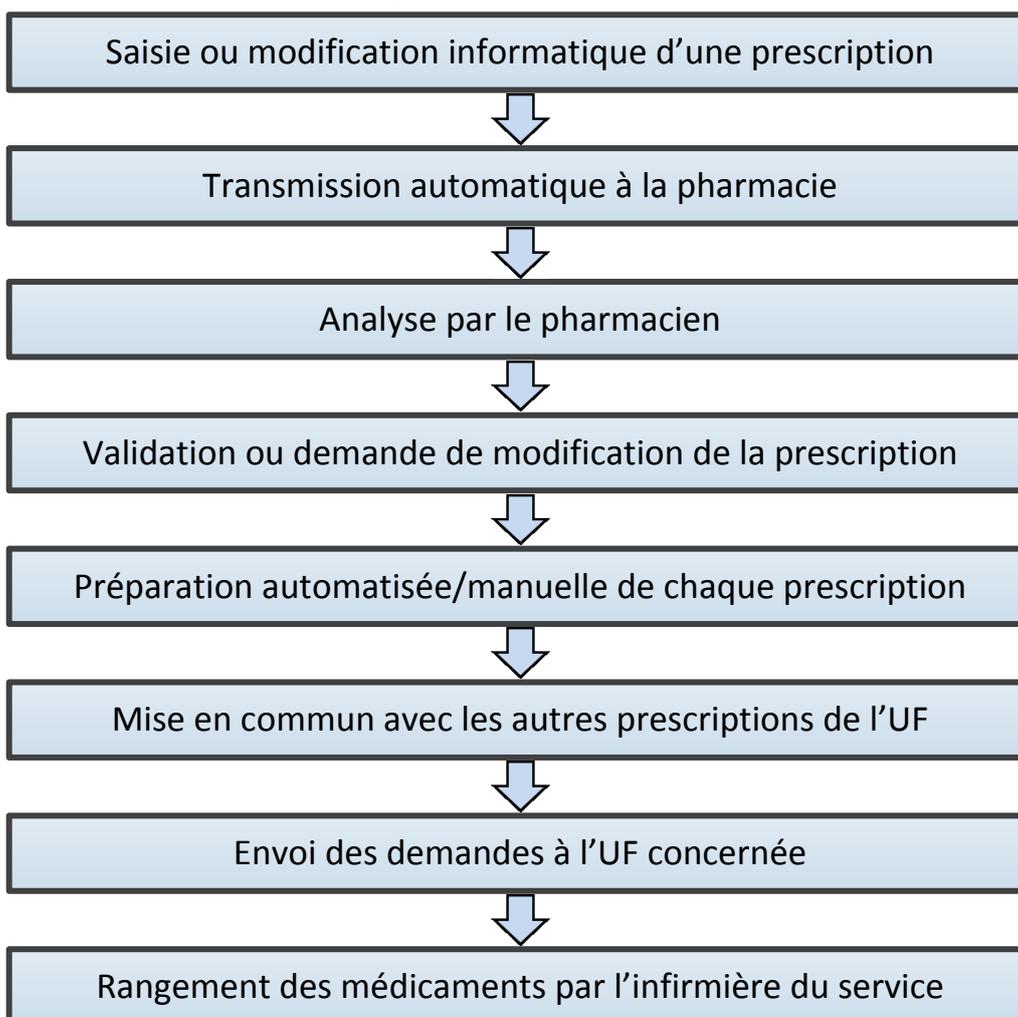


Figure 1.5 : Cheminement DJIN

↳ L'armoire automatisée : un parallèle pertinent à la DJIN ?

En plus de la dispensation et distribution sous toutes ses formes, une variante intéressante est apparue. Il s'agit des armoires « automatisées » ou « sécurisées ». Ces armoires informatisées ont l'énorme avantage de pouvoir s'interfacer facilement avec les logiciels de gestion des pharmacies et permettent donc de faire un suivi en temps réel de leur contenu. De plus, l'obligation de s'identifier pour pouvoir en retirer des produits sécurise énormément le circuit et permet aux infirmières d'être rassurées quant au produit pris, un affichage lumineux étant présent dans la plupart des cas pour indiquer où est le produit à retirer. Enfin, étant interfacée, ces armoires peuvent générer en automatique des demandes de réapprovisionnement à la pharmacie.

1.3 LE CDM : CONTRAINT ET SOUS PRESSION

1.3.1 Introduction

L'iatrogénie médicamenteuse (**Définition 1.4 et 1.5**), fléau moderne des établissements de santé, est aujourd'hui une priorité majeure et sa prévention est désormais synonyme de sécurisation du CDM. Ceci est d'autant plus vrai que, la population française vieillissante est de plus en plus sensible à ces problèmes. Ayant pour conséquences, entre autres, un rallongement des durées moyennes de séjours (**Dms**) et une augmentation de la mortalité, l'iatrogénie médicamenteuse est donc difficilement acceptable au niveau des coûts autant humains que financiers pour les établissements de santé, et des solutions doivent alors être impérativement trouvées pour la contrer.

Définition 1.4 : « L'iatrogénie médicamenteuse correspond à tous effets indésirables sur la santé résultant de la prise concomitante de plusieurs médicaments.² »

Définition 1.5 : « La notion d'iatrogénie médicamenteuse est utilisée lorsque la thérapeutique médicamenteuse induit des effets, réactions, évènements ou accidents indésirables, tant en raison des effets propres des médicaments concernés qu'à cause du contexte et des modalités d'utilisation. » [**SCHMITT 2006**]

Dans le but de pouvoir connaître l'incidence de ce type d'erreur et leur propension, des études ont été menées et celles-ci ont mis en évidence que ce problème était jusque-là sous-estimé en France. C'est notamment le cas de l' [**ENEIS 11**]. Réalisée sur un panel de 61 établissements de santé (noté **ES**), il apparaît que 32.9% des **EIG** (Evènement Indésirable Grave, **Définition 1.6**) sont d'origine médicamenteuse, que 51,2% d'entre eux sont évitables et que 54,5% ont provoqué une hospitalisation.

Définition 1.6 : « Il s'agit d'un effet indésirable létal, ou susceptible de mettre la vie en danger, ou entraînant une invalidité ou une incapacité importantes ou durables, ou provoquant ou prolongeant une hospitalisation, ou se manifestant par une anomalie ou une malformation congénitale. »³

Une étude de 2007, EMIR⁴, sur les hospitalisations liées à un évènement indésirable médicamenteux (noté **EIM**) menée sur 63 ES, pour le court séjour, a estimé le nombre annuel d'hospitalisations dues à des EIM à 143 915, soit un taux d'incidence de 3.60%, et le nombre total d'EIM à 1 480 885 par an.

Au niveau international, il est aussi constaté, du fait des nombreuses études dans divers pays, que la sécurisation de la prise en charge thérapeutique médicamenteuse est un enjeu mondial.

² <http://www.ars.poitou-charentes.sante.fr/latrogenie-medicamenteuse.90105.0.html>

³ Décret n° 99 du 29 janvier 2004

⁴ Effets indésirables des Médicaments : Incidence et Risque

Par exemple, une étude datant de 2002 réalisée au Etats-Unis, montre que dans 36 établissements, 19% des doses dispensées et administrées comportaient une erreur [DGOS 12].

Ainsi, depuis plusieurs années maintenant, les établissements de santé sont incités à mettre en œuvre une politique de prévention et de maîtrise des erreurs médicamenteuses, cela dans l'optique de garantir la sécurité de la prise en charge thérapeutique du patient (**Définition 1.7**).

Définition 1.7 : « La prise en charge médicamenteuse ou thérapeutique est un processus combinant des étapes pluridisciplinaires et interdépendantes visant un objectif commun: l'utilisation sécurisée, appropriée et efficiente du médicament chez le patient pris en charge par un établissement de santé (Arrêté du 6 avril 2011). »

Dans un premier temps, certaines contraintes que le CDM doit supporter sont présentées. Nous nous intéressons ensuite aux problèmes récurrents constatés sur le circuit avant de terminer par les nombreuses erreurs qui découlent de ces problèmes et de ces contraintes.

1.3.2 Les contraintes du CDM

Le CDM est soumis à des contraintes internes et externes avec lesquelles il doit composer s'il veut pérenniser. Touchant ce circuit à tous ses niveaux, ces contraintes sont diverses et variées, mais peuvent se regrouper en trois domaines majeurs : organisationnel, humain et matériel. Voyons plus en détails les trois principales contraintes de ces domaines.

1.3.2.1 La transversalité du CDM

Comme déjà énoncé, le CDM est l'un des procédés de soins les plus transversaux sur lesquels sont construits les établissements de santé. Celui-ci est au carrefour d'un nombre important d'acteurs et composé de quantités significatives et très diversifiées de produits à usage thérapeutique dont la complexité ne cesse d'augmenter, tant au niveau de la gestion des risques, des procédures administratives (ATU, médicament hors groupement homogène de séjour noté GHS), qu'au niveau des bénéfiques qu'ils fournissent.

Cette première contrainte, qui est aussi la plus importante, se situe donc à la croisée des domaines organisationnel et humain. Commençons par ce dernier :

↳ Le domaine humain :

En se basant sur le modèle de Glouberman et Mintzberg [GLOUBERMAN 2001], et en le focalisant sur le CDM, il est possible de classer en quatre grandes catégories les différents acteurs intervenants dans le circuit (Figure 1.6).

Cette classification s'est opérée comme suit. Elle dépend de la proximité de l'acteur avec le patient et le terrain, et du pouvoir de décision (économique, hiérarchique) de chacun. Plus il sera proche du patient/terrain, plus il sera « down », et plus il s'en éloignera plus il sera « up ». De même, plus il aura de pouvoir de décision, d'indépendance décisionnelle, plus il sera « out » et moins il en aura, plus il sera « in » (dans le sens « dans le système »).

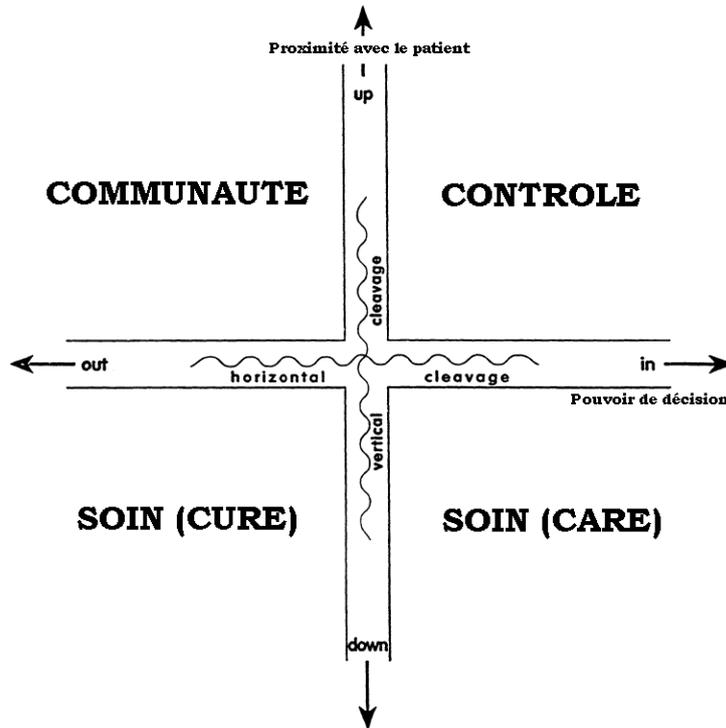


Figure 1.6 : Classification de Glouberman et Mintzberg

En bas à droite, il sera donc retrouvé le personnel infirmier ainsi que les préparateurs et autres magasiniers agissant directement sur le circuit et les patients (« down »), et qui le connaissent donc le mieux. Ils n'ont que très peu de pouvoir de décision (« in ») étant sous l'autorité hiérarchique.

En bas à gauche, se situe au contraire les médecins et pharmaciens avec un fort pouvoir de décision (« out ») et qui sont très proches des patients (« down »). En effet, ceux-ci sont autonomes dans leurs décisions et rendent donc peu de comptes sur la façon dont ils traitent les patients.

En haut à gauche, se trouve ce qui s'appelle la « communauté ». il s'agit des grands décisionnaires (« out ») très peu sur le terrain (« up »). Dans le cas du CDM, ceci représente la direction des différents services (logistique, pharmacie, unité de soins).

Finalement, le bloc en haut à droite symbolise l'administration standard, responsable du bon fonctionnement administratif de l'hôpital (« in »), mais sans connaissance du monde clinique (« up »).

↳ Le domaine organisationnel :

Touchant à la fois le côté structurel et hiérarchique de l'hôpital, la contrainte de transversalité organisationnelle repose ici sur deux grandes causes. La première concerne l'agencement spatial des établissements de santé. En effet, les locaux (monosites ou multisites) ne sont pas toujours en adéquation avec l'idée de circuit. Devoir naviguer entre différents stocks éparpillés sur plusieurs lieux ne fait que compliquer la tâche des services pharmacie et logistique. Il s'agit généralement d'un facteur limitant pour ces établissements quant au développement de nouveaux projets, ceux-ci ne pouvant pas ou difficilement optimiser leur fonctionnement. C'est le cas notamment du CHU de Clermont-Ferrand, où couplé au fait qu'il existe deux sites pour ce CHU séparés de 4 kilomètres, les locaux de l'hôpital Gabriel Montpied, exigü et datant des années 1970,

ne se prêtent plus aux exigences de fonctionnement d'une PUI, et imposent donc aux personnels et responsables de redoubler d'efforts pour adapter les flux, ceux-ci étant pourtant facilement exploitables dans des locaux adaptés.

La seconde cause majeure au problème de transversalité est la multiplication des échelons décisionnaires, bloquant ainsi souvent le dialogue interservices. Fort heureusement, les dernières lois et réformes (par exemple loi Hôpital, patients, santé et territoire noté **HPST**) tendent à vouloir rationaliser le fonctionnement des hôpitaux en rapprochant les activités médicales et administratives par exemple⁵. De plus, cette multiplicité d'échelons a aussi eu pour conséquence de ralentir le dialogue intra-service, mais aussi de le complexifier, en se demandant « qui fait quoi, et décide ? ».

La transversalité du CDM se fera donc sentir par l'apparition de clivages entre ces acteurs, le circuit pouvant se résumer en terme humain à la gestion de quatre organisations à la fois (Figure 1.6). Les clivages qui vont apparaître seront :

- ❖ Un clivage vertical, entre les acteurs décisionnaires et les exécutants, se traduisant par une difficulté de dialogue notamment.
- ❖ Un clivage horizontal, entre acteurs de terrain et administratifs, se traduisant par des incompréhensions quant à la dualité réelle/supposée du fonctionnement du circuit.

1.3.2.2 Le patient, un flux ?

Bien que la logique nous pousserait à répondre oui, la réalité ne nous permet pas de l'affirmer si aisément. En effet, les trajectoires des patients dans les établissements de santé sont très évolutives et variées, en étant tributaires de nombreux acteurs (accueil, consultation, hospitalisation, bloc opératoire, examen, soin infirmier, repas, etc.). Les composantes de ce flux sont donc interconnectées, dépendantes et indépendantes à la fois, s'entremêlant dans un problème d'ordonnement, jusqu'ici sans solution, bien plus complexe que ce qui existe dans le monde industriel. Malheureusement, dans les hôpitaux, la coordination de l'activité des services entre eux n'en est encore qu'à ses balbutiements. Même si la loi HPST et la nouvelle gouvernance hospitalière en pôles (un ensemble de services) ont permis de voir une amélioration en se basant sur un pilotage médico-économique de ces derniers et non plus médico-soignant, qui reste cependant encore majoritaire [**DAHAN 10**].

1.3.2.3 Une évolution technologique dissonante

Plus une structure hospitalière sera grande, plus elle aura besoin d'innover pour rester compétitive et en adéquation avec la société. Le problème d'un tel schéma de raisonnement, et cela d'autant plus dans le complexe actuel, est qu'une structure imposante ne pourra innover de manière équivalente et synchronisée entre ses différents pôles et services. En effet, les coûts pour arriver à de tels résultats ne sont plus supportables en l'état pour les établissements hospitaliers français, d'où l'apparition d'une dissonance technologique.

Cet état ne complique que davantage la gestion du CDM, les investissements étant très coûteux (robots, matériels de stockage, logiciel de gestion), et ne permettent donc pas de jouer sur tous les tableaux à la fois, pour obtenir un résultat global unifié et compétitif.

⁵ « Pôle d'activité : au service de la diffusion de la gestion », Gestions Hospitalières n°509 – octobre 2011, p.525-528.

Un exemple simple, peut être celui du CHU de Clermont-Ferrand. Le mode de gestion des stocks en plein/vide (voir [partie 1.4.3.2](#)) – ou gestion en mode Kanban avec des armoires de service dédiées – actuellement en déploiement ne peut l'être dans tous les services à la fois pour cause de manque de personnel et de matériel. Ceci rajoute alors un nouveau mode de gestion présent dans l'hôpital, et qui, bien qu'apportant des améliorations significatives, ne peut parfaitement fonctionner avec l'état actuel du système (ce qui revient à une problématique organisationnelle).

1.3.3 Les problèmes constatés

Abordons maintenant les problèmes qui ressortent des différentes études gouvernementales. Il apparaît que le déroulement du CDM est souvent perturbé par des lacunes/anomalies internes au circuit, où découlant d'une entité externe, que l'on nommera ici dysfonctionnement et qui peuvent se définir comme suit :

Définition 1.8 : « Un dysfonctionnement correspond à une anomalie ou une difficulté de fonctionnement dans l'organisation d'un processus de soin empêchant les professionnels de santé de réaliser de manière efficiente l'objectif thérapeutique, diagnostique ou préventif pour lequel le processus a été structuré. » (Arrêté du 6 Avril 2011)

Ce dysfonctionnement est souvent couplé à l'apparition d'un événement indésirable (noté **EI**) de nouveau défini par l'arrêté du 6 avril 2011 comme étant :

Définition 1.9 : « Une réaction nocive et non recherchée survenant chez l'homme utilisant ou exposé à un produit de santé sans préjuger d'un lien de cause à effet. »

Les études menées ont classé ces anomalies/lacunes, suivant les différentes grandes phases du CDM où les risques peuvent aboutir à un simple dysfonctionnement, ou à un EI sont :

- ❖ Au niveau du patient et des informations transmises : La capacité du patient à gérer son traitement est souvent problématique et l'éducation thérapeutique est des plus recommandées. Il y a donc un manque de dialogue pharmacien/patient.
- ❖ Au niveau des prescriptions : Les supports de prescriptions sont trop souvent hétérogènes d'où une forte discontinuité des informations transmises et des confusions qui, de plus, se conjuguent à une multiplicité des dénominations (DCI, appellation du service, etc.). Des anomalies surviennent aussi au niveau des prescriptions de sortie.
- ❖ Au niveau de la dispensation, distribution, et gestion des stocks dans l'unité : La mise en place de l'analyse pharmaceutique des prescriptions et son application, la gestion des stocks, des périmés, la fréquence des commandes souvent réactive et pas proactive (dépannage interservices, déplacements trop fréquents à la pharmacie, non standardisation des rangements dans les armoires de services), le transport des produits.
- ❖ Au niveau de la préparation des doses et le suivi du traitement et du patient : L'IDE préparant le pilulier/traitement n'est pas forcément celle qui va l'administrer, l'administration du traitement n'est pas suivie et validée en temps réel, la surveillance du patient n'est pas continue.

Un aspect en lien avec la validation des prescriptions est apparu durant ces études. Il s'agit de l'importance que ces validations ont quand ces prescriptions sont faites par des juniors ou par des faisant fonctions d'internes (communément appelés **FFI**). Ces interventions pharmaceutiques peuvent alors demander un temps de validation plus important notamment en début de semestre.

Nous allons aborder dans ce qui suit la situation sous un autre angle. Désormais nous nous concentrerons sur l'erreur induite par les différents problèmes vus ci-dessus. En effet, à moins d'identifier le problème avant tout incident, seule une étude approfondie de l'erreur permettra d'apporter une solution au problème qui en est la source.

1.3.4 Erreurs et Evènements Indésirables

Les EIM, précédemment définis, évènements défavorables au patient, ayant un caractère certain de gravité et menant dans la plupart des cas à un rallongement de la durée de séjour, à une incapacité ou bien à un risque vital, ont pour cause une erreur qui a été commise durant la prise en charge thérapeutique du patient. Cette erreur appelée erreur médicamenteuse (noté **ErM**), est définie par l'arrêté du 6 avril 2011 :

Définition 1.10 : « L'erreur médicamenteuse est l'omission ou la réalisation non intentionnelle d'un acte au cours du processus de soins impliquant un médicament, qui peut être à l'origine d'un risque ou d'un événement indésirable pour le patient. »

La question primordiale à ce niveau est donc :

« Où et pourquoi des erreurs ont-elles lieu ? »

Comme il sera possible de le voir dans ce qui suit, l'erreur peut trouver sa source dans de nombreuses situations, telle que la mauvaise conception du médicament, des informations erronées qui lui sont associées (confusion de dénomination, conditionnement inadapté, problème d'étiquetage ou de notice d'information, etc.), mais encore dans l'organisation du circuit de prise en charge thérapeutique (organisation du circuit du médicament, facteurs humains, facteurs environnementaux, pratiques professionnelles, etc.).

Effectivement il est constaté que, dans tout système complexe (centrale nucléaire, transport aérien, hôpital...), la réalité de fonctionnement n'est jamais aussi simple que ce qu'il a pu être décrit lors de la conception. Une analyse détaillée permet de constater que ces systèmes fonctionnent en partie autrement, qu'ils atteignent les objectifs fixés et réalisent leurs performances pour d'autres raisons que celles émises par les concepteurs.

Dès lors les acteurs doivent trouver des réponses aux difficultés qui se présentent, en exploitant des possibilités d'actions non prévues ou reconnues, variant d'une situation à une autre. Ce constat s'accroît de par la part croissante d'interventions humaines que le système subit et notamment dans le cas de l'hôpital et de la prise en charge thérapeutique des patients. De surcroît, les performances réalisées chaque jour sont parfois soumises à des incidents.

En outre, toute « modification du système » (nouvel équipement, nouveau logiciel, nouvelle réglementation, nouveau médicament, etc.) peut détruire des barrières de sécurité importantes (mises en place par les concepteurs ou tout simplement apparues au fil du fonctionnement sous l'impulsion des acteurs), et le dégrader de façon incompréhensible (si cette situation n'était pas envisagée), impliquant donc le devoir de redoubler de vigilance.

Ces « modifications du système » plus communément appelées « états limites » (du système et/ou de l'organisation) sont sources d'une partie des erreurs médicamenteuses commises. En effet, l'apparition de l'un d'entre eux a pour conséquence d'éloigner le système de sa routine de fonctionnement et donc d'accentuer les risques de voir apparaître des erreurs.

Du côté des erreurs, malgré les nombreuses études menées, il est difficile de donner avec précision la part d'erreurs engendrées par chaque étape du circuit. Afin de pouvoir cerner de manière la plus efficiente possible la répartition des erreurs médicamenteuses dans le CDM, nous allons nous baser sur deux études. La première est [CASTOT 10], document produit par l'Afssaps (désormais connue comme Agence Nationale de Sécurité du Médicament notée ANSM) concernant la mise en place du « Guichet erreurs médicamenteuses » que nous noterons GEM. La seconde concerne l'étude MERVEIL, dont les résultats présentés par Etienne Schmitt vont permettre de dresser un bilan de la répartition des ErM.

1.3.4.1 Le guichet des erreurs médicamenteuses de l'Afssaps :

Ce premier rapport de l'Afssaps permet d'obtenir une répartition des ErM dans le CDM et fournit une première identification des causes des erreurs. Le Tableau 1.4 suivant donne le nombre d'erreurs relevées durant les quatre années de l'étude pour chaque étape du circuit.

Tableau 1.4 : Nombre d'erreurs à chaque étape déclarées au GEM de l'Afssaps

	2005	2006	2007	2008	TOTAL
Erreur de prescription	8	9	15	19	51
Erreur de transcription	1	1	1	3	6
Erreur de lecture d'ordonnance	0	0	4	3	7
Erreur de délivrance	8	23	31	47	109
Erreur de préparation	14	35	7	31	87
Erreur d'administration	25	91	101	206	423
Erreur de suivi thérapeutique	4	3	0	0	7
Autres	0	8	27	7	42
Non spécifié	1	0	0	8	9
TOTAL	61	170	186	324	741

Sur la Figure 1.7, il apparaît donc que la majorité des erreurs soient de type « administration » et qu'elles aient lieu au plus près du patient. Les diagrammes en Annexe 1 complètent ces résultats en fournissant une description plus détaillée de chaque type d'erreur et les causes qui leurs sont liées.

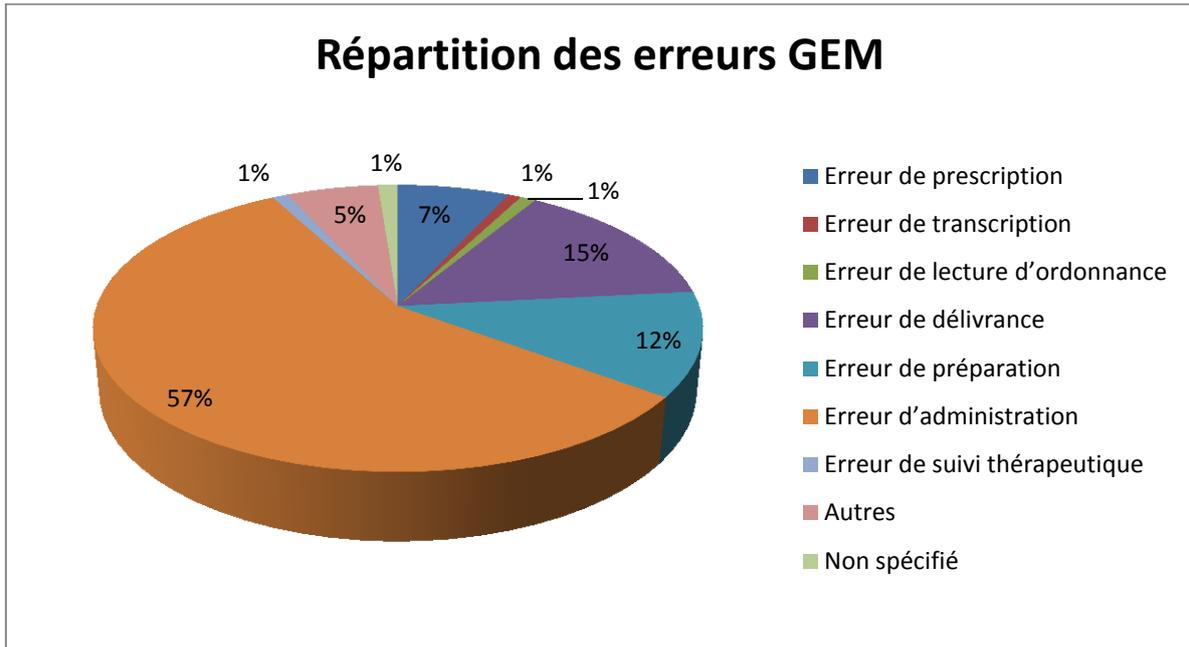


Figure 1.7 : Répartition des erreurs du GEM

L'ordre d'incidence des erreurs est donné par la Figure 1.8 :

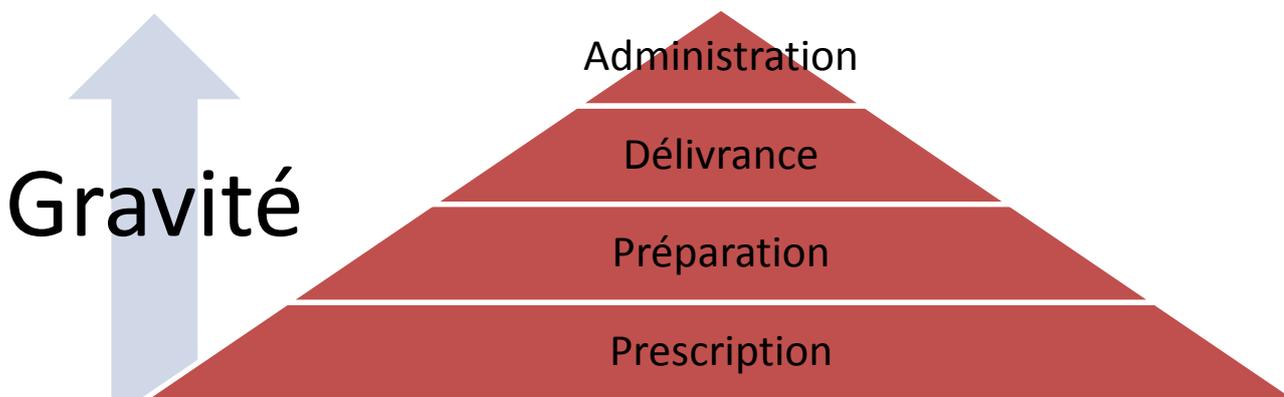


Figure 1.8 : Les 4 étapes qui ont généré le plus d'erreurs pour GEM

1.3.4.2 L'étude MERVEIL :

Cette étude menée sur la base de la méthode REMED (Revue des Erreurs liées aux Médicaments Et Dispositifs médicaux associés), pour signaler les erreurs médicamenteuses, permet tout comme la précédente d'identifier les étapes les plus émettrices d'ErM. Cependant cette étude se limite à un panel de 295 ErM significatives de par leur proportion et leurs causes. De surcroît, MERVEIL s'attarde sur l'aspect classification des erreurs et leurs taux d'interception à chaque étape (voir **Annexe 2**). La Figure 1.9 donne la répartition des ErM de cette étude :

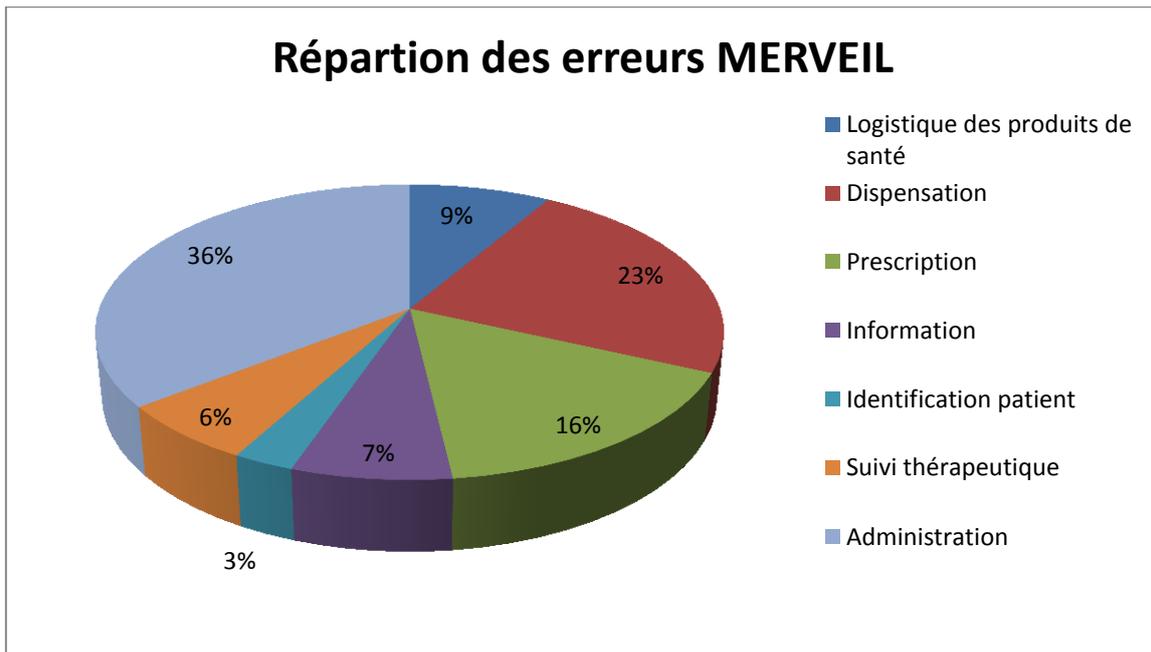


Figure 1.9 : Répartition des erreurs dans l'étude MERVEIL

Le découpage des étapes du CDM étant moins fin dans cette étude, et le nombre d'ErM moins élevé, des variations apparaissent par rapport aux résultats du document de l'Afssaps. Ainsi l'ordre d'incidence des erreurs est donné par la Figure 1.10 :

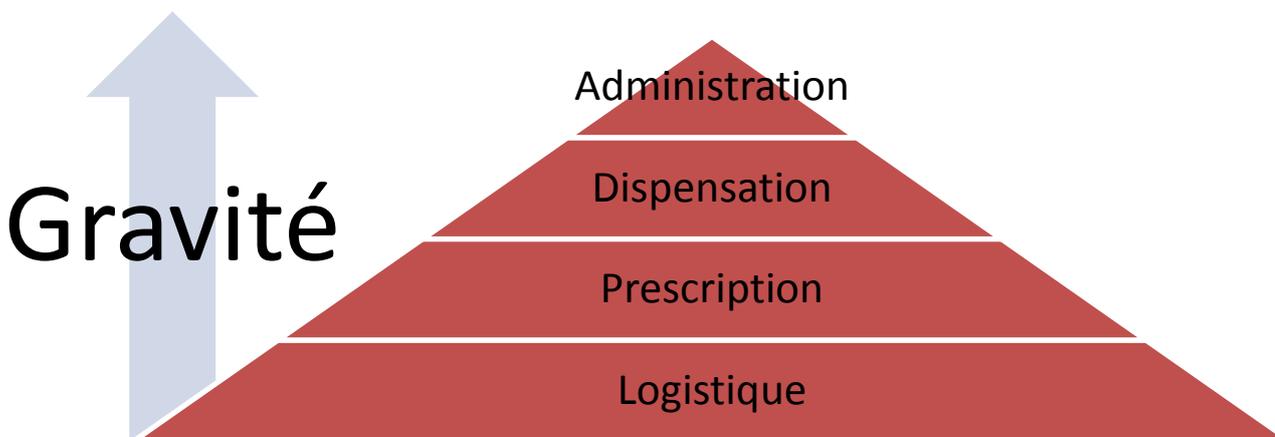


Figure 1.10 : Les 4 étapes générant le plus d'erreurs pour l'étude MERVEIL

De ces deux études, il apparaît clairement que la plus grosse source d'erreurs est l'étape d'administration. Cela se comprend parfaitement, du fait du grand nombre de médicaments à manipuler, du grand nombre de personnes qui peuvent intervenir sur une même action et aussi du fait que l'administration est en bout de chaîne. De plus, une erreur non identifiée précédemment se répercutera nécessairement à ce niveau du circuit.

En réalisant une moyenne des deux études, à titre purement indicatif, la part de chaque étape du circuit dans les erreurs médicamenteuses est de :

Tableau 1.5 : Part moyenne des erreurs issues des grandes étapes du CDM

Etape	Part d'erreurs
Prescription	11.5%
Dispensation	25%
Transport/Logistique	9% (inconnue dans GME)
Administration	46.5%
Suivi thérapeutique	3.5%

Afin d'illustrer le fait que ces données restent indicatives et ne sont pas généralisables, il est possible de citer deux autres études menées. La première datant de 1995 [BATES 95] et la seconde plus récente [BERNHEIM 05]. Chacune d'elle mets en avant des répartitions encore différentes :

Tableau 1.6 : Répartition des causes de l'iatrogénie médicamenteuse

Etape	Part d'erreurs [BATES 95]	Part d'erreurs [BERNHEIM 05]
Prescription	56%	37%
Transcription	6%	18%
Délivrance	4%	22%
Administration	34%	23%

L'explication de telles variations n'est pas unique et se justifie de diverses manières. Tout d'abord, l'époque et les moyens utilisés pour mener ces études varient grandement, tout comme ce qui est intégré dans l'étape « prescription » par exemple. De même le fonctionnement de l'établissement, ainsi que le pays d'origine peuvent jouer sur les résultats obtenus. Ces différents paramètres qui reflètent la réalité du terrain ne permettent donc pas d'identifier avec certitude la part de responsabilités de chaque étape dans les erreurs mais permet au moins de situer les zones les plus à risques : prescription et administration.

Maintenant que l'influence de chaque étape a été plus ou moins mise en évidence, essayons de voir quelles sont les origines de ces erreurs à chaque grande étape.

1.3.4.3 Les erreurs au niveau de la prescription :

D'après le rapport de Juillet 2010 de l'Afssaps [CASTOT 10], 8% des erreurs médicamenteuses relevées (en 2009) sont des erreurs de prescription, et 6.9% pour la période 2005-2008. Les différentes études ont permis d'identifier non seulement les causes mais aussi les raisons sous-jacentes à ces erreurs en lien avec les logiciels de prescription informatisée, les erreurs de retranscriptions, etc. De surcroît, des erreurs « techniques » découlant de pratiques professionnelles à risque (étant donné les conséquences) sont aussi observées. Il y a notamment :

- ❖ Un laxisme dans la communication médecin-patient ;
- ❖ La réalisation d'avis médicaux spécialisés par téléphone ;
- ❖ Les erreurs dans le choix des traitements ;
- ❖ Le manque de prévention ou de surveillance ;
- ❖ Une carence au niveau des échanges d'informations entre professionnels.

En plus de ces erreurs techniques et de ces pratiques, se rajoutent des facteurs de risques « environnementaux », dans le sens où ils font partie du cadre de travail entourant la prescription. Il est constaté des :

- ❖ facteurs liés aux tâches à accomplir : absence de protocoles, planification des tâches non adaptée, retard de prestation des examens, défaut d'accessibilité de l'information, protocoles non adaptés, soins ne relevant pas du champ d'expertise du service, etc.

Exemple : La prise en compte des traitements personnels :

Délicats à gérer, il est primordial de les prendre en compte dès l'admission du patient, sous peine de voir les conséquences dramatiques d'un arrêt brutal, d'une double prise, ou d'une interaction médicamenteuse.

Cette difficulté se fait donc ressentir autant du côté patient que soignant. Le premier car il doit modifier ses habitudes de traitements et le second car il doit s'y adapter. De plus, au-delà de la prescription, l'administration de ce traitement pose aussi problème : faut-il laisser les médicaments au lit du patient, faut-il les stocker dans les armoires ou les chariots de services, etc. Les habitudes variant d'un service à l'autre, il est donc difficile d'en faire ressortir une tendance.

- ❖ facteurs individuels : les connaissances théoriques des professionnels, les défauts d'aptitude des professionnels, l'insuffisance d'échange d'informations entre les professionnels et la famille, etc.

Exemple 1: Des pratiques de prescriptions hétérogènes :

Celles-ci peuvent varier entre différents hôpitaux pour des services équivalents, chacun d'entre eux possédant sa propre culture, du fait des médecins y travaillant. Cette culture se caractérise notamment par des méthodes de traitements et des médicaments utilisés différents. Cela se voit aussi par la définition de protocoles dans certains services (chimiothérapie, etc.) afin de standardiser les démarches.

Exemple 2: Une formation à la prescription jugée insuffisante :

En effet celle-ci se fait généralement « sur le tas » pendant l'externat et l'internat, ou les médecins en formation intègrent progressivement les pratiques et les outils de services, les pharmaciens et le personnel soignant participant de ce fait à leur formation par le biais de conseils et de validations pharmaceutiques pour les premiers.

La critique pouvant être adressée à ce mode d'apprentissage est que cette formation « pratique » ne se fait en parallèle d'aucune formation théorique. Cependant la grande diversité des outils de prescription (plus ou moins performants) et l'arsenal thérapeutique médicamenteux ayant explosé durant cette décennie n'aident pas non plus à rendre cette formation « sur le tas » aisée.

- ❖ facteurs concernant la capacité de travail : charge de travail importante, fourniture ou équipement non disponible ou inadapté, etc.

Exemple : La diversité du parc logiciel pour faire de la prescription.

- ❖ facteurs liés à l'équipe : défaut de communication interne, supervision inadaptée des médecins ou des autres personnels, etc.
- ❖ facteurs concernant l'organisation : défaut de coordination entre les services, gestion du personnel ou des ressources humaines inadéquate, etc.
- ❖ facteurs liés au contexte institutionnel : ressources sanitaires insuffisantes ou défectueuses, absence de culture de signalement des situations dangereuses, etc.

- ❖ facteurs liés aux produits : diversité et multiplication des dénominations pour une même molécule, changement récurrent, allergie du patient, etc.

Exemple : Des médicaments qui changent trop souvent :

Difficulté signalée tant par les prescripteurs que par les soignants, celle-ci porte sur la multiplicité des génériques et sur la faible durée des marchés qui induisent de fréquentes variations d'autant plus gênantes que le nom des spécialités pouvant être prescrites notamment dans le domaine des génériques peuvent être très proches les unes des autres.

En résumé, les risques majeurs identifiés lors de l'étape de prescription sont :

- ❖ Les prescriptions faites oralement (sauf en cas de force majeure) ;
- ❖ L'absence d'identification du prescripteur ;
- ❖ La non-identification ou une mauvaise identification du patient ;
- ❖ La mauvaise lisibilité de la prescription ;
- ❖ Des données erronées dans les prescriptions informatiques ;
- ❖ L'usage d'abréviations ;
- ❖ L'imprécision, des erreurs sur un libellé : nom, forme, dosage, voie d'administration, etc. ;
- ❖ Le choix d'un médicament non adéquat ;
- ❖ Le non-respect d'une contre-indication ;
- ❖ L'existence d'interactions.

1.3.4.4 Les erreurs au niveau de la dispensation :

Le rapport de juillet 2008 de l'Agence Nationale d'Appui à la Performance (notée **ANAP**) [**LABORIE 08**] ayant porté sur une expertise de l'organisation du CDM dans les hôpitaux et cliniques, met en évidence que 22% à 25% des erreurs dans ce circuit proviennent d'une erreur de dispensation. En comparaison avec ce chiffre, le rapport de l'Afssaps [**CASTOT 10**], indique que 12% et 12.6% des erreurs médicamenteuses (entre 2005 et 2008) représentent respectivement des erreurs de délivrance et des erreurs de préparation (soit une erreur de dispensation supposée de 24.6%). Quant au rapport de juin 2009 [**CASTOT 09**], ces mêmes pourcentages sont respectivement de 14.7% et 11.7% (erreur de dispensation supposée de 26.4%). Globalement cette étape est donc en cause dans environ 20 à 25% des ErM, ce qui en fait la seconde cause génératrice d'erreurs.

Durant cette étape de dispensation, les erreurs pouvant être commises sont liées à l'analyse de l'ordonnance, la communication d'information, la préparation de formes galéniques (**Définition 1.11**), et la délivrance des médicaments aux US ou aux patients lors de la rétrocession. Ces erreurs sont donc de type :

- ❖ Erreur de dosage ;
- ❖ Erreur d'omission ;
- ❖ Erreur de médicament (nom similaire, présentation, etc.) ;
- ❖ Erreur de pratique (lecture difficile ou trop rapide de la prescription, distractions, interruption dans le travail, etc.) ;
- ❖ Erreur de forme galénique ;
- ❖ Erreur de substitution.

Définition 1.11 : « Forme galénique : forme physique (comprimé, gélule, ampoule, etc.) permettant la dispensation du médicament et la libération de son principe actif. »

1.3.4.5 Les erreurs au niveau du transport :

Bien qu'étant en théorie l'une des étapes du CDM les plus simples, elle se révèle compliquée étant donné la nature des produits sur lesquelles elle doit agir. Cependant, bien que soumise à de nombreux problèmes (mauvaise manipulation, détérioration des produits, etc.), cette étape n'implique que peu d'erreurs en lien direct avec le patient. La principale est une erreur de livraison en termes de lieu ou d'heure. Dans les deux cas, cela se traduira par un retard de prise du traitement pour les patients, ce qui parfois peut être très problématique.

1.3.4.6 Les erreurs au niveau de l'administration :

Etant en bout de chaîne du CDM, cette étape est tributaire de toutes les erreurs et problèmes que le circuit a pu subir jusqu'ici sans que cela ne soit détecté. L'infirmier est alors le dernier rempart contre cela et se doit d'être extrêmement vigilant. En effet, en plus des anomalies héritées du système, cette étape présente elle-même un certain nombre d'erreurs à ne pas commettre. Toujours selon les rapports de l'Afssaps [[CASTOT 09](#)] et [[CASTOT 10](#)], ces erreurs d'administration représentent respectivement pour les années 2005 à 2008 et l'année 2010, 57.1% et 58.7% des erreurs médicamenteuses commises. Cependant, dans la majorité des cas, elles proviennent de confusion ou de risques de confusions des produits à administrer. Les autres causes principales de ces ErM sont liées à un défaut d'information, un défaut de conditionnement, une mauvaise dénomination, ou des défauts de qualité. En outre, près du tiers des cas s'avère être des erreurs de pratique impliquant le professionnel de santé, dont l'essentiel correspond à des erreurs d'inattention, parfois liées à des circonstances particulières (exemple : contexte d'urgence), ou des erreurs d'utilisation par le patient, principalement et par ordre d'importance :

- ❖ Erreur de sélection du produit ;
- ❖ Erreur de lecture ou d'interprétation de l'étiquetage ;
- ❖ Erreur de lecture ou d'interprétation de l'ordonnance ;
- ❖ Mauvaise programmation des dispositifs d'administration ;
- ❖ Erreur de patient ;
- ❖ Erreur de préparation du pilulier ;
- ❖ Erreur de manipulation.

Au final, comme dit précédemment, ces ErM sont sources d'évènements indésirables pour le patient et le personnel. Cependant, toute ErM n'est pas cause d'un EI, et tout EI n'est pas de même gravité comme il sera possible de la voir dans ce qui suit.

C'est pourquoi l'erreur médicamenteuse peut être abordée sous trois angles. Elle peut être « avérée » ou « potentielle » lorsqu'elle s'est déjà produite et « latente » lorsque que celle-ci n'a jamais eu lieu mais est quand même identifiée. « Potentielle » traduit le fait que celle-ci n'a pas eu lieu, car interceptée avant l'acte sur le patient, et ne génère donc pas un EI.

Au final, seule une analyse *a posteriori* de l'erreur permettra de la caractériser de manière efficace et de la qualifier de par sa nature, son type, et la gravité de ses conséquences cliniques pour le patient.

1.3.5 Classification des EIM et étude ENEIS

Les nombreuses études sur les erreurs médicamenteuses ont permis de classer les conséquences de celles-ci suivant trois grandes catégories : les évitables, les inévitables et celles ne devant jamais arriver.

Les EIM évitables (**Définition 1.12**) sont de nouveau définies par l'arrêté du 6 avril 2011 :

Définition 1.12 : *Événement indésirable qui ne serait pas survenu si les soins avaient été conformes à la prise en charge considérée comme satisfaisante au moment de sa survenue.*

Il est important de noter que l'appréciation du caractère évitable d'un EIM est fondée sur un questionnement qui doit avoir lieu en amont. Il est nécessaire de se poser des questions telles que : Quel est le rapport bénéfice/risque du soin associé à l'erreur, le degré de déviation du soin par rapport à ce qui est défini (protocole, etc.), etc.

Les secondes, EIM inévitables, (**Définition 1.13**) sont définies par l'article R. 5121-152⁶(modifié par Décret n°2012-1244 du 8 novembre 2012 - art. 5) :

Définition 1.13 : *Réaction nocive et non voulue à un médicament, se produisant aux posologies normalement utilisées chez l'homme pour la prophylaxie, le diagnostic ou le traitement d'une maladie ou pour la restauration, la correction ou la modification d'une fonction physiologique ou résultant d'un mésusage du médicament ou produit.*

Pour cette seconde catégorie, l'exemple le plus parlant est une allergie non signalée et ne pouvant donc être connue par le personnel médical pour que celui-ci adapte le traitement. Cette seconde définition introduit néanmoins une nouvelle notion qui a son importance : *le mésusage*. Définit dans le même article que l'évènement inévitable, celui-ci se caractérise par :

Définition 1.14 : *Une utilisation intentionnelle et inappropriée d'un médicament ou d'un produit en rapport avec la dose autorisée ou prescrite, la voie d'administration, les indications, ou non conforme aux termes de l'autorisation de mise sur le marché ou de l'enregistrement ainsi qu'aux recommandations de bonnes pratiques.*

Il y a donc, dans ce cas génération intentionnelle d'évènement indésirable. Cependant, il est important de noter que l'on cherche parfois à utiliser des produits de manière « inappropriée » afin d'obtenir un effet précis. On parlera alors d' « interaction médicamenteuse contrôlée »

⁶ Code de la santé publique,

http://www.legifrance.gouv.fr/affichCodeArticle.do;jsessionid=AEB4EAC9086E3732E0DD8A4F41A6084D.tpdjo05v_3?idArticle=LEGIARTI000026596742&cidTexte=LEGITEXT000006072665&dateTexte=20121205

La dernière des catégories d'EIM concerne les « *Never events* ». Du fait de la démarche entreprise par la « National Health Service » en Grande-Bretagne et l'AFSSAPS, ces événements correspondent à des fautes très lourdes (évitables) qui impactent fortement la santé du patient. Ces deux entités ont dressé une liste des « *Never events* » les plus communs et les causes qui y sont liées. Parmi celles-ci, on retrouve :

- ❖ Erreur lors de l'administration du chlorure de potassium injectable ;
- ❖ Erreur d'administration d'insuline ;
- ❖ Erreur lors de la prise en charge de patients atteints de fibrillation auriculaire avec des médicaments anticoagulants ;
- ❖ Erreur d'administration des anticancéreux notamment en pédiatrie ;
- ❖ Erreur de préparation de spécialités injectables pour lesquelles les modes de préparation sont à risque ;
- ❖ Erreur lors de l'administration ou l'utilisation de petits conditionnements unidose en matière plastique (exemple : unidose de sérum physiologique, solution antiseptique...) notamment à la maternité ou en pédiatrie ;
- ❖ Erreur de programmation des dispositifs d'administration (pompes à perfusion, seringues électriques...)
- ❖ Erreur d'administration de produit par injection intrathécale au lieu de la voie intraveineuse ;
- ❖ Erreur d'administration par injection parentérale au lieu de la voie orale ou entérale ;
- ❖ Erreur du rythme d'administration du méthotrexate par voie orale (hors cancérologie) ;
- ❖ Erreur d'administration de spécialités utilisées en anesthésie réanimation au bloc opératoire ;
- ❖ Erreur d'administration de gaz à usage médical.

Visant directement les EI, l'étude ENEIS, enquête nationale des événements indésirables graves associés aux soins, datant de Septembre 2011 [ENEIS 11], permet de dresser un tableau de la situation et de l'évolution de ces événements dans les établissements (71 en 2004, 81 en 2009) durant la période 2004/2009. Ce rapport se concentre donc sur les événements dans leur globalité mais permet ainsi d'obtenir des chiffres concernant les erreurs médicamenteuses.

Le Tableau 1.7 met en lumière le fait que presque 78% des EIG ont lieu au plus près du patient, c'est-à-dire dans sa chambre. En termes de fréquence, le nombre d'EIG diminue sur ces cinq années, ceci étant dû notamment à la politique menée pour combattre ces erreurs.

Tableau 1.7 : Localisation des EIG

Lieu de survenue	2004			2009		
	Nombre total	Nombre	%	Nombre total	Nombre	%
Dans l'établissement	249			209		
Chambre du patient		193	77.51		156	74.64
Bloc opératoire		17	6.82		21	10.04
Unité de réanimation		29	11.64		20	9.56
Autre localisation		8	4.03		12	5.76

Les Tableaux 1.8 et 1.9 permettent de voir qu'environ un quart à un cinquième des EIG sont dues à une erreur médicamenteuse. La place de celle-ci dans les EIG n'est donc pas négligeable. Ces

tableaux mettent aussi en avant le fait que parmi ces EIG identifiés, environ un tiers de ceux-ci sont évitables et parmi elles, 33% (2004) et 41% (2009) sont dues à des médicaments.

Tableau 1.8 : Répartition des EIG par année par type d'incident

EIG lié à	EIG Total				EIG Evitables			
	2004		2009		2004		2009	
	Nombre	%	Nombre	%	Nombre	%	Nombre	%
Une procédure	191	74.9	159	74.3	66	69.5	58	66.7
Un produit de santé	71	27.8	83	38.8	30	31.6	41	47.1
Dont médicament	52	20.4	56	26.2	22	23.2	24	27.6
Dont DMI	16	6.3	12	5.6	7	7.4	5	5.7
Dont autre DM	0	0	8	3.7	0	0	5	5.7
Une infection liée aux soins	57	22.3	63	29.4	17	17.9	28	32.2

Tableau 1.9 : Répartition des EIG par année et combinaison d'incident

EIG lié à	EIG Total				EIG Evitables			
	2004		2009		2004		2009	
	Nombre	%	Nombre	%	Nombre	%	Nombre	%
Total	255		214		95		87	
Une procédure seule	117	45.9	72	33.6	41	43.2	21	24.1
Un produit de santé seul	35	13.7	32	15	15	15.8	18	20.7
Une infection liée aux soins seule	4	1.6	9	4.2	0	0	5	5.7
Une procédure et un produit de santé	24	9.4	35	16.4	10	10	14	16.1
Un produit de santé et une infection	3	1.2	2	0.9	2	2.1	0	0

Pour terminer, les deux Tableaux 1.10 et 1.11 permettent de se faire une idée de la fréquence (densité d'incidence) de ces EI sur une période de 1000 jours d'hospitalisation. Ainsi 1.7 jours d'hospitalisation sur 1000 sont dus à un médicament et parmi eux 0.7 jours (pour 2009), soit presque la moitié, était évitable.

Tableau 1.10 : Répartition des EIG par type et année pour 1000 jours d'hospitalisation

EIG lié à	EIG Total				EIG Evitables			
	2004		2009		2004		2009	
	Nombre	‰	Nombre	‰	Nombre	‰	Nombre	‰
Une procédure	191	5.5	159	4.3	66	2.0	58	1.7
Un produit de santé	71	1.6	83	2.5	30	0.7	41	1.1
Dont médicament	52	1.3	56	1.7	22	0.6	24	0.7
Dont DMI	16	0.3	12	0.4	7	0.1	5	0.2
Dont autre DM	0	0	8	0.2	0	0	5	0.2
Une infection liée aux soins	57	1.3	63	1.9	17	0.5	28	0.9

Tableau 1.11 : Répartition des EIG par combinaison d'incident et année pour 1000 jours d'hospitalisation

EIG lié à	EIG Total				EIG Evitables			
	2004		2009		2004		2009	
	Nombre	‰	Nombre	‰	Nombre	‰	Nombre	‰
Total	255		214		95		87	
Une procédure seule	117	3.9	72	1.9	41	1.3	21	0.6
Un produit de santé seul	35	0.9	32	1.1	15	0.4	18	0.6
Une infection liée aux soins seule	4	0.07	9	0.3	0	0	5	0.1
Une procédure et un produit de santé	24	0.5	35	0.9	10	0.2	14	0.5
Un produit de santé et une infection	3	0.05	2	0.06	2	0.02	0	0

1.4 COMMENT SECURISER LE CDM ?

La **DGOS** en partenariat avec les autres directions du ministère, la Haute Autorité de Santé (notée **HAS**), l'**ANSM**, la Mission de la Tarification à l'Activité (**MT2A**) et l'Agence Nationale d'Appui à la Performance, ont donc mis en place et piloté plusieurs dispositifs, mesures et lois afin de mener une politique de sécurisation du CDM et de la prise en charge dans les établissements. On notera en particulier :

- ❖ La lutte contre l'iatrogénie médicamenteuse définie par la loi de Santé Publique du 09 août 2004 comme un axe prioritaire dont les objectifs sont l'amélioration de la qualité des soins et la réduction de l'iatrogénie liée aux produits de santé ;
- ❖ L'arrêté du 6 avril 2011 relatif au management de la prise en charge médicamenteuse ;
- ❖ Un document de travail (2004), élaboré par la DHOS en coordination avec un groupe d'experts, concernant la prise en charge thérapeutique du patient hospitalisé ;
- ❖ La Loi Hôpital, Patients, Santé, Territoires n°2009-879 du 21 juillet 2009 place aussi la qualité et la sécurité des soins au cœur des missions des ES ;
- ❖ Le décret du 31 octobre 2008 introduisant le contrat de bon usage des médicaments (noté **CBU**), des produits et des prestations et instaurant le dispositif contractuel. Par conséquent, cela impose aux établissements de santé de s'engager sur des actions visant à améliorer et à sécuriser le circuit du médicament ;
- ❖ Ce dispositif contractuel fut également à l'origine de la mise en place d'une politique régionale du médicament et des produits de santé animée par les Observatoires des Médicaments, des Dispositifs Médicaux et des Innovations Thérapeutiques (notés **OMEDIT**) ;
- ❖ La prise en charge médicamenteuse devenant une pratique exigible prioritaire dans la version 2010 du manuel de certification des établissements conçu par la HAS ;
- ❖ Les chantiers sur l'organisation du circuit du médicament, menés par l'ANAP (Mission nationale d'Expertise et d'Audit Hospitalier notée **MEAH**).

A ces démarches s'ajoute le plan « Hôpital 2012 » (investissement et modernisation) qui incite de nouveaux établissements à lancer le démarrage de projets (plus de 70 % des projets retenus sont liés à l'informatisation). Il y a donc actuellement un grand nombre de projets dont les résultats ne seront visibles que dans 4 à 5 ans et qui touchent directement à la sécurisation du CDM.

La sécurisation du CDM sera donc une démarche complexe et difficile à mettre en œuvre (le touchant à tous les niveaux) comme nous le verrons dans ce qui suit. Une première certitude est que celle-ci dépendra très fortement du contexte local qui est tributaire des personnes et des moyens dont disposent les établissements de santé. En effet, bien que pilotable à différents niveaux, que ce soit national, régional ou local, c'est ce dernier qui offre le plus d'opportunités aux professionnels pour s'engager dans des actions de sécurisation.

De plus, la sécurisation du CDM passera par celle des trois sous-systèmes interdépendants : le processus dit clinique, de la prescription à l'administration ; le processus logistique, de la réception/production du médicament jusqu'au lit du patient ; le processus d'évaluation du résultat de la thérapie médicamenteuse (cause et effets d'une administration/non administration, administration conforme/non conforme, etc.).

D'une manière plus générale, sécuriser un système physique revient à accroître ce que l'on appelle sa résilience.

Définition 1.15 : « La résilience est l'aptitude d'une organisation à résister aux situations présentant des dangers avec un minimum de dommages. La résilience d'un système, au sens technique, humain, et organisationnel, se construit en deux temps :

- lors de la conception, en cherchant à développer un système sûr, résistant aux événements imprévus, pourvu de défenses efficaces.
- lors de la vie du système, en analysant les incidents, leur gestion par les acteurs et le comportement des défenses, afin d'identifier aussi bien les fragilités que les ressources qui ont permis de les gérer au mieux et de tirer les leçons de cette analyse. » [\[ANAES 03\]](#)

Dans cette définition, il apparaît clairement que la place de l'analyse des incidents est considérée comme une étape indispensable et indissociable de la sécurisation du CDM. Elle est notamment caractérisée par ce qui a été vu précédemment, les différents rapports et études menées en lien avec les EI, mais aussi par un fort développement des structures internes pour le suivi des incidents et de nouveaux référentiels qualité de plus en plus exigeants avec le monde hospitalier.

1.4.1 Les recommandations ministérielles et les lois

La sécurisation du CDM passera donc par une dynamique qualité se traduisant au niveau de chaque sous-système par une amélioration du cadre de contrôle du circuit ainsi que par une modernisation des outils et une augmentation des moyens à la disposition du personnel hospitalier. Ainsi, dans le cadre de cette amélioration on pourrait retrouver des prescriptions faites par un médecin avec une assistance (logiciel d'aide à la prescription, pharmacien), une validation informatique (ou en temps réel si le pharmacien est présent durant la consultation) de la prescription par le pharmacien, une amélioration de la dispensation (par l'insertion de robot), de l'administration et de la surveillance par l'infirmière.

Cette dynamique qualité doit cependant être engagée autant au niveau organisationnel, économique que stratégique afin de toucher à tous les niveaux du CDM, d'une manière transversale et verticale.

Bien entendu, de nombreuses lois et arrêtés ministériels ont été établis afin de définir et de rendre plus sûr le CDM. L'un d'entre eux fait tout particulièrement le lien avec cet aspect qualité auquel doit être soumis le CDM afin de le sécuriser. Il s'agit de l'arrêté du 6 avril 2011 sur le management de la qualité.

Celui-ci évoque notamment le fait que désormais,

« [...] les établissements de santé se doivent de disposer d'un système de management de la qualité afin d'assurer la qualité et la sécurité de la prise en charge médicamenteuse du patient qui s'inscrit dans une politique d'amélioration continue de la qualité, de la sécurité des soins et de la gestion des risques qui sont associés à ces derniers. »

Désormais l'aspect qualité prédomine et implique de nombreuses actions à mener pour les établissements de santé. L'identification et l'analyse du circuit en est une partie, étant donné que celles-ci permettront une meilleure compréhension du système. L'informatisation de la prise en charge médicamenteuse en est une autre, étant citée comme un point essentiel de la sécurisation par de nombreux documents ([[DHOS 09](#)], [[DAHAN 10](#)], [[CAUTERMAN 07](#)]).

Reprenons cet arrêté, en se concentrant sur certains de ses articles afin de mettre en évidence toutes les actions devant être entreprises et formalisées.

Introduisant dans un premier temps la notion de management de la qualité (**Article 3** : « Système de management de la qualité : exigences générales ») se référant à l'article L. 6111-2⁷ du code de la santé publique, il impose par la suite la mise en place d'une disposition organisationnelle de ce système de management lui permettant de rendre compte de ses activités aux différentes instances de l'établissement (**Article 5** : Dispositions organisationnelles). La mise en place d'un système documentaire contenant les documents relatifs à la qualité et sa maîtrise sont imposées par l'**Article 6**. Il impose de plus que la direction doit s'assurer de l'accessibilité de ces documents et de leur diffusion. Le suivant (**Article 7** : « responsabilité et formation ») implique la notion de responsabilité de formation du personnel, afin de le mettre à niveau quant aux questions de qualité et de sécurité de la prise en charge médicamenteuse.

Ces premiers articles se concentrant sur le lien direct de la qualité avec le personnel, les suivants se focalisent plus sur l'étude et la gestion des ErM, EI, etc. Ainsi l'**Article 8** (« Etude des risques encourus lors de la prise en charge de médicaments ») oblige l'établissement à mener cette étude et à s'assurer de la réalisation et de la rédaction de procédures et de modes opératoires engageant les responsabilités médicales, pharmaceutiques, infirmières, etc. permettant de réduire les risques :

- ❖ S'assurer de l'identité du patient durant l'hospitalisation, notamment lors de l'administration du traitement ;
- ❖ S'assurer que la prescription garantit la qualité de la prise en charge tout au long du séjour et qu'elle est conforme au standard défini ;
- ❖ Garantir que le médicament est administré sous la bonne forme pharmaceutique, la bonne dose, au bon moment, par la bonne voie d'administration pour être en concordance avec la prescription médicale et que le schéma posologique, la préparation et le suivi thérapeutique sont conformes ;

⁷<http://www.legifrance.gouv.fr/affichCodeArticle.do?cidTexte=LEGITEXT000006072665&idArticle=LEGIARTI000020886452&dateTexte=>

- ❖ S'assurer de la qualité du circuit logistique lié au CDM (approvisionnement, délivrance, concordance bon de réception et bon de demande, conditions de stockage) ;
- ❖ S'assurer que le patient est bien informé sur son traitement, s'il est apte à comprendre ;
- ❖ S'assurer de la traçabilité lors des différentes étapes du CDM.

A son tour l'**Article 9** oblige lui aussi à mettre en place une structure permettant l'analyse et la détermination d'actions d'amélioration pour pallier aux ErM, EI ou dysfonctionnements liés à la prise en charge médicamenteuse. Pour sa part l'**Article 10** (« Formalisation de l'organisation adoptée pour traiter les déclarations internes en vue d'améliorer la qualité et la sécurité de la prise en charge médicamenteuse ») impose l'enregistrement des procédures précédemment définies dans le système documentaire. Enfin, l'**Article 12** (« Communication ») se concentre sur la nécessité de communiquer à tout le personnel les actions entreprises au niveau de la prise en charge médicamenteuse et les procédures mises en place pour la déclaration de tous les incidents.

1.4.2 Que faire à chaque étape du CDM d'après ces lois ?

L'arrêté du 6 Avril 2011 faisant un tour d'horizon des différentes lois et articles existants, les recommandations suivantes en sont majoritairement extraites, la plupart des rapports ministériels ou d'agences de l'Etat préconisant des mesures très similaires. Ainsi ces recommandations se concentrent sur 9 items : la prescription, la dispensation, la préparation, l'approvisionnement, la détention et le stockage, le transport, l'information du patient (gestion de son traitement), l'administration des médicaments et enfin la gestion des stupéfiants.

1.4.2.1 Pour la prescription

Celle-ci est rédigée pendant ou après l'examen du patient hospitalisé, sur une ordonnance (papier ou informatique) où sont spécifiées les informations suivantes :

- ❖ Le nom, la qualité et, le cas échéant, la qualification, le titre ou la spécialité du prescripteur (en référence à l'article R. 5121-91⁸ du CSP), son identifiant lorsqu'il existe, ainsi que le nom, l'adresse de l'établissement et les coordonnées téléphoniques et électroniques auxquelles il peut être contacté. Pour terminer il doit aussi apparaître sur ce document sa signature et la date à laquelle l'ordonnance a été rédigée ;
- ❖ La dénomination du/des médicament(s), du/des produit(s) prescrit(s), ou le(s) principe(s) actif(s) du/des médicament(s) désigné(s) par la dénomination commune internationale (notée **DCI**), ainsi que la posologie, le mode d'administration et, s'il s'agit d'une préparation, la formule détaillée ;
- ❖ La durée de traitement ;
- ❖ Les nom et prénom du patient, son sexe, son âge, sa taille et son poids.

La gestion de la transmission de la prescription reste à l'appréciation de l'établissement de santé et peut donc se faire sous forme papier ou bien de manière informatisée sous réserve de la présence d'une signature électronique permettant de l'authentifier et de l'identifier et qu'il soit

⁸ Code de la santé publique,

<http://www.legifrance.gouv.fr/affichCodeArticle.do?cidTexte=LEGITEXT000006072665&idArticle=LEGIARTI000006914836&dateTexte=&categorieLien=cid>

possible de l'imprimer. Pour le cas des patients non hospitalisés, l'ordonnance doit être rédigée conformément à l'article R. 5132-3⁹ du CSP.

1.4.2.2 Pour la dispensation.

En se basant sur l'article R. 4235-48 du CSP régissant cette étape, les principes suivants se doivent d'être appliqués lors de cette étape:

- ❖ L'intégralité de l'acte de dispensation du médicament est assuré par le pharmacien qui en plus de cette délivrance, se doit de :
 - ✓ Réaliser l'analyse pharmaceutique de l'ordonnance médicale si elle existe ;
 - ✓ Faire la préparation éventuelle des doses à administrer ;
 - ✓ Mettre à disposition les informations et des conseils nécessaires au bon usage du médicament.
- ❖ Le pharmacien a aussi pour devoir de participer au soutien du patient par ses conseils et dans le domaine de ses compétences ;
- ❖ Les médicaments peuvent être encore délivrés globalement à l'unité de soins en attendant une informatisation du circuit. Cette livraison se fera en renouvellement d'une dotation préalablement validée et définie par le pharmacien et le médecin responsable de l'unité de soins ou à défaut celui désigné par l'ensemble des prescripteurs concernés ;
- ❖ Les médicaments à risque doivent être validés d'un point de vue pharmaceutique.

1.4.2.3 Pour les préparations.

Celles-ci doivent être conçues conformément aux Bonnes Pratiques de Préparation prévues à l'article L. 5126-2¹⁰ du CSP pour les établissements disposant d'une PUI ayant les agréments nécessaires.

1.4.2.4 Pour l'approvisionnement.

Le CSP, le code des marchés publics, ainsi que les autres réglementations en adéquation, font office de référence. Ainsi, le pôle achats et approvisionnements devra s'y conformer afin d'appliquer toute la politique de prévention des risques liés aux médicaments.

L'objectif étant dès lors de faire de la fonction approvisionnement, l'acteur de la sécurisation de l'acheminement des produits et de leur stockage dans l'établissement. Quant à la fonction réception (si existante) de la PUI, son rôle est de fournir un premier contrôle de conformité des produits délivrés en vérifiant l'adéquation entre eux et les bons de commandes, et de livraison.

1.4.2.5 Pour la détention et stockage.

Le stockage des médicaments doit avoir lieu dans des locaux dotés d'armoires ou de tout autre dispositif de rangement disposant d'un mode de fermeture sécurisée garantissant les conditions de conservation et l'intégrité des produits. Ces locaux devant être eux aussi en conformité avec les bonnes pratiques de pharmacie hospitalière.

⁹ Code de la santé publique,

<http://www.legifrance.gouv.fr/affichCodeArticle.do?idArticle=LEGIARTI000006915532&cidTexte=LEGITEXT000006072665&dateTexte=20081204>

¹⁰<http://www.legifrance.gouv.fr/affichCodeArticle.do?cidTexte=LEGITEXT000006072665&idArticle=LEGIARTI000006690068&dateTexte=&categorieLien=cid>

Les conditions liées au stockage et à la détention de médicaments dans le cadre d'une PUI sont :

- ❖ Les dispositifs de rangements de chaque unité sont discutés et définis par un accord entre le pharmacien et le responsable de cette tâche pour chaque unité (cadre de santé ou équivalent) ;
- ❖ La sécurisation de l'accès aux médicaments détenus dans chaque unité passe par la définition des procédures de commandes, de réceptions, et des conditions de stockage. Cette fonction est tenue par la personne désignée par le responsable de l'unité. Il peut s'agir d'un cadre de santé ou équivalent, mais aussi d'un IDE ;
- ❖ La dotation de médicaments d'une unité, dont le but est de parer aux besoins urgents, est définie par le pharmacien et le médecin responsable de l'unité (ou à défaut celui désigné par l'ensemble des prescripteurs) après consultation du cadre de santé ou de l'infirmier désigné ;
- ❖ Le bon respect des règles de stockage (étiquetage, mode de détention, etc.) est sous la responsabilité du pharmacien ou de la personne en charge désignée par celui-ci. Il contrôle de plus l'adéquation des quantités stockées avec les prescriptions ;
- ❖ La détention des médicaments doit se faire autant que possible dans leur emballage d'origine. Si cela ne peut être fait, le conditionnement doit vérifier et respecter les dispositions suivantes :
 - ✓ Stupéfiants : étiquette blanche avec large filet rouge ;
 - ✓ Liste I : étiquette blanche avec large filet rouge ;
 - ✓ Liste II : étiquette blanche avec large filet vert.

Toutes ces étiquettes étant soumises elle-même à des obligations de signalisation précises. Il devra y figurer :

- ❖ Dénomination de la spécialité, DCI ou principes actifs ;
- ❖ Dosage (en quantité et/ou concentration) ;
- ❖ Forme pharmaceutique ;
- ❖ Voie d'administration ;
- ❖ Date de péremption ;
- ❖ Numéro de lot.

La traçabilité des médicaments implique que l'identification de ceux-ci doit pouvoir avoir lieu à n'importe quel moment dans le circuit et ce jusqu'à l'administration.

Des audits réguliers permettant de s'assurer de la bonne qualité des conditions de détention et de stockage des médicaments dans les unités fonctionnelles seront réalisés par un pharmacien (ou une personne désignée par celui-ci). Le but étant d'éviter toute confusion entre les produits durant le stockage. Ces audits produiront des procès-verbaux datés et cosignés par le pharmacien et le responsable de l'unité fonctionnelle (ou leurs équivalents).

1.4.2.6 Pour le transport

Les critères importants liés au transport sont les bonnes conditions d'hygiène et de sécurité permettant de respecter les contraintes inhérentes à chaque produit (température, exposition à la lumière, faible stabilité, urgent, etc.). Un responsable du transport PUI/Unité fonctionnelle est de plus désigné.

1.4.2.7 Pour l'administration

Seul le personnel défini comme compétent dans les cadres des lois est autorisé à effectuer l'administration des médicaments aux patients. L'administration est de surcroît soumise à des vérifications et contrôles obligatoires :

- ❖ Identité du patient et des médicaments à administrer, au regard de la prescription médicale.
- ❖ Date de péremption des médicaments et leur aspect.
- ❖ Mode d'administration.

Toute administration de médicament (notamment les stupéfiants) doit être accompagnée d'un enregistrement (dose administrée et heure d'administration) qui a lieu sur le support de prescription et cela au moment où l'action est réalisée. De même, toute non administration doit être signalée sur le support de prescription qui sera accompagnée d'une déclaration de cet évènement au pharmacien et au prescripteur.

1.4.2.8 Pour la gestion du traitement personnel du patient

Afin d'éviter tout problème d'iatrogénie, aucun médicament hors livret thérapeutique de l'établissement ne sera laissé à disposition du patient. Ceci ne concerne pas les stupéfiants étant de toute manière placés dans des armoires sécurisées en permanence.

Dans le cas où un patient ne serait pas en condition de gérer lui-même son traitement, il sera nécessaire de sécuriser celui-ci en ne les lui laissant pas à disposition.

1.4.2.9 Dispositions particulières concernant les stupéfiants

Comme déjà énoncé, le cas des produits classés comme stupéfiants est soumis à des articles précis du CSP. Ainsi leur prescription doit être en conformité avec l'article R. 5132-29¹¹ pour les patients étant soignés dans un ES. En ce qui concerne les patients non hospitalisés ou les prescriptions de sortie, celles-ci doivent être conformes à l'article R. 5132-5¹².

Pour tout ce qui est en rapport avec l'administration de ce type de produits (stupéfiants ou soumis à la même réglementation), cette action doit faire l'objet d'une transcription, sur un document spécial ou sur le document de prescription, des données suivantes comme mentionnées à l'article R. 5132-3 :

- ❖ Le nom de l'établissement ;
- ❖ La désignation de l'unité de soins ;
- ❖ La date et l'heure d'administration ;
- ❖ Les nom et prénom du malade ;
- ❖ La dénomination du médicament et sa forme pharmaceutique ;
- ❖ La dose administrée et la posologie en toutes lettres ;
- ❖ L'identification du prescripteur ;
- ❖ L'identification de la personne ayant procédé à l'administration et sa signature.

Ces documents, identifiant l'administration, seront en plus datés et signés par le médecin responsable de l'US pour être conservés, à la pharmacie, pour une durée de trois ans. Ce document

¹¹<http://www.legifrance.gouv.fr/affichCodeArticle.do?idArticle=LEGIARTI000006915577&cidTexte=LEGITEXT000006072665&dateTexte=20081204>

¹²http://www.legifrance.gouv.fr/affichCodeArticle.do?jsessionid=13AC0020877EC96AE656754F896940D8.tpdjo04v_1?idArticle=LEGIARTI000006915537&cidTexte=LEGITEXT000006072665&dateTexte=20110607

peut aussi bien prendre une forme informatique que papier, le plus important étant la présence des date et signature (pouvant donc être électronique).

Un autre aspect important de la gestion des stupéfiants et apparentés est la demande en cas de besoins urgents. Afin de fournir ces produits, la pharmacie doit obtenir de la part du service, un état récapitulatif, signé par le médecin responsable, des stocks de l'unité accompagnés des doubles des documents de prescriptions, avec la demande formulée sur un imprimé de couleur rose. Ce document récapitulatif sera de plus fourni avec les relevés d'administration concernant les médicaments qui ont été prélevés dans cette dotation.

Le personnel de la pharmacie pouvant délivrer ces médicaments au cadre de santé (ou son équivalent) désigné par le médecin responsable de l'unité de soins et le pharmacien, ou bien au prescripteur lui-même, sont :

- ❖ Le pharmacien ;
- ❖ Les internes en pharmacie ;
- ❖ Les étudiants de cinquième année hospitalo-universitaire délégués par le pharmacien ;
- ❖ Les préparateurs en pharmacie sous contrôle des pharmaciens.

La sécurité entourant les médicaments apparentés stupéfiants nécessite un stockage spécial et adéquat. Ainsi, ceux-ci se retrouvent dans des armoires (ou tout autre dispositif), des locaux verrouillés (clé, code, etc.) réservés et dédiés à leur stockage.

1.4.3 Quelles voies de sécurisation ?

Parmi les nombreuses recommandations, six voies de progrès ressortent de manière répétée et permettent une amélioration significative et rapide de la sécurité du CDM. Chacune d'entre elles se situent cependant à un niveau différent : certaines proposent une refonte globale du fonctionnement et d'autres seulement l'ajout d'outils supplémentaires.

L'intérêt de cette partie n'est pas de fournir toutes les améliorations possibles mais d'en citer quelques-unes pour mettre en évidence le fait que le CDM a dans sa généralité de nombreuses possibilités d'évolution, toutes ne pouvant s'appliquer à chaque hôpital, étant donné qu'elles sont tributaires des moyens disponibles, de la taille de l'établissement, de ce qui est déjà en place, etc. En effet, la plupart des projets d'amélioration dans un hôpital ne peuvent pas comme en entreprise se faire durant une période d'arrêt. L'hôpital a la désagréable contrainte de devoir fonctionner 100% du temps, imposant donc de devoir mener en parallèle fonctionnement normal et mise en place des nouvelles solutions.

Afin d'être le plus logique possible, les solutions suivantes seront présentées par ordre supposée de mise en marche. En effet, il serait peu logique d'installer un robot de dispensation si le système n'est pas informatisé, tout comme il sera complexe et lourd à gérer de faire du nominatif sans le matériel adéquat. L'ordre suivi sera donc : Informatisation > Gestion des armoires > Dispensation > Dose unitaire > Automatisation > Pharmacien intégré au service.

1.4.3.1 Informatisation

Devant l'évidence que l'informatisation du CDM est devenue une condition essentielle à sa sécurisation, permettant un meilleur suivi des différentes étapes du circuit, celle-ci est désormais vue comme la pierre angulaire de nombreux projets hospitaliers. Ces systèmes d'information (noté **SI**), se posent donc en élément stratégique pour les hôpitaux en quête d'efficacité organisationnelle.

Cependant, accorder à la seule informatisation l'amélioration de la sécurisation serait une erreur, celle-ci ne pouvant à elle seule, solutionner tous les problèmes. Un projet tel que l'informatisation de la prise en charge médicamenteuse repose tout d'abord sur la qualité et l'efficacité de l'organisation du CDM.

↳ Mais qu'apporte-t-elle réellement?

Les systèmes d'information et l'informatisation permettent notamment d'avoir un meilleur suivi des modalités de dispensation et d'administration des produits grâce à une meilleure maîtrise des flux informationnels. Elle constitue donc une base solide pour le respect du CBU, en permettant le recueil et la transmission quantitative et qualitative des données. En effet, de nombreux logiciels permettent de suivre en temps réel les commandes aux fournisseurs, les demandes des services (globale ou nominative) ainsi que le niveau des stocks. L'autre avantage est que l'informatisation a une forte propension à décloisonner une organisation autant d'une manière intra-service qu'interservices du fait de cette circulation des informations.

↳ Quelle place tient-elle actuellement ?

De nombreuses études ont identifié l'avancement des projets d'informatisation ou la place qu'elle tient déjà dans certains d'entre eux. Le rapport [\[DHOS 09\]](#), permet de dresser un bilan de l'informatisation :

- ❖ Une solution informatique est déjà en place dans un établissement sur trois (69% CHU, 24% des établissements privés) ;
- ❖ Les trois quarts des établissements ont un projet informatique en cours de déploiement ;
- ❖ La solution informatique déjà mise en place ou en déploiement, concerne dans 80% des cas, toutes les étapes de la prise en charge médicamenteuse, inscrivant alors celle-ci dans une démarche d'informatisation intégrée de la production des soins ;
- ❖ A la date de cette enquête, en moyenne 21% des lits sont informatisés. Le rythme moyen de déploiement informatique est de 8% des lits par an ;
- ❖ La mise en place de l'informatisation se ressent fortement sur la charge de travail d'une PUI : de manière générale, le déploiement nécessite environ 0.35 ETP pharmacien et 0.15 ETP préparateur pour 100 lits par an. Cette valeur globale est de plus dépendante de la typologie de la structure qui réalise ce déploiement. De cette étude, retenons les valeurs suivantes : CHU : 0.9 ETP pharmacien/100 lits/an, clinique privée : 0.1 ETP pharmacien/100 lits/an ;
- ❖ Il ressort la forte pluridisciplinarité des équipes projets : médecin, cadre, personnel soignant, pharmacien, préparateur, direction des systèmes d'information ;
- ❖ La jeunesse du marché des SIH, fait que celui-ci est en constante évolution et est encore très atomisé (12% de parts de marché pour le leader). Il est aussi constaté qu'une segmentation du marché se met en place (par catégorie et par taille d'établissement) ;
- ❖ Thériaque (secteur public essentiellement), Vidal (plus utilisée dans le secteur privé) et Claude Bernard sont trois bases de données d'aide à la prescription de médicaments validées par la HAS ;
- ❖ Le budget de l'informatisation a tendance à se répartir en 3 tranches : 33% du marché autour de 20k€, 33% autour de 50k€, 33% entre 100k€ et 250k€.

↳ Quelles contraintes liées à l'informatisation ?

La plus évidente d'entre elles est qu'une telle démarche nécessite un fort investissement de la part des pharmaciens, médecins et infirmiers. Malheureusement, ils ne disposent pas forcément de l'envie ou du temps nécessaire, ce qui rallonge d'autant plus le temps de déploiement de l'informatisation.

De surcroît, le projet doit faire face la plupart du temps à une forte réticence du personnel au changement de leurs habitudes et vis-à-vis des formations qu'ils devront suivre afin de pouvoir utiliser et travailler avec les nouveaux outils.

La dernière contrainte, et non la moindre, est le coût de la mise en place d'un tel projet, qui se chiffre pour une structure de type CHU en millions d'euros, rajoutant pression et obligation rapide de résultats pour le retour sur investissement.

↳ Comment mettre celle-ci en marche ?

Le développement de l'informatisation se fait en deux temps. En premier, une phase projet permettant de définir le cadre puis suit alors une phase de mise en route et de montée en charge jusqu'à l'utilisation en routine.

La première phase permet donc de définir ce à quoi va servir l'informatisation et comment celle-ci va s'intégrer au fonctionnement actuel de l'hôpital. Celle-ci va notamment concerner la sélection de logiciels. Il est important de noter que la couverture fonctionnelle d'un logiciel n'est pas l'unique critère de sélection pour un outil informatique. L'investissement demandé, le coût de fonctionnement, le retour sur investissement, l'intégration, l'interopérabilité avec des logiciels déjà en place, etc., sont aussi des éléments primordiaux pour la sélection, afin de vérifier que la solution qui sera mise en place est en adéquation avec les attentes et les besoins de l'établissement.

Couplé à ces aspects techniques et financiers, la sélection des outils informatiques doit aussi prendre en compte des aspects organisationnels. Il est notamment nécessaire de considérer:

- ❖ L'adaptation des organisations ;
- ❖ La méthode d'intégration et de conduite des opérations, l'adéquation des différentes étapes du projet et leur évaluation ;
- ❖ Le choix d'un ou plusieurs services pilotes et l'évaluation de cette phase test ;
- ❖ La formation et l'assistance des professionnels utilisateurs ;
- ❖ La rédaction d'une charte des conditions d'usage des logiciels ;
- ❖ La conformité avec les exigences de sécurité, la confidentialité des informations à caractère personnel et la disponibilité du système d'information (noté **SI**).

Afin de mener le projet à terme et cela dans de bonnes conditions, l'implication dès l'initialisation de la démarche des professionnels de santé (médecins, soignants, pharmaciens, logisticiens, préparateurs, cadres de santé) est un impératif. Ceci a pour objectif de forcer ces professionnels à prendre des décisions liées aux outils informatiques, pour que ceux-ci soient au maximum en adéquation avec leur pratique et cela pour une sécurité optimale.

Le passage à la seconde phase sera entériné par un essai permettant de valider l'utilisation des outils et leur conformité avec les objectifs fixés. En contribuant à la sécurisation par des contrôles logiques (sur des données ou sur des actions automatisées), la phase d'usage de ces logiciels va permettre de vérifier que tous ces contrôles sont bien opérationnels. De même

l'ergonomie des logiciels, des équipements liés (douchette, etc.) et l'utilisation correcte de ceux-ci par les professionnels de santé seront aussi soumis à vérification.

Il est important de signaler que, même si l'informatisation permet de sécuriser la prise en charge médicamenteuse, elle amène cependant son lot d'erreurs et appelle donc à une vigilance de tout instant de la part des professionnels. C'est pourquoi l'arrêté du 6 avril 2011 formule, par le biais de ses articles 8 et 9, le fait que l'utilisation des outils informatiques est soumise à une analyse préalable et que tous les dysfonctionnements du SI et les EI qui en découlent font l'objet de procédures de déclaration pour analyse et correction.

1.4.3.2 Une nouvelle approche de la gestion et du stockage dans les services

Aujourd'hui, il existe trois grands modes de gestion des stocks dans les services et chacun d'entre eux est lié à un type d'armoire :

- ❖ La gestion standard avec accès libre pour le personnel soignant (armoire classique) ;
- ❖ La gestion plein/vide (armoire avec double-bac) ;
- ❖ La gestion sécurisée avec des armoires adaptées (armoire informatisée).

Chacune d'entre elles présente bien entendu des avantages et des inconvénients. Cependant comme nous allons le voir, il serait nécessaire de tendre vers un fonctionnement avec des armoires sécurisées. Pour le moment reprenons chaque cas.

↳ Les armoires en fonctionnement standard :

Elles présentent le gros avantage d'être fonctionnelles. En effet, le personnel soignant peut facilement y accéder, de jour comme de nuit, et dispose de l'entièreté des médicaments dont il peut avoir besoin. L'armoire est de plus gérée par le service, qui transmet ses demandes à intervalles réguliers à la pharmacie, cette dernière en assurant donc l'approvisionnement. Cependant, malgré l'avantage majeur de donner de la flexibilité aux services, elles présentent de gros désavantages :

- ❖ Une faille sécuritaire importante : le contrôle qui doit avoir lieu au moment de l'administration afin de comparer produits administrés/produits prescrits est sous l'entière responsabilité et vigilance de l'infirmière. Dès lors, il existe un fort risque de confusion entre deux produits, risque s'intensifiant avec l'augmentation de la fréquence de modification des produits (renouvellement des marchés, passage en générique, etc.) ;
- ❖ Il a été constaté que ces armoires créent majoritairement un sur-stockage dans les services et ceci pour deux raisons. Un manque de contrôle des stocks disponibles et le besoin du sentiment de « sécurité » inspiré par cette forte disponibilité de produits. Le phénomène de stocks « fantômes » ou de réserves « secrètes » est donc encore très présent malgré la volonté de réduction affichée par les établissements ;
- ❖ A ce problème cité ci-dessus, se greffe celui de la gestion des retours (médicaments non administrés). Souvent mis de côté pour la raison du « Ce médicament est souvent utilisé », seuls les médicaments les plus coûteux profitent de ce flux de retour ;
- ❖ Ces deux problèmes précédents ont donc pour conséquence de générer des périmés, dont la seule solution pour les éliminer est un contrôle régulier des armoires ;
- ❖ Le temps d'inventaire de ces armoires étant très long (pour les infirmières ou les préparateurs), il n'est donc que rarement fait complètement ;
- ❖ Dernier point, ces armoires permettent le « coulage », le personnel soignant se servant pour ses propres besoins, chose non traçable par le biais de demandes globales.

↳ La gestion en mode plein/vide :

Le stade intermédiaire est ici représenté par la gestion en mode plein/vide des armoires. Ce système basé sur l'utilisation de deux bacs dotés d'étiquettes à positionner sur un tableau de commande – à la manière du Kanban – pour réapprovisionner les stocks commence à tenir une place de plus en plus importante dans les établissements. En effet, cette gestion représente un gain de temps infirmier dans son utilisation journalière et un gain de place, le remplissage étant optimisé. De surcroît, ce système plein/vide est plus sécurisé, remplaçant une retranscription manuelle des médicaments à demander par un système automatisé de douchette/code barre ou bien par des puces RFID. Cela évite ainsi une rupture en proposant une gestion des commandes semi-automatisée. Ce fonctionnement permet donc de palier en partie aux problèmes liés à la gestion des périmés, des surstocks et aussi du coulage.

Cependant la mise en place de ces armoires n'est pas exempt de problèmes. Notamment :

- ❖ La définition des dotations doit être réalisée entre les pharmaciens et le cadre de santé du service ;
- ❖ Une mise à jour régulière des dotations est nécessaire.

Ce mode de gestion est notamment recommandé pour les services où la mise en place d'une gestion nominative est impossible. C'est le cas notamment des services de réanimation et des urgences où le personnel se doit d'être le plus réactif possible, le délai entre prescription et administration devant être le plus bref possible. La gestion plein/vide leur permet donc de bénéficier des avantages du système en libre accès tout en sécurisant et rationalisant en partie leurs stocks.

↳ Le mode sécurisé :

Pour terminer, ce mode de gestion est celui préconisé dans de nombreux rapports tel que le guide de la Haute Autorité de Santé de Juillet 2011 [**HAS 11**] pour les cas où la DJIN n'est pas mise en place ou applicable. Les armoires sécurisées présentent alors l'énorme avantage de pouvoir être interfacées avec le logiciel de gestion de la pharmacie et donc d'être en lien avec les prescriptions faites aux patients et d'introduire ainsi une étape de vérification et de contrôle supplémentaire. En effet, pour que le personnel soignant puisse accéder aux médicaments celui-ci doit s'identifier par l'intermédiaire d'un badge et d'un identifiant puis faire la requête nécessaire permettant l'accès aux médicaments voulus. Ces armoires possèdent des tiroirs dont l'ouverture est commandée informatiquement et identifiée de manière lumineuse généralement (un voyant s'allume devant le tiroir à ouvrir). Les médicaments (formes orales, injectables) sont stockés sous forme unitaire reconditionnée ou sur-conditionnée permettant alors leur identification. La connexion des armoires au serveur/logiciel de la pharmacie permet la gestion en temps réel du stock, celle-ci pouvant déclencher une demande de réapprovisionnement dès que ses stocks sont inférieurs au seuil de sécurité. Ces armoires possèdent néanmoins, elles aussi des défauts :

- ❖ La demande d'accès aux médicaments est souvent considérée comme une perte de temps par le personnel soignant si celui-ci doit réagir vite à un problème ;
- ❖ Etant informatisée, toute coupure électrique ou problème logiciel, handicape le fonctionnement du service, même si ces problèmes sont très rares ;
- ❖ Le coût de ces armoires est très élevé compte tenu de leur capacité de stockage.

Au final, seule une gestion composite de ces armoires permettra de répondre aux différentes demandes des services. Il serait possible d'imaginer une solution où toutes les armoires de services

permettant de répondre aux urgences seraient des armoires sécurisées excepté dans les services de réanimation et les urgences ou un mode plein/vide leur garantirait de la flexibilité et un temps de réponse plus rapide.

Pour terminer, des recommandations d'ordre général sont aussi exposées :

- ❖ Privilégier le système plein/vide dans le cas de délivrance globale ;
- ❖ Favoriser l'adoption du conditionnement unitaire (voir partie suivante) ;
- ❖ Harmoniser et clarifier les étiquetages des armoires ;
- ❖ Aménager les locaux en favorisant un éclairage suffisant ;
- ❖ Responsabiliser le personnel en nommant un référent pour la gestion du stock dans l'US.

1.4.3.3 La dose unitaire

L'aspect intrinsèque d'un médicament ou de son emballage peut autant être gage de sécurité que d'erreur une fois déballé ou sorti de sa boîte si les informations dessus ne sont pas suffisantes. Dans le but de réduire au minimum ces erreurs, les pharmaciens hospitaliers demandent depuis plusieurs années la disponibilité de conditionnements entièrement individuels dans les hôpitaux, rendant possible dès lors une dispensation nominative sans perte d'informations en termes de dénomination du médicament (ou nom de la molécule), de son dosage, de sa date de péremption, et de son numéro de lot.

Deux approches sont alors envisageables et même complémentaires.

Tout d'abord le conditionnement unitaire pourrait être à la charge de l'établissement, ce conditionnement défini au niveau de l'établissement étant alors homogène quant aux informations retranscrites et au mode d'identification choisi (code barre ou data matrix). Un ES faisant ce choix peut alors adopter deux solutions. Soit celui-ci va sur-conditionner les doses sous blisters, en ampoules, etc. en utilisant un sachet sur lesquels seront définis ses propres critères d'identifications, soit il déconditionnera les médicaments reçus (applicables pour les formes sèches) pour les reconditionner avec le formalisme et l'emballage voulus par le biais d'un automate. Cette étape de déconditionnement - reconditionnement présente néanmoins deux inconvénients majeurs (en plus de la mobilisation de personnel dédié) qui sont une perte potentielle de stabilité de ces produits reconditionnés et l'autorisation de mise sur le marché qui ne serait alors plus garantie. Il serait alors préférable de privilégier des produits à fort taux de rotation et ceux pour la dispensation nominative. De surcroît, la grande diversité de formes et de tailles de blisters/plaquettes, impose de disposer d'automates capables de s'adapter ou forcera l'établissement à investir dans des robots pour automatiser la découpe des blisters/plaquettes.

La seconde approche, quant à elle, serait que l'industriel se charge de la réalisation du conditionnement unitaire, garantissant la stabilité du produit sur toute la chaîne, ce qui ne poserait pas de problème au niveau de l'autorisation de mise sur le marché. La production de nombreux médicaments suit déjà cette logique, notamment les formes orales sèches, en affichant le nom, le numéro de lot et la date de péremption pour chaque comprimé, lesquelles se présentent alors pour la plupart sous forme de plaquettes ou de blisters (découpables facilement suivant le nombre d'unités souhaitées pour la dispensation).

Au final, s'il est possible pour les ES d'utiliser la plupart des formes de conditionnement déjà faites par les industriels, il pourrait être pertinent de limiter le sur-conditionnement à des produits à risque dont une identification sans faille est impérative.

1.4.3.4 Des dispensations différentes

Une autre solution permettant de sécuriser hautement le CDM est l'utilisation de différents modes de délivrance/dispensation au sein d'une même pharmacie, la plupart des hôpitaux étant encore sur un système proche de la délivrance globale ou globalisée. Ainsi comme le précise l'article 7 de l'arrêté du 31 mars 1999, la délivrance peut se faire « globalement ou individuellement ». C'est donc vers cette deuxième tendance que la plupart des hôpitaux s'oriente, la délivrance nominative étant désormais partie intégrante des indicateurs du CBU depuis le décret n°2005-1023 du 24 août 2005.

Cette DIN ou DJIN consiste à délivrer pour chaque patient la dose exacte de chaque médicament, les différents produits étant regroupés par prise, et elle a pour effets majeurs de diminuer les stocks et le nombre de références dans les armoires de services, étant donné que les piluliers ne sont plus préparés dans les services mais au niveau de la pharmacie, ne nécessitant dès lors quasiment plus de stocks dans les services.

Bien que pendant de nombreuses années, cette DIN a constitué l'objectif affiché, des études [DAHAN 10], [LABORIE 08] ont mis en avant qu'il ne serait pas pertinent de fonctionner avec un système 100% en DIN malgré ses nombreux avantages. En effet, comme déjà cité plus haut, certains services ne peuvent fonctionner ainsi, les dotations pour soins urgents et les dotations pour les premières doses¹³ doivent aussi être prises en compte et ne rentrent donc pas dans le cadre de DIN.

1.4.3.5 Automatisation & Robotisation

Pour terminer, le point final de cette chaîne de sécurisation consiste en la mise en place de robots et d'automates permettant de sécuriser par leur fonctionnement le CDM. Ces robots/automates sont de formes diverses et variées suivant à quelle place ils sont situés dans le circuit. Nous retrouverons donc des robots de stockage, des automates de préparation, de conditionnement, des armoires robotisées. Faisons rapidement un tour d'horizon de chaque.

↳ Les robots stockeurs :

Ces robots peuvent se présenter sous formes de stockeurs rotatifs (Image 1.1 (a)), de magasins linéaires (Image 1.1 (b)) disposant de tapis pour aider à la préparation, etc. Alors que les premiers permettent de stocker sur une surface minimale un grand nombre de références (et disposent même de capacité de réfrigération pour certains), les seconds offrent des capacités de travail plus importantes que les précédents (peuvent alimenter plusieurs postes en même temps) mais nécessitent plus de place. Ces deux types de robots ont trouvé tout d'abord leur place dans des officines avant de se démocratiser dans le milieu hospitalier.

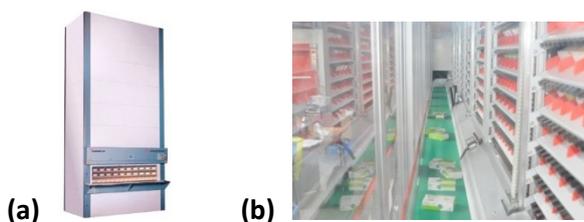


Image 1.1 : (a) Stockeur rotatif vertical (b) stockeur avec tapis

¹³ Dose initiale issue de la première prescription, à l'admission ou lors d'un changement de thérapeutique, qui doit être dispensée rapidement sans attendre le cycle de la DIN.

↳ Les automates de préparation :

Ces automates peuvent revêtir aussi plusieurs formes suivant le degré de robotisation de la préparation. Certains ne seront que de simples aides à la préparation pendant que d'autres la réaliseront entièrement (Image 1.2).

Ce marché est nettement plus développé que les simples robots de stockage et de nombreux fabricants sont donc positionnés sur ce segment. On retrouve notamment l'entreprise McKesson spécialisée dans l'informatique et la robotique hospitalière, Swisslog, Euraf, Omnicell, etc.



Image 1.2 : Robot de dispensation de McKesson



Image 1.3 : Robot de dispensation d'EURAF

Ces automates, comme vus sur les images ci-dessus, peuvent être de taille très variable allant de la dimension d'une simple armoire à celle de plusieurs pièces.

↳ Les robots de conditionnement :

Comme déjà expliqué plus haut, ces robots permettent de conditionner ou de sur-conditionner des formes orales sèches ou liquides, en fournissant des banderoles de médicaments ou des sachets individuels.

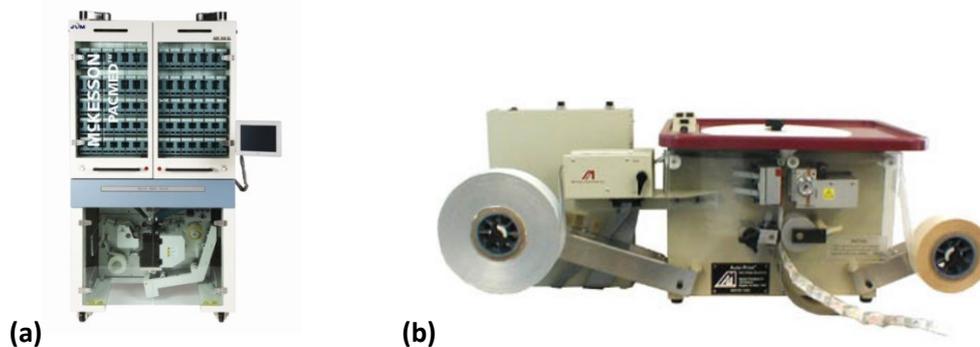


Image 1.4 : (a) PACMED de McKesson (b) Autoprint de EURAF

↳ Les armoires informatisées :

Le fonctionnement de ces armoires ayant été détaillé dans la partie correspondante, il ne sera pas décrit de nouveau ici. Il est aussi possible de trouver des chariots informatisés ayant un rôle similaire aux armoires.



Image 1.5 : Armoires et chariots informatisés et sécurisés

1.4.3.6 Des pharmaciens intégrés aux services.

Bien que l'intégration des pharmaciens dans les services puisse se faire sans qu'aucune des solutions suscitées ne soit mise en place, est ici présentée en dernier. L'hypothèse de base est que du fait des différentes actions entreprises, du temps pharmacien a pu être dégagé pour le délocaliser dans les services.

Intervenant autant au niveau de la politique générale du CDM dans son établissement que sur des activités cliniques, le métier de pharmacien s'est malheureusement mué en un univers tourné vers la distribution des produits manufacturés faisant oublier quelque peu sa valeur ajoutée. Il est donc utile de recentrer une partie de l'activité des pharmaciens sur le patient afin de tirer parti de sa vraie valeur, à savoir sa capacité à accompagner la distribution des produits de santé et de garantir par la même, sécurité et efficacité à la collectivité. Comme l'a très bien résumé Walton en 1961, la pharmacie clinique consiste en l'utilisation optimale des jugements et des connaissances pharmaceutiques et biomédicales du pharmacien, dans le but d'améliorer l'efficacité, la sécurité, l'économie et la précision selon lesquelles les médicaments doivent être utilisés dans les traitements des patients, faisant du pharmacien hospitalier un acteur incontournable de la sécurisation du CDM.

Ainsi le pharmacien en unité de soins peut intervenir à différents niveaux de la prise en charge médicamenteuse. En lien avec les patients et les différents professionnels de santé, celui-ci documente les pratiques médicamenteuses des patients (observations et entretiens pharmaceutiques), dans le but d'aider à la prise en charge du patient (validation des prescriptions, suivi thérapeutique, éducation thérapeutique, consultation de sortie).

1.5 CONCLUSION

Lors de ce premier chapitre, nous avons vu que le CDM est une entité complexe soumise à de nombreux aléas et problèmes. Constitué de nombreuses étapes, d'un cadre réglementaire en constante évolution et de plus en plus contraignant, ce circuit est de surcroît composé de un ou plusieurs sous-circuits interdépendants (dispensation nominative, délivrance globale, etc.). Les nombreuses contraintes et problèmes qu'il rencontre sont divers et variés, empêchant des solutions d'être trouvées, ces dernières ne pouvant répondre en une fois à toutes ces problématiques.

Ainsi, afin d'identifier avec plus de précisions le pourquoi de ces problèmes, nous avons parcouru de nombreux rapports – [CASTOT 10], [ENEIS 11] entre autres – pour classifier et différencier les erreurs qui en découlent, et notamment les ErM, erreurs très rarement issues d'une erreur individuelle et qui trouvent le plus souvent leur source dans des problèmes liés à l'environnement de travail (physique ou moral) menant l'individu à commettre un impair et à ne pas être en mesure de le signaler.

Par conséquent, il est impératif de renforcer ce circuit, approvisionnant chaque jour des milliers de patients, dans le but de le rendre plus sûr mais aussi plus efficace et performant. De nombreuses recommandations ont été faites autant au niveau national que local, afin d'apporter des réponses et des solutions à ces embuches rencontrées, telles que le CBU, auxquelles se rajoutent des voies de recommandations prenant diverses formes et portant sur de nombreux aspects (robotisation, informatisation, etc.) touchant l'intégralité ou une partie du CDM.

Ce dernier pouvant être vu comme une chaîne de production [HUET 11], nous allons tenter de trouver des outils permettant de mener une telle démarche, mais aussi de faire en sorte que ces outils soient utilisables et fonctionnels pour la profession la plus familière avec le CDM, c'est-à-dire les pharmaciens.

Cela sera l'objet du chapitre suivant, celui-ci portant sur des langages, méthodes et méthodologies originaires du monde industriel mais pouvant se prêter à nos travaux. Ce chapitre abordera notamment une des solutions permettant d'améliorer efficacement et rapidement le CDM, et qui reste pour nous l'une des plus pertinentes, si ce n'est la plus, à savoir la simulation.

III

DE L'EXISTANT À SA COMPRÉHENSION COMMENT Y PARVENIR?

RESUME DU CHAPITRE 2

Il y a de nombreuses possibilités pour comprendre et améliorer l'efficacité des procédures de dispensation, mais il est très difficile d'évaluer « le bénéfice risque » pour chacune d'entre elles. Mener une série d'expériences sur le terrain serait une possibilité, mais s'avèrerait être autant une perte de temps qu'une source de problèmes, certaines modifications pouvant générer des situations non prévues, potentiellement dangereuse et pouvant être non réversible.

La simulation prend alors tout son sens, cette dernière permettant de s'affranchir des conséquences pouvant survenir lors d'évolutions structurelles ou organisationnelles et donne alors l'occasion de mener plus loin certaine expérience, impossible à réaliser sur le terrain.

Cependant l'utilisation de la simulation requiert d'avoir en amont identifié, modélisé et analysé le fonctionnement du système étudié et nécessite donc l'emploi d'outils (langages graphiques, méthodes, méthodologies) dont certains seront détaillés ici. Ces outils sont certes nécessaires et obligatoires pour comprendre un système, mais, ils sont surtout à la source de toute simulation, celle-ci ne pouvant être appliquée sans étude préalable à un système inconnu, quel que soit le degré de granularité et la méthode de simulation utilisée.

Il s'agira donc de parcourir dans ce chapitre les différents outils rendant possible la simulation avant de s'intéresser à celle-ci.

SOMMAIRE DU CHAPITRE

2.1	INTRODUCTION	53
2.2	COMMENT CARACTERISER LE CDM ?.....	53
2.2.1	Pourquoi parle-t-on de processus dans le soin ?	53
2.2.2	Modélisation et modèle !.....	54
2.2.3	Qu'est-ce qu'un processus ?	55
2.2.4	L'approche par processus en quelques mots.....	57
2.2.5	Modélisation et granularité des processus	57
2.2.6	Les différents niveaux des processus.....	59
2.2.7	Les différentes classes de processus.....	60
2.3	QUELS LANGAGES POUR LA CARTOGRAPHIE ?	61
2.3.1	Les langages choisis	62
2.3.2	Evaluation des langages	65
2.4	METHODES & METHODOLOGIES.....	67
2.4.1	Les méthodologies de modélisation.....	68
2.4.2	Les méthodes de modélisation	71
2.4.3	Conclusion au niveau des méthodes.....	76
2.5	LA SIMULATION : POURQUOI ET COMMENT ?	76
2.5.1	Qu'est-ce que la simulation ?	76
2.5.2	Doit-elle être utilisée à chaque fois ?	77
2.5.3	Quels avantages et inconvénients ?	78
2.5.4	Les différentes classes de simulation	79
2.5.5	Les niveaux d'abstractions	80
2.6	LES CODES DE LA GENERICITE	82
2.6.1	Le spectre de la généricité.....	83
2.6.2	Générique versus Spécifique.....	84
2.6.3	La réutilisabilité	86
2.7	CONCLUSION	89

2.1 INTRODUCTION

La nécessité d'étude du CDM dans le but de répondre aux nombreux problèmes qui ont été identifiés en son sein, impose de connaître avec exactitude comment celui-ci se comporte vis-à-vis des entités extérieures, mais aussi comment celles qui le composent interagissent entre elles, chose à laquelle il est communément fait référence par le terme « processus ». Ainsi, il est nécessaire de comprendre pourquoi cette idée de processus est si importante et comment elle s'imbrique avec des notions plus générales, telle que celles de modélisation et modèle notamment. De même, identifier la place que cette notion occupe dans le monde hospitalier actuel est une obligation afin de justifier la base de notre travail en mettant en évidence cette nécessité d'étude.

Nous nous intéresserons, en premier lieu, aux définitions des notions – processus, modélisation et modèle – et aux processus en particulier à travers l'étude de leurs différents niveaux et classes. S'en suivra la présentation, la comparaison et l'évaluation de certains langages permettant de les modéliser, dans le but d'en choisir un en particulier. Par la suite, différentes méthodes et méthodologies seront abordées afin de compléter la gamme des outils présentés, celles-ci complétant l'utilisation des langages et permettant l'exploitation des processus.

La simulation de flux offrant une très grande flexibilité dans l'analyse des systèmes étudiés, permet entre autres de caractériser des situations très complexes. Nous nous pencherons donc sur celle-ci à travers ses grands types, ses avantages, inconvénients, etc. afin de la présenter de manière la plus complète possible, avant de terminer cette partie sur un thème central pour nos travaux : la généralité.

2.2 COMMENT CARACTERISER LE CDM ?

Dans le but d'arriver au développement d'un modèle générique du CDM, il va être nécessaire d'étudier et de modéliser soit en partie, soit entièrement les nombreux types de CDM existants ou potentiels. D'autre part, en observant certains textes de lois et obligations légales françaises pour les hôpitaux, il ressort qu'une connaissance approfondie et une représentation complète et détaillée de leurs processus se révèlent nécessaire.

2.2.1 Pourquoi parle-t-on de processus dans le soin ?

Ce terme et la notion associée sont apparus dans un certain nombre de textes de lois et de rapports gouvernementaux tel que l'arrêté du 6 avril 2011. A travers ces textes, il est possible de voir que l'idée même de processus, dans le sens général du terme, s'est ancrée ces dernières années dans le monde hospitalier afin de définir l'enchaînement des activités qui y ont lieu. Voyons tout d'abord comment ces différents textes y font référence (Tableau 2.1).

Dans ce tableau, la colonne « Référence » signifie que le texte parle de processus sans aller plus loin. La suivante qu'il y a une définition du processus dans ce texte, la troisième qu'il y a un travail réalisé pour identifier la composition des processus, puis que ceux-ci sont modélisés, cartographiés et enfin qu'une analyse a été menée à leur rencontre.

Tableau 2.1 : Lien entre documents gouvernementales et processus

Document	Comment chaque texte évoque les processus ?				
	Référence	Définition	Identification	Cartographie	Analyse
Arrêté du 6 Avril 2011 ¹⁴		X			
Rapport sur la sécurisation du CDM à l'APHP ¹⁵ [DAHAN 10]		X	X	X	X
Le circuit du médicament à l'hôpital [CUBAYNES 11]		X	X		
Guide sur la qualité de la prise en charge médicamenteuse [DGOS 12]		X	X	X	
Guichet erreurs médicamenteuses [CASTOT 10]	X				
Rapport National Securimed [DHOS 09]		X			

Il est donc bien constaté que cette notion de processus s'insère depuis plusieurs années dans les textes liés au monde hospitalier et y prend même de plus en plus de place.

Avant de rentrer dans le détail de cette notion, il est impératif de replacer le contexte des processus, ceux-ci ne pouvant exister par eux-mêmes. En effet, la création d'un processus passe par l'étape de « modélisation » et par la construction d'un « modèle », comme nous allons le voir de suite.

2.2.2 Modélisation et modèle !

Tout d'abord, que représente la « modélisation » :

Définition 2.1 : « Une action d'élaboration et de construction intentionnelle, par composition de symboles, de modèles susceptibles de rendre intelligible un phénomène perçu complexe, et d'amplifier le raisonnement de l'acteur projetant une intervention délibérée au sein du phénomène ; raisonnement visant notamment à anticiper les conséquences de ces projets d'action possible. » [LE MOIGNE 77]

D'où la notion de « modèle » comme étant le résultat de son application :

Définition 2.2 : « Un modèle est un élément conceptuel visant à la compréhension et au diagnostic, c'est une vue de l'esprit, analytique ou algorithmique représentant des phénomènes et leurs relations ». [HUET 11]

Les modèles ainsi obtenus peuvent être classifiés et triés et plusieurs sous-catégories existent et les compartimentent. Cependant, un modèle peut appartenir à une ou plusieurs sous-catégories à la fois.

¹⁴<http://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000023865866&dateTexte=&categorieLien=id>

¹⁵ Assistance Publique des Hôpitaux de Paris

↳ Suivant le niveau de détails du modèle :

- ❖ Le modèle de « connaissance » : ce type de modèle est le plus poussé, celui-ci étant issu d'une analyse complète du système, les processus et phénomènes de celui-ci étant parfaitement connus.

Bien que ce type de modèle soit l'objectif de toute analyse, il est souvent difficile d'y parvenir, les systèmes étudiés étant trop complexes, ou trop mal connus. La solution est donc de construire des modèles purement empiriques, qui ne se basent que sur des mesures et des résultats issus des processus.

- ❖ Le modèle « boîte noire » : pour cette catégorie de modèle, le fonctionnement interne du système reste méconnu de manière volontaire ou involontaire, et seules les entrées et sorties seront prises en comptes et étudiées afin d'établir les règles globales le régissant.
- ❖ Le modèle « boîte grise » : ce dernier type est un niveau intermédiaire entre les deux précédents modèles. A ce niveau, seules certaines parties des processus et des phénomènes régissant le système sont connus ce qui permet d'en paramétrer une partie et laisse donc le reste en tant que « boîte noire ».

↳ Suivant l'utilisation qui sera faites du modèle et qui mènera à la création de:

- ❖ Modèles de simulation (ou simulateurs) : utilisé indépendamment du processus représenté, il se doit de posséder un comportement aussi semblable que possible à celui-ci. L'utilisation de tels modèles se justifie dans le cadre de validation pour la conception d'un système (conception d'une chaîne de production, etc.), pour la formation de personnels (simulateurs de pratique), pour la prévision à long terme, etc.
- ❖ Modèles de prédiction (ou prédicteurs) : utilisé en parallèle du processus dont il est le modèle, il prédit la sortie de celui-ci à une échelle de temps courte devant ses constantes de temps.

Permettant donc de représenter la structure et le fonctionnement d'un système, l'atout majeur de la modélisation est de permettre une mise à plat complète de ce qui est étudié afin d'obtenir une meilleure compréhension du système, un meilleur dialogue entre les acteurs, d'évaluer les performances, les dysfonctionnements, de le dimensionner, etc. Découle donc de ces définitions le principe de modélisation du processus qui est de « rendre intelligible un phénomène perçu et d'en amplifier le raisonnement de l'auteur ». Voyons comment cela est possible.

2.2.3 Qu'est-ce qu'un processus ?

Bien que le terme soit en lui-même très usité, obtenir une définition complète et non ambiguë est quelque peu difficile. En effet, les processus ne peuvent se limiter à la simple définition de juxtaposition ou de superposition de différents actes (isolés ou en lien) sur lesquels ils sont basés. La notion de processus permet d'apporter cette idée de flux dynamique représentant un phénomène précis qui parcourt l'ensemble d'un système de son entrée à sa sortie.

Ainsi l'arrêté du 6 Avril 2011 définit un processus comme :

Définition 2.3 : « Toute activité utilisant des ressources et gérée de manière à permettre la transformation d'éléments d'entrée en éléments de sortie, peut être considérée comme un processus. »

Graphiquement la modélisation de cette définition serait la suivante (Figure 2.1).



Figure 2.1 : Modélisation du processus selon la définition ministérielle

Cette définition pose donc les bases de la notion, mais certaines précisions lui font défaut. Elle est cependant très proche de la définition issue de la norme ISO 8402, AFNOR. Celle-ci définit un processus comme :

Définition 2.4 : « L'ensemble de moyens et d'activités liés qui transforment des éléments entrants en éléments sortants. Ces moyens peuvent inclure le personnel, les finances, les installations, les équipements, les techniques et les méthodes. »

Cette idée d'ordonnancement, primordiale, apparaît bien dans la norme AFNOR mais est manquante dans la définition ministérielle, d'autant plus que l'ordre des activités souvent rigide dans le CDM a un fort impact sur celui-ci.

Cette fois-ci, l'autre part importante manquante à ces deux définitions, est qu'un processus a un objectif fixé, spécifié (en général un produit ou un service), cet objectif étant destiné ou orienté vers un bénéficiaire (humain ou non) interne ou externe. Philippe Lorino introduit donc cette notion dans sa définition [[LORINO 95](#)] :

Définition 2.5 : « Ensemble d'activités reliées entre elles par des flux d'information (ou de matière porteuse d'information) significatifs, et qui se combinent pour fournir un produit matériel ou immatériel important et bien défini. »

Celle-ci est même complétée par [[LORINO 98](#)] où il est précisé qu'un processus existe « pour produire un résultat ou output ayant de la valeur pour un client externe. »

Ainsi, la définition d'un processus peut être vue d'une manière plus générale, comme une suite ordonnée d'activités (répétitives ou non) et de moyens mis en œuvre (ressources mécaniques, humaines, informatiques, etc.) pour atteindre un objectif.

La modélisation de cette définition d'un processus pourrait être la suivante (Figure 2.2) :

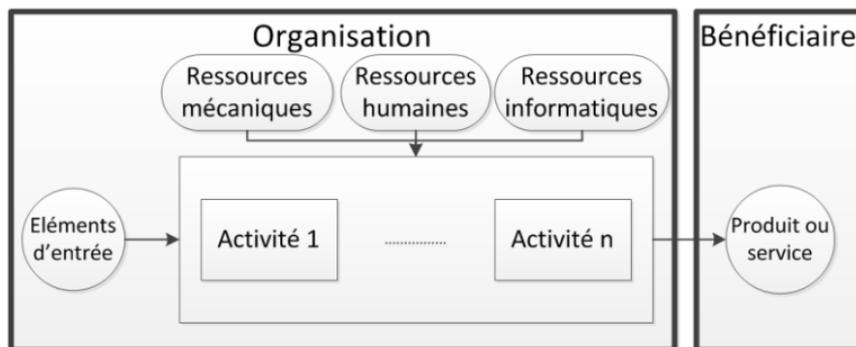


Figure 2.2 : Modélisation de notre définition d'un processus

Pour terminer et afin de compléter les informations sur la notion de ressources utilisées, il est important de signaler que les activités sont réalisées par différentes fonctions ou métiers, ce qui rajoute au processus l'idée d'interactivité et de corrélation entre les acteurs.

2.2.4 L'approche par processus en quelques mots

Le processus étant l'élément de base, cette approche permet de s'affranchir des frontières inhérentes à une organisation hiérarchique. En effet, elle apporte une vision transversale, indispensable aux organisations traditionnellement verticales comme l'hôpital où le dialogue interservices est souvent inopérant, et permet d'améliorer les relations entre les différentes composantes de l'organisation.

D'après la norme ISO 9001:2008, l'approche processus consiste en l'application d'un système de processus au sein d'un organisme, ainsi que l'identification, les interactions et le management de ces processus en vue d'obtenir le résultat souhaité. L'un des avantages de l'approche processus est la maîtrise permanente qu'elle permet sur les relations entre les processus individuels au sein du système de processus, ainsi que sur les combinaisons et interactions. Lorsqu'elle est utilisée dans un système de management de qualité, cette approche souligne l'importance de : comprendre et de remplir les exigences ; de considérer les processus en terme de valeur ajoutée ; de mesurer la performance et l'efficacité des processus ; d'améliorer en permanence les processus sur la base de mesures objectives.

2.2.5 Modélisation et granularité des processus

Ces deux notions sont interdépendantes dès que l'on s'intéresse à l'étude des processus. La granularité permet de définir à quels niveaux (microscopique, mésoscopique, macroscopique), il sera nécessaire de se placer pour pouvoir étudier de la manière la plus adéquate et performante possible notre processus.

La modélisation comme déjà expliquée rend possible l'identification des rouages (activités, sous-processus, processus) et des interfaces entre ces rouages, dans le but de mettre en évidence les liens opérationnels entre les entités d'entrée et celles de sortie. Mais jusqu'où le niveau de détails doit-il être porté ?

Intéressons-nous à cette granularité. Chacun de ces niveaux peut être résumé comme suit :

- ❖ Le niveau macroscopique peut être assimilé à une vue agrégée du système, considérant donc l'entièreté de celui-ci sans se concentrer précisément sur une portion ou un flux en

particulier. Ce niveau fait opposition au niveau microscopique comme symbolisant ce qui est observable à l'œil nu ;

- ❖ Le niveau microscopique permet quant à lui de se concentrer sur les interactions individuelles de chaque élément, ceux-ci étant vus comme des entités atomiques indécomposables. C'est ce niveau qui est le plus adéquat pour la simulation, celui-ci permettant de gérer avec précision l'affectation de ressources (disponibilité, zone d'activité, etc.) ou de donner la possibilité d'obtenir des indicateurs très précis (charge de travail, affectation, etc.) ;
- ❖ Le niveau mésoscopique pour terminer, est un intermédiaire entre les deux précédents, permettant d'agréger une ou plusieurs entités du niveau microscopique dont des caractéristiques sont communes, et de former ainsi des sous-groupes de la vue macroscopique, ce niveau permet d'obtenir une évaluation des performances du système satisfaisante sans pour autant devoir rentrer dans une étude approfondie de celui-ci.

Les trois termes utilisés précédemment permettent donc d'aborder l'analyse sous un angle différent, selon le niveau voulu, mais aussi suivant les objectifs et les problématiques rencontrés.

Dans [RODIER 10], ces objectifs se synthétisent en trois principaux :

- ❖ Concevoir ou adapter le système en y dimensionnant entités et flux le composant ;
- ❖ Configurer le système en planifiant les flux, les activités et les ressources ;
- ❖ Piloter le système en évaluant ses performances afin de pouvoir ajuster et optimiser son fonctionnement.

Ces objectifs résumés avec les niveaux de granularité sont présentés dans le Tableau 2.2.

Tableau 2.2 : Matrice Objectifs / Granularité [Rodier 10]

	Macroscopique	Mésoscopique	Microscopique
	<i>Vue globale</i>	<i>Vue processus</i>	<i>Vue activité</i>
Concevoir	Dimensionner		
	Systeme	Processus	Activités
Configurer	Planifier		
	Systeme	Processus	Activités
Piloter	Evaluer et ajuster		
	Systeme	Processus	Activités

Ces trois objectifs se trouvent cependant plus adaptés dans le cadre de l'élaboration d'un système et de son modèle. Etant pour notre part dans un contexte où le modèle servira en premier lieu d'outil de dialogue, les objectifs et la matrice précédente sont à modifier :

- ❖ Comprendre et analyser le fonctionnement du système par l'identification des éléments clés, des rouages afin de pouvoir en débattre ;
- ❖ Evaluer ses performances et configurer le système par le biais d'un contrôle du système et sur le système en suivant les choix faits en amont ;

- ❖ Adapter, améliorer le système existant en le restructurant et en modifiant son fonctionnement de base.

Tableau 2.3 : Matrice Objectifs / Granularité

	Macroscopique	Mésoscopique	Microscopique
	<i>Vue globale</i>	<i>Vue processus</i>	<i>Vue activité</i>
Comprendre & Analyser	Identifier		
	Système	Processus	Activités
Evaluer & Configurer	Contrôler		
	Système	Processus	Activités
Adapter & Améliorer	Restructurer		
	Système	Processus	Activités

Dans la logique aucune règle n'est prédéfinie, et le choix est donc laissé aux utilisateurs en fonction des objectifs fixés, des processus étudiés. Comme dit plus haut, il faut se placer de manière adéquate et performante, ce qui implique de ne pas tomber dans des excès de détails qui pourraient nuire à la compréhension et à la mise en pratique des processus. La précision « nécessaire et suffisante » sera donc l'objectif visé pour permettre d'étudier et de maîtriser le système. L'utilisation d'indicateurs et de curseurs à l'intérieur même de cette modélisation, la rendant alors dynamique, permettra de valider les choix de granularité précédemment faits, en constatant lors de cette étape de mise en pratique qu'un processus n'est alors pas suffisamment ou trop détaillé.

2.2.6 Les différents niveaux des processus

La granularité de la modélisation des processus, se traduit au niveau de celui-ci par un changement de dénomination et de zone d'étude. Ainsi, nous proposons une décomposition à 7 niveaux de détails, permettant d'identifier les processus du plus général au plus précis (Tableau 2.4).

Tableau 2.4 : Les différents niveaux de processus

Niveau de vue	Libellé du niveau de détails	Processus
Macroscopique	Méga-méga-processus	<i>Représentation synthétique du système et des domaines d'activité</i>
	Méga-processus	<i>Représentation de l'ensemble des processus d'un domaine d'activité</i>
Mésoscopique	Processus	<i>Représentation d'un processus de ce domaine</i>
	Sous-processus	<i>Représentation d'une partie de ce processus</i>
Microscopique	Activité	<i>Ensemble d'opérations possédant un lien logique et composant avec d'autres un sous-processus</i>
	Opération	<i>Ensemble de tâches réalisées par un acteur donné</i>
	Tâche	<i>Action élémentaire réalisée par un acteur donnée dans un temps et un lieu précis</i>

L'utilisation de cette échelle de décomposition ne doit pas se faire nécessairement dans son entièreté. Les 7 niveaux présents ne servent dans leur globalité que si les processus à décomposer sont très agrégés et nécessitent un haut degré de précision dans leur modélisation.

Pour illustrer cette décomposition, utilisons un exemple du CDM :

Tableau 2.5 : Exemple des différents niveaux de processus

Niveau de vue	Libellé du niveau de détails	Zone d'étude	Acteur(s)	Processus
Macroscopique	Méga-méga-processus	<i>Hôpital</i>	<i>Infirmier(e), médecin, brancardier, etc.</i>	<i>Soigner un patient</i>
	Méga-processus	<i>Thérapie médicamenteuse</i>	<i>Infirmier(e), médecin, ASH, interne, etc.</i>	<i>Prise en charge médicamenteuse</i>
Mésoscopique	Processus	<i>Pharmacie</i>	<i>Préparateur, interne en pharmacie, pharmacien, etc.</i>	<i>Dispensation du traitement</i>
	Sous-processus		<i>Pharmacien, interne en pharmacie</i>	<i>Contrôle du traitement</i>
Microscopique	Activité		<i>Pharmacien, interne en pharmacie</i>	<i>Contrôle des médicaments dangereux</i>
	Opération			<i>Contrôle de la posologie</i>
	Tâche			<i>Cliquer sur le bouton de validation</i>

Afin de terminer le tour d'horizon des caractéristiques des processus, intéressons-nous désormais aux grandes classes de processus qui existent.

2.2.7 Les différentes classes de processus

Il est possible de tirer de la norme ISO 9001 trois grandes classes de processus :

- ❖ Les processus opérationnels ou de réalisation ;
- ❖ Les processus dits de « support » ;
- ❖ Les processus de direction ou de pilotage (relatifs aux activités de management).

Cependant un type de processus n'est pas figé par définition. Il peut être support pour une organisation et opérationnel pour une autre. La nature du processus dépend donc de l'activité de l'organisation. L'exemple suivant illustre cela (Figure 2.3). Dans cet exemple l'activité de conseil est donc le cœur de métier de la société de conseil, dispensatrice d'un service, contrairement au constructeur automobile où cette activité de conseil n'est qu'un support.

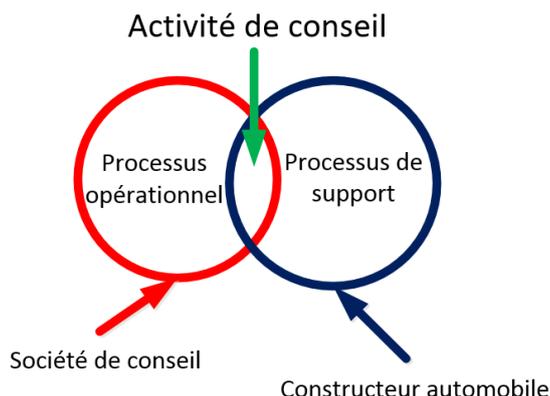


Figure 2.3 : Exemple d'un processus commun à deux organisations d'activités différentes

Intéressons-nous plus en détail à chaque classe :

- ❖ Les processus opérationnels : ceux-ci contribuent à la réalisation du produit ou du service. Ces processus font référence à l'activité même de l'organisation. Plus généralement, ils regroupent toutes les activités qui touchent le service et le produit. Dans le cadre d'une pharmacie, il s'agirait par exemple de la réception, de l'analyse des ordonnances, de la préparation des demandes, etc., ces processus étant eux-mêmes décomposables en sous processus.
- ❖ Les processus supports : ils concernent les activités contribuant au bon fonctionnement des autres processus en leur fournissant les ressources nécessaires à leur bon déroulement. Généralement transversaux (les ressources humaines par exemple), ils ne créent pas de valeur ajoutée (perceptible par le bénéficiaire du processus) au produit ou au service. Les exemples typiques de cette famille sont l'entretien des infrastructures, du matériel, les systèmes d'information ou dans le cadre d'une pharmacie, la gestion des prescriptions ou les demandes de réapprovisionnement.
- ❖ Les processus de pilotage : étant connus aussi sous le nom de « processus de direction », ils sont essentiels au bon fonctionnement de l'organisation en définissant une ligne conductrice et les actions à mettre en œuvre pour mener à bien la politique établie. Ils sont donc sous la responsabilité de l'équipe dirigeante et liés aux business plans, aux budgets, à la stratégie, etc. De surcroît ils assurent la cohérence des processus opérationnels et de support en les pilotant et en les surveillant.

2.3 QUELS LANGAGES POUR LA CARTOGRAPHIE ?

L'idée de processus étant mieux cernée, intéressons-nous au comment et avec quels outils les processus peuvent être cartographiés. L'intérêt est ici de trouver un langage qui permet de décrire tous les types de processus vus ci-dessus et qui doit être capable de s'adapter et de traduire chaque niveau de modélisation. De surcroît, comme il le sera vu plus tard, une difficulté supplémentaire sera de choisir le bon langage permettant de modéliser ce que l'on souhaite et d'utiliser ceci pour communiquer avec le personnel hospitalier.

Depuis de nombreuses années maintenant, de nombreux langages ont été mis au point pour aider à la modélisation et la cartographie des processus. Certains d'entre eux se basent sur ce qui existent déjà, alors que d'autres ont apporté de véritables révolutions. Nous en étudierons dans ce

qui suit cinq différents, chacun d'entre eux possédant une orientation propre. Nous commencerons par les plus anciens puis remonterons chronologiquement par ordre de création, en donnant un exemple à chaque fois. L'exemple utilisé sera la réalisation d'une prescription par un médecin dans un service. Ce processus se situe à un niveau mésoscopique et fait partie de la classe des processus opérationnel. L'axe suivant détaille cet ordre d'apparition (Figure 2.4) :

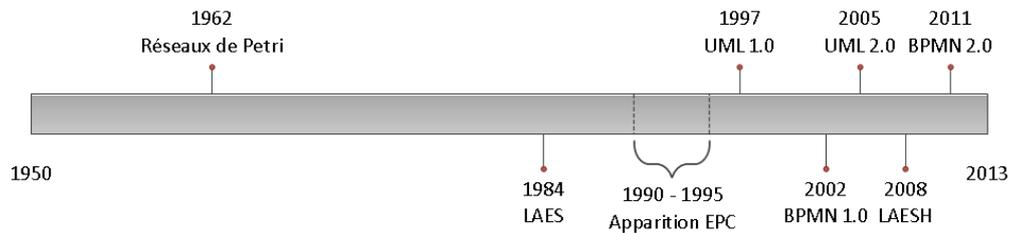


Figure 2.4 : Axe chronologique de l'apparition des langages

2.3.1 Les langages choisis

2.3.1.1 Les Réseaux de Pétri (RdP)

Ces réseaux sont des modèles mathématiques permettant de représenter de nombreux types de systèmes par la description des relations qui existent entre les événements et les conditions qui les composent et les régissent. La logique de ce langage est des plus simples, celle-ci se basant sur l'idée de « jetons » permettant d'autoriser le passage d'une situation à une autre si les conditions sont remplies (ce qui sous-entend d'avoir le nombre de jetons nécessaires). Ces réseaux sont composés de places (événements) et de transitions (conditions). Ce langage est très simple, mais la nécessité d'alternance place/transition peut très vite faire exploser la taille du diagramme. Il a l'inconvénient de présenter une sémantique assez pauvre, dû à sa simplicité de représentation (Figure 2.5).

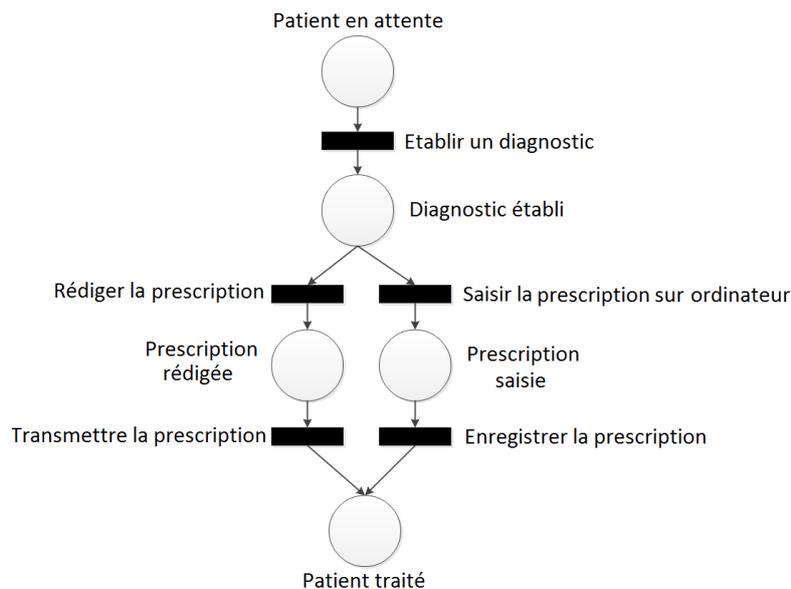


Figure 2.5 : Exemple de Réseaux de Pétri

2.3.1.2 Event-driven Chain Process (EPC)

Ce type de langage (Figure 2.6) s’oriente vers la modélisation des processus métier, et s’utilise notamment dans le cadre de la conception d’ERP (Entreprise Ressource Planning), tel que SAP. Tout comme le précédent, EPC utilise des pictogrammes pour se construire et suit une règle identique qui est l’alternance de blocs « action » (évènement) et de blocs « situation » représentant l’état du système ou d’une entité à cet instant précis. Il permet de plus, de par sa structure, d’unifier dans un même diagramme les informations transmises dans le système, les ressources utilisées, les fonctions, etc. chose que l’on ne retrouvera que dans les deux derniers langages (BPMN et LAESH).

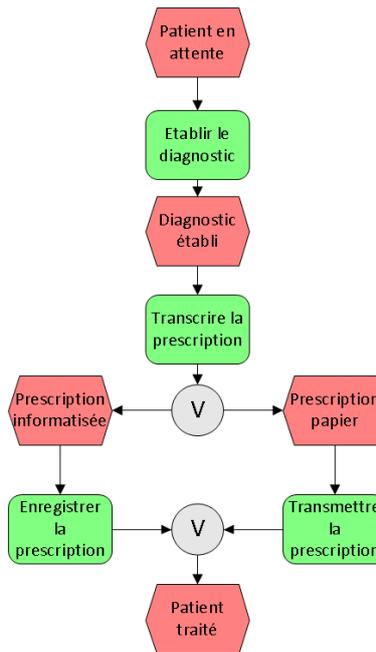


Figure 2.6 : Exemple de diagramme EPC

2.3.1.3 « Unified Modeling Language » (UML)

Issu du monde du génie logiciel pour la conception orientée objet, UML dispose de nombreux formalismes à bases de pictogrammes pour faire de la modélisation graphique. Dans notre cas, seul le diagramme d’activité se prête à notre problème de cartographie des processus, celui-ci permettant de décrire entre autres les cas d’utilisation du système, ce que nous assimilerons à une fonction métier présente dans celui-ci. Facilement compréhensible du fait de sa représentation sous forme d’organigramme, ce diagramme se concentre sur ce que voient les acteurs qui collaborent avec le système dans le cadre d’un processus métier. Il est composé de blocs « action » (Figure 2.7).

2.3.1.4 Business Process Model and Notation (BPMN)

Tout comme EPC, BPMN a pour but de modéliser les processus métier. Il a pour fonction première de fournir une notation standardisée compréhensible par tous les acteurs d’une entreprise. Basé lui aussi sur la simple logique d’un enchaînement de blocs « action » décrivant les processus et procédures d’une organisation, il a l’avantage d’incorporer aussi des évènements représentant une palette très large de différents signaux (erreur, temps, arrêt, etc.) mais aussi des acteurs et des données (Figure 2.8).

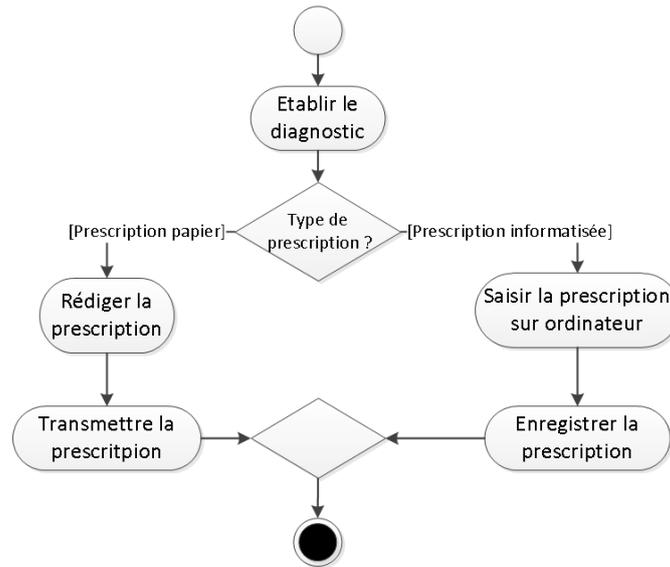


Figure 2.7 : Exemple de diagramme UML

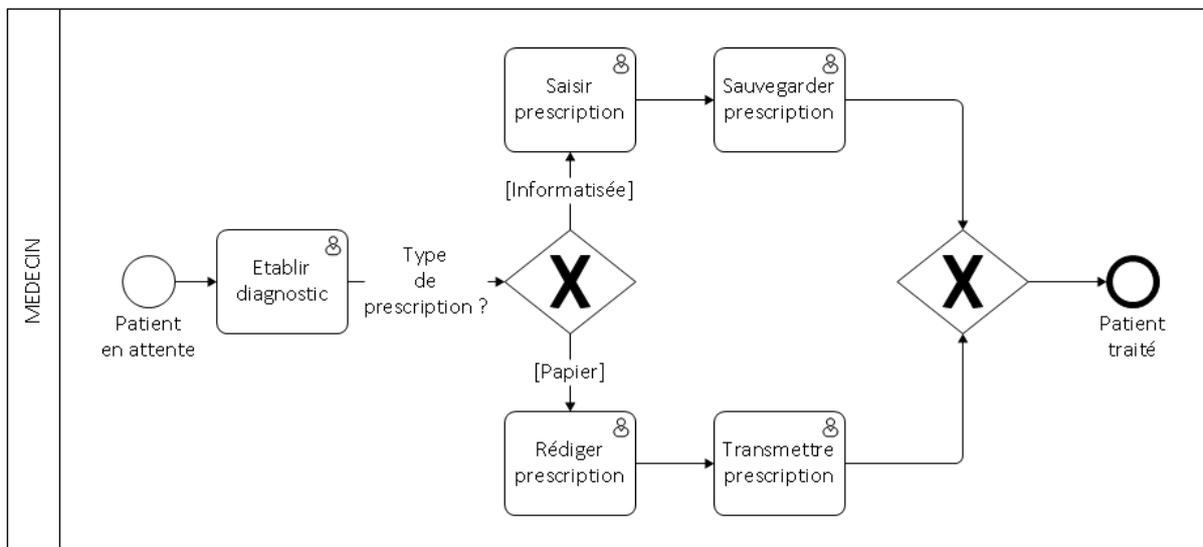


Figure 2.8 : Exemple de diagramme BPMN

2.3.1.5 Langage d'Analyse et d'Etude des Systèmes Hospitaliers (LAESH)

Lui aussi basé sur des pictogrammes, celui-ci permet de représenter à deux niveaux de détails (une vue globale et une vue détaillée) les processus en se basant sur trois concepts : la catégorie, la phase et le chemin. Il est issu du langage LAES pour modéliser les systèmes complexes et présente l'avantage d'avoir une vue globale très simple (Figure 2.9 (a)) à réaliser et à comprendre. Sa vue détaillée quant à elle, permet l'intégration des ressources et d'autres informations de manière très poussée. (Figure 2.9 (b)). Quelques informations pour la bonne compréhension de ce langage sont nécessaires. Le terme « EXEC » signifie qu'une tâche est réalisée par un acteur ici « 1 » correspondant au médecin en un temps « Ti ». Le dernier chiffre correspondant à une possibilité de mise en attente de l'action, seulement possible ici dans le cas où la prescription est transmise au format papier.

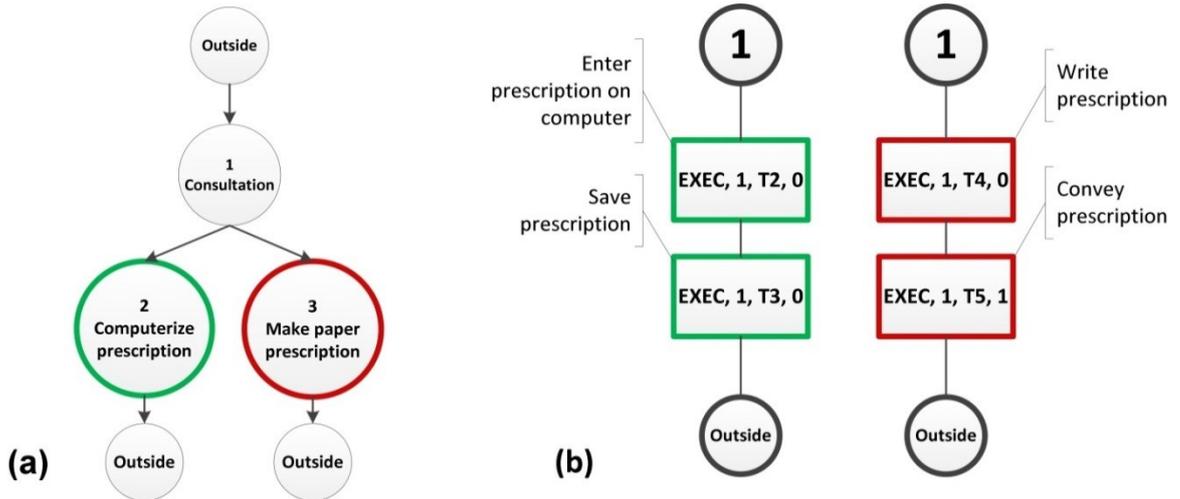


Figure 2.9 : (a) Vue globale LAESH ; (b) Vue détaillée LAESH

2.3.2 Evaluation des langages

Chaque langage étant succinctement présenté pour voir lequel serait le plus adapté au CDM, il a été nécessaire d'évaluer chacun d'entre eux suivant des critères définis. Deux lignes d'études sont à suivre. Une première concernant l'aspect humain, c'est-à-dire comment les utilisateurs travailleront avec, et l'autre qui concerne l'aspect technique du langage. Du premier il y aura donc un questionnaire sur la simplicité d'utilisation, la lisibilité, etc. de celui-ci. Du second, il faudra s'interroger sur des questions concernant la prise en compte du temps dans sa représentation, de la capacité de diffusion de ce langage, etc.

Afin d'éviter toute ambiguïté détaillons chaque critère :

☞ Aspect humain :

- ❖ Lisibilité : un processus déjà modélisé est-il facilement compréhensible ? Y-a-t-il suffisamment d'informations fournies par le langage pour comprendre le rôle de chaque élément ?
- ❖ Capacité d'expression : les éléments du langage permettent-ils de décrire un nombre conséquent de situations différentes ?
- ❖ Facilité de représentation : y a-t-il un nombre important de contraintes dans l'utilisation des éléments du langage : enchaînement des éléments, disposition, etc. ?
- ❖ Simplicité d'utilisation : le langage est-il composé d'un nombre important d'éléments à utiliser ?

☞ Aspect technique :

- ❖ Représentabilité globale : une vue globale du niveau microscopique du modèle sera-t-elle compréhensible ?
- ❖ Prise en compte de l'aspect temporel : le langage permet-il d'intégrer une représentation du temps ?
- ❖ Capacité de diffusion : le langage est-il disponible dans un nombre important de logiciels afin que les utilisateurs puissent facilement l'utiliser et communiquer avec ? Existe-t-il un

logiciel commun permettant de faire cette modélisation ? Celle-ci nécessite-t-elle beaucoup de travail ?

Bien que de nombreux autres critères auraient pu être retenus, nous nous sommes limités à ceux-ci pour éviter de complexifier le choix et de ne pas rallonger la période d'évaluation des langages par les praticiens. En effet il leur a été demandé d'évaluer chacun des langages proposés, le but étant à terme, que celui choisi soit utilisé par les pharmaciens et d'autres personnes travaillant sur le CDM.

L'intérêt majeur est donc ici de connaître la position et le ressenti de personnes n'ayant jamais eu ce genre d'expérience. Ainsi, il a fallu trouver un exemple simple d'application qui leur serait rapidement appréhendable. Le choix s'est porté sur la validation des prescriptions par les pharmaciens. Ceci a nécessité de fixer le cadre de cet exemple :

- ❖ Evènement déclencheur : nouvelle prescription ;
- ❖ Question à se poser : y a-t-il une analyse des prescriptions ? La dernière prescription est-elle validée ?
- ❖ Etapes : analyse, dialogue avec le médecin et validation de l'ordonnance ;
- ❖ Acteur : Pharmacien/Interne ;
- ❖ Evènement final : toutes les prescriptions sont validées ;

Une fois chaque langage expliqué, il leur a été fourni un exemple ainsi qu'un fichier d'aide pour chacun, nous leur avons demandé de modéliser la validation des prescriptions. Une fois ceci terminé, une discussion eut lieu avec eux pour voir s'ils avaient compris les langages et si leurs modélisations étaient correctes. Durant cette discussion, il leur a été demandé de noter suivant les quatre critères humains précédemment énoncés ces différents langages :

- ❖ Lisibilité ;
- ❖ Capacité d'expression ;
- ❖ Facilité de représentation ;
- ❖ Simplicité d'utilisation.

Chaque critère a donc été noté entre zéro et cinq, et les résultats obtenus sont présentés dans le Tableau 2.6.

Pendant avant de tirer des conclusions de cette évaluation, hiérarchiser ces différents critères serait pertinent pour procéder de la manière la plus efficace possible. Ainsi la méthode AMCH (Analyse Multi Critère Hiérarchique) de **Thomas.L Saaty** a été utilisée. Celle-ci a permis de pondérer les différents critères pour obtenir la classification suivante :

- 1 – Représentativité globale/Prise en compte du temps ;
- 2 – Capacité d'expression ;
- 3 – Lisibilité ;
- 4 – Capacité de diffusion ;
- 5 – Facilité de représentation ;
- 6 – Simplicité d'utilisation.

Tableau 2.6 : Notation des différents langages

Critères	Langages						
	RdP	EPC	UML Act.	BPMN	LAESH Globale	LAESH Détaillé	
Critères humains							
Lisibilité	2	3	4	4	5	1	
Capacité d'expression	2	4	3	5	1	5	
Facilité de représentation (contraintes)	3	3	3	2	5	1	
Simplicité d'utilisation	4	3	4	2	5	1	
Critères techniques							
Représentabilité globale	1	1	3	4	5	2	
Prise en compte de l'aspect temporel	0	1	3	4	0	5	
Capacité de diffusion	3	2	4	3	5	1	

Les notes obtenues pour chaque langage sont les suivantes :

Tableau 2.7 : Note obtenue pour chaque langage

Langage	RdP	EPC	UML Act.	BPMN	LAESH Globale	LAESH Détaillé
Note	1.64	2.09	3.29	3.69	3.24	2.71

2.3.2.1 Conclusion

Il apparaît donc que BPMN serait le langage qui répondrait le mieux aux différents critères avec en second choix UML. Ce constat est assez logique, les deux langages étant assez proches. Il s'agissait aussi globalement du constat fait après la présentation des langages aux praticiens qui mettaient en avant ces deux-là.

BPMN (tout comme UML) ne sont pas pour autant exempts de défauts et ceci se voit notamment au niveau macroscopique où les deux manquent de quelques aménagements pour être plus expressifs et lisibles. Cependant un tel travail peut être réalisé en complément du langage existant et ne nécessite pas pour autant de refondre ou de modifier en profondeur ces langages.

En choisissant BPMN comme langage de modélisation, il est donc désormais possible d'avancer dans la démarche, en se penchant sur les méthodes et méthodologies.

2.4 METHODES & METHODOLOGIES

Un langage permet de spécifier et d'analyser le fonctionnement d'un système complexe mais il n'est pas toujours incorporé dans une méthode ou méthodologie sur lequel elles s'appuient. Ainsi, il est possible, soit de se limiter aux méthodes et méthodologies telles qu'elles ont été conçues, soit

d'adapter celles-ci afin de pouvoir incorporer des langages qui sont apparus à la suite de leurs créations et qui n'ont pas été nécessairement pensés pour s'y adapter.

2.4.1 Les méthodologies de modélisation

Une méthodologie de modélisation fournit un guide général sur lequel se baser mais sans pour autant donner les outils qui devront être utilisés. Elle peut être définie comme suit :

Définition 2.6 : « Une méthodologie de modélisation est un ensemble de méthodes, d'outils, d'approches et de concepts permettant de modéliser un système » [CHAUVET 09].

Elle a donc pour rôle de combiner des approches, des langages ou des méthodes issus d'horizons différents afin de les intégrer dans une démarche logique.

Une notion supplémentaire de notre point de vue devrait être rajoutée pour rendre complète cette définition. Il s'agit de la « **reproductibilité** » de cette méthodologie. En effet l'intérêt d'une telle démarche serait alors de pouvoir être répétée si nécessaire pour mener plus loin une étude, ou d'être applicable à d'autres systèmes afin de les étudier.

2.4.1.1 La méthodologie GRAI-GIM

La méthodologie « GRAI Integrated Methodology » (**GIM**) se base sur la méthode du même nom GRAI (Graphes à Résultats et Activités Inter-relies) mises au point toutes les deux à l'université de Bordeaux [DOUMEINGTS 84][MARCOTTE 94]. Cette méthodologie s'applique dans une optique similaire à celle de la méthode GRAI, c'est-à-dire l'amélioration des performances du système étudié.

Son cycle de vie suit une logique évolutive – Modélisation de l'existant, Diagnostic, Conception du système cible – et couvre un plus grand nombre de sous-systèmes (physique, décisionnel et information) que la méthode GRAI qui se concentre sur le décisionnel. Les Figures 2.10 et 2.11¹⁶ illustrent l'étendue des domaines couverts par cette méthodologie.

La méthode GRAI sera quant à elle présentée dans la partie suivante concernant les méthodes de modélisation.

2.4.1.2 La méthodologie ASCI

ASCI signifiant *Analyse, Spécification, Conception, Implémentation* est une méthodologie développée dans le but de permettre la conception d'outils d'aide à la décision appelés dans le cadre de celle-ci modèles d'actions (modèles de simulation, modèles métaheuristiques, modèles analytiques, etc.) [GOURGAND 91, GOURGAND92]. Cette méthodologie permet donc de concevoir la modélisation d'une classe de systèmes, le modèle générique de connaissance (noté **MGC**) de cette classe, et de construire une bibliothèque de composants logiciels qui est utilisée pour générer un modèle d'action (équivalent à un programme informatique) (Figure 2.12). Le cadre conceptuel de la méthodologie ASCI repose sur un processus de modélisation séparant explicitement le recueil des connaissances de son exploitation. Chaque étape impose donc de choisir les outils les plus adéquats en se basant sur les caractéristiques du système, les objectifs visés, etc. Enfin, la vision

¹⁶ Issu de : <http://www.easy-dim.org/emea/emea-2012/Vallespir%20GRAI.pdf/>

systemique traduisant l'application d'ASCI consiste en la décomposition hiérarchique du système avec le paradigme objet en trois sous-systèmes interdépendants (communicants et complémentaires). Il s'agit des sous-systèmes logique (SSL), physique (SSP) et décisionnel (SSD).

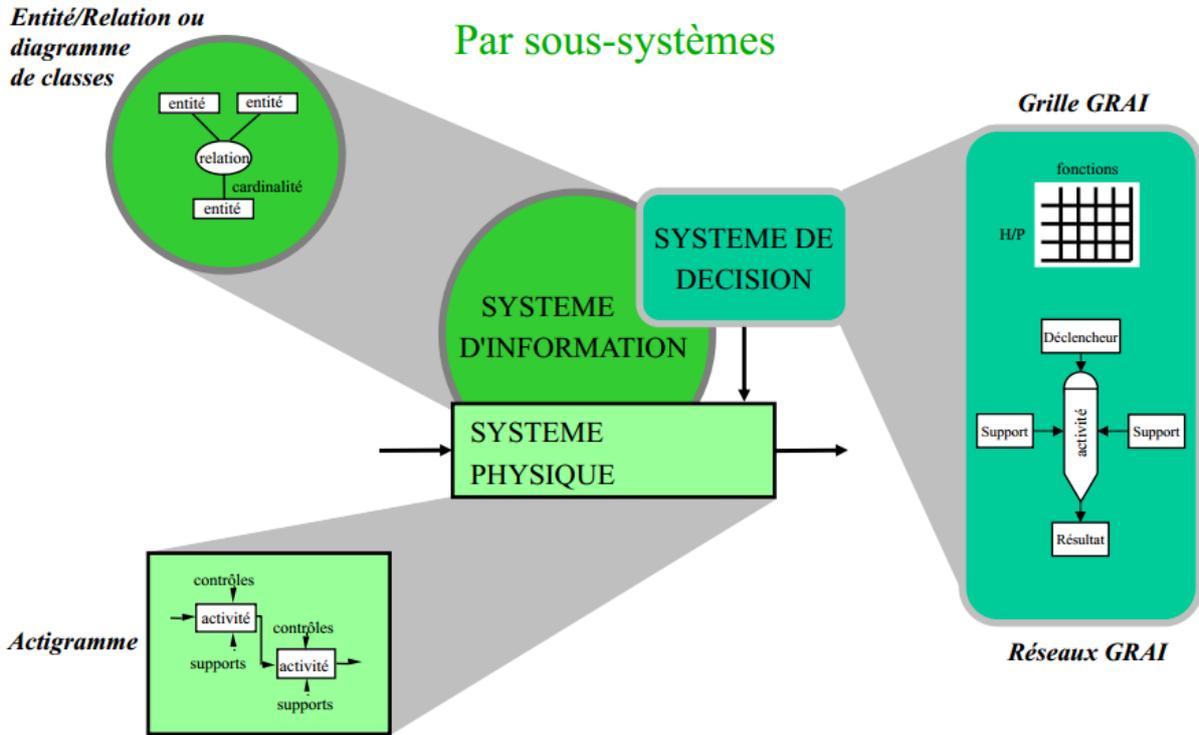


Figure 2.10 : Couverture de modélisation de la méthodologie GRAI-GIM

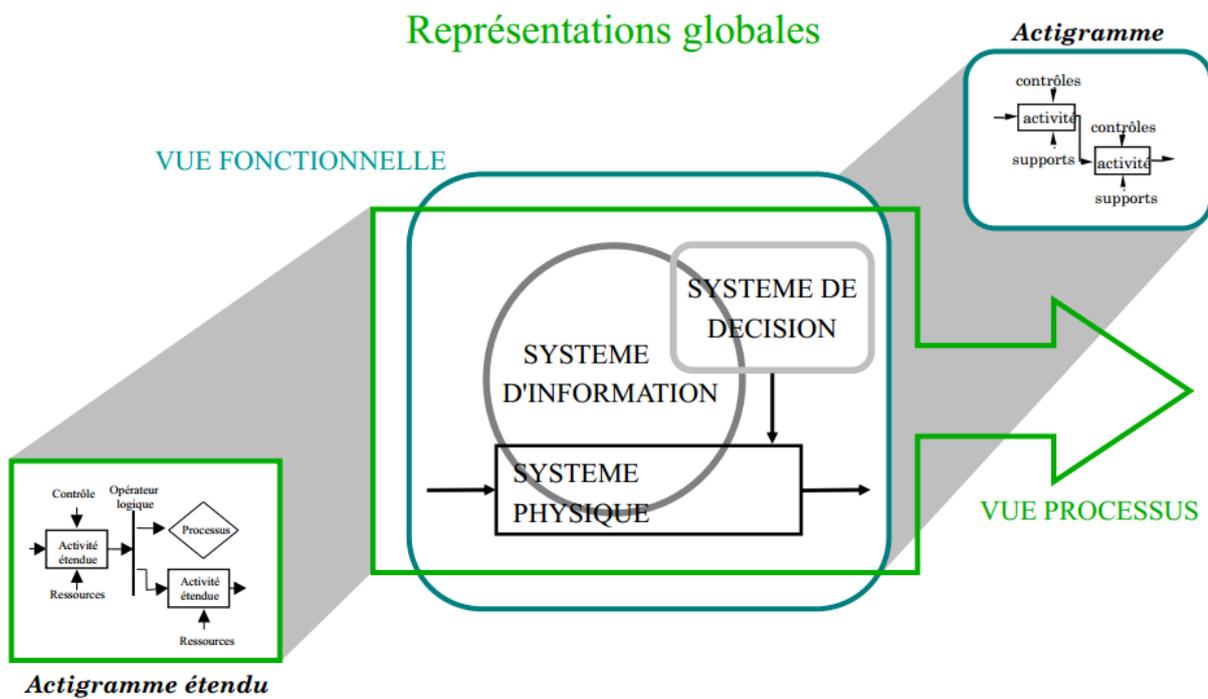


Figure 2.11 : Vue synthétique basé sur GRAI-GIM

Nous nous arrêterons ici quant aux méthodologies, notre choix se portant sur ASCI, celle-ci ayant déjà été utilisée avec succès dans le domaine hospitalier ([HUET 10], [RODIER 10]) du fait de sa grande flexibilité et adaptabilité.

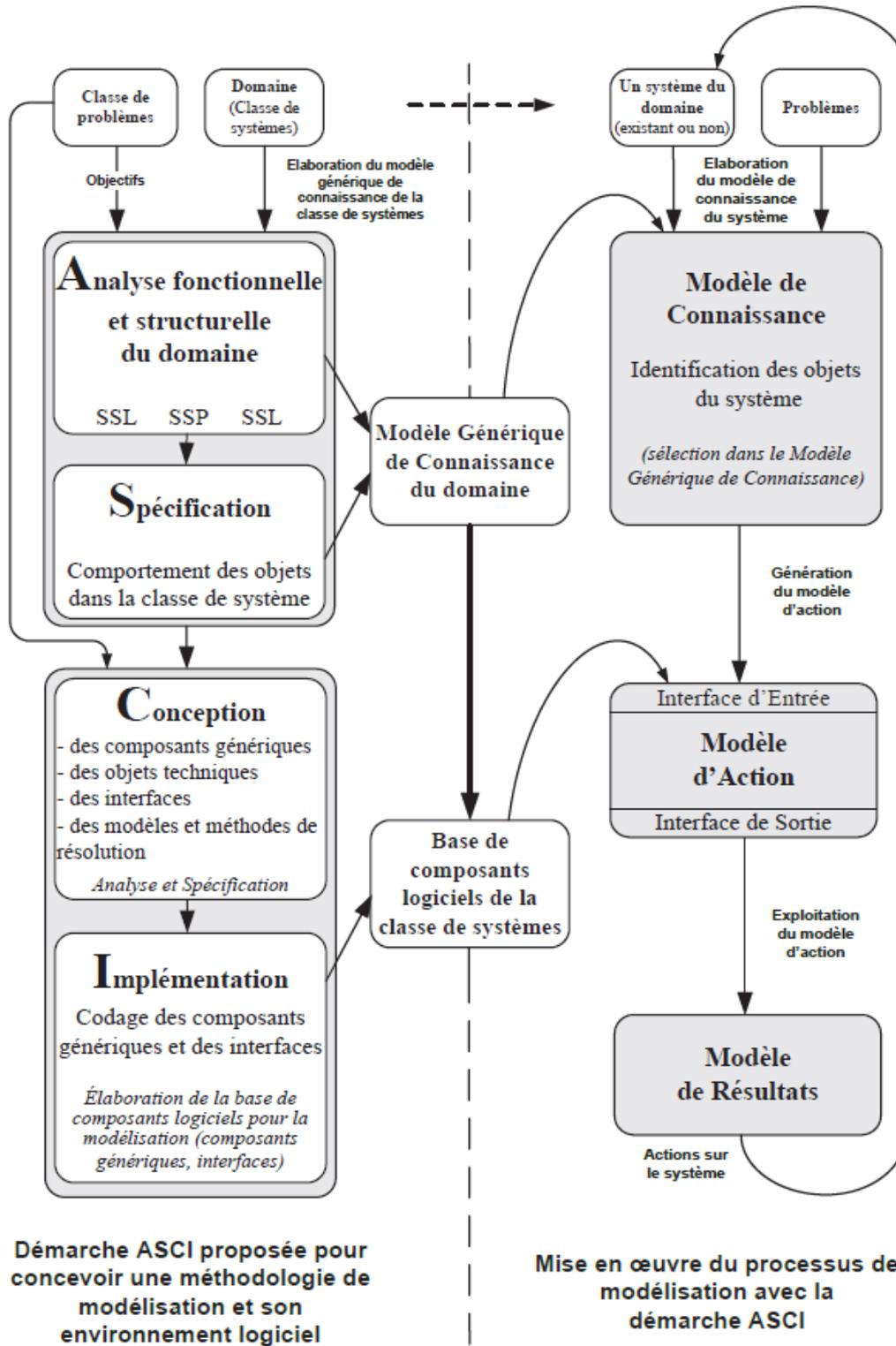


Figure 2.12 : La méthodologie ASCI

2.4.2 Les méthodes de modélisation

Une méthode de modélisation explique comment concevoir et structurer un modèle de connaissance d'un système à partir d'un formalisme donné et d'une ou plusieurs approches.

Définition 2.7 : « Une méthode est une technique de résolution de problèmes caractérisée par un ensemble de règles bien définies qui conduisent, pour le problème, à une résolution correcte » [CALVEZ 90].

De plus, d'après [PIERREVAL 90], une méthode doit permettre de :

- ❖ Guider l'analyste dans sa démarche ;
- ❖ Faciliter la compréhension du système étudié ;
- ❖ Diagnostiquer les anomalies du système étudié ;
- ❖ Communiquer avec les acteurs concernés relevant de différents domaines ainsi que le concepteur avec l'utilisateur.

Dans ce qui va suivre, les méthodes présentées seront SADT/IDEF_x, MERISE et OSSAD.

2.4.2.1 La méthode GRAI

Comme dit précédemment, cette méthode, basée sur la modélisation d'entreprise, s'axe sur l'amélioration des performances en se concentrant sur la partie décisionnelle du système étudié afin de concevoir ou de repenser les systèmes de production le composant.

Cette méthode suit une logique s'appuyant sur 3 axes qui sont :

- ❖ Le modèle conceptuel de référence qui définit la structure conceptuelle de référence du système de production de toute entreprise industrielle, de services ou de toute organisation suivant trois angles différents : l'angle gestion de production (système d'information), l'angle structure de production (système physique) et l'angle structure décisionnelle (système décisionnel) ;
- ❖ La grille GRAI qui découpe de manière fonctionnelle la structure décisionnelle périodique du système de pilotage (gestion des ressources, de la planification de productions, etc.) ;
- ❖ Les réseaux GRAI qui reprennent les centres de décisions identifiés dans la grille et représentant le fonctionnement (partiel ou global) de ces centres suivant des modèles d'activités comprenant des activités de décision ou d'exécution.

Voyons le déroulement de la méthode GRAI. La Figure 2.13 permet de voir les grandes étapes ainsi que les actions à entreprendre à chaque niveau. A noter que l'ordre de ces différentes étapes n'est pas rigide. Il est possible que certaines d'entre elles se chevauchent et qu'il y ait des boucles dans leur utilisation. Les différents outils de la méthode GRAI apparaissent sur le côté droit et il y est fait référence par les termes « grille » et « réseaux ».

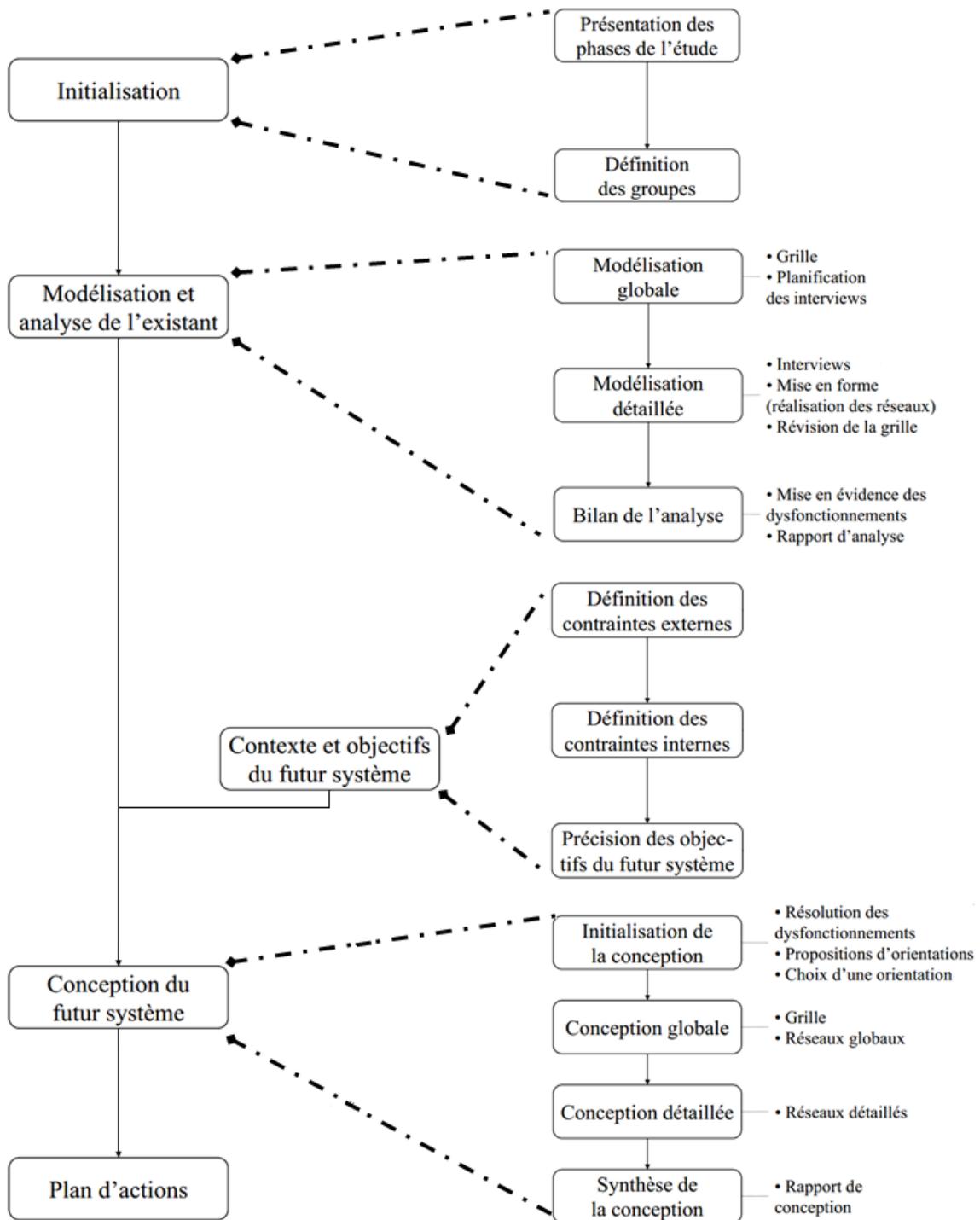


Figure 2.13 : Déroulement de la méthode GRAI

2.4.2.2 La méthode IDEF-SADT

Le langage « *Integrated computer aided manufacturing DEFinition language* » (IDEF) a été mis au point durant les années 1970 par l'US Air Force [MAYER 92] et s'est rapidement popularisé avec notamment son utilisation au sein du Département de la Défense américain. A partir de ce langage, différentes méthodes ont été construites pour couvrir un grand nombre de domaines d'application et permettre ainsi de les modéliser. Ces différentes méthodes sont :

- ❖ IDEF0 décomposition des activités ;
- ❖ IDEF1 flux d'informations ;
- ❖ IDEF1x structures de données ;
- ❖ IDEF2 simulation ;
- ❖ IDEF3 description des processus ;
- ❖ IDEF4 conception orientée objet ;
- ❖ IDEF5 description d'ontologies ;
- ❖ IDEF6 rationalités conceptuelles ;
- ❖ IDEF7 méthode d'audit pour les systèmes d'information ;
- ❖ IDEF8 interfaces utilisateur ;
- ❖ IDEF9 conception dirigée par les scénarii des systèmes d'information ;
- ❖ IDEF10 architectures d'implantation ;
- ❖ IDEF11 artefacts informationnels ;
- ❖ IDEF12 organisation du système ;
- ❖ IDEF13 formalisme tri-schémas ;
- ❖ IDEF14 conception de réseaux.

Comme il est possible de le constater, la méthode varie selon le domaine d'utilisation, et l'utilisateur dispose alors d'une multitude de méthodes suivant le domaine dans lequel il se trouve. En prenant l'exemple de l'industrie, cette vision normalement constituée de 15 facettes sera souvent simplifiée, en utilisant majoritairement les méthodes IDEF0, IDEF1 et IDEF3 (Figure 2.14¹⁷).

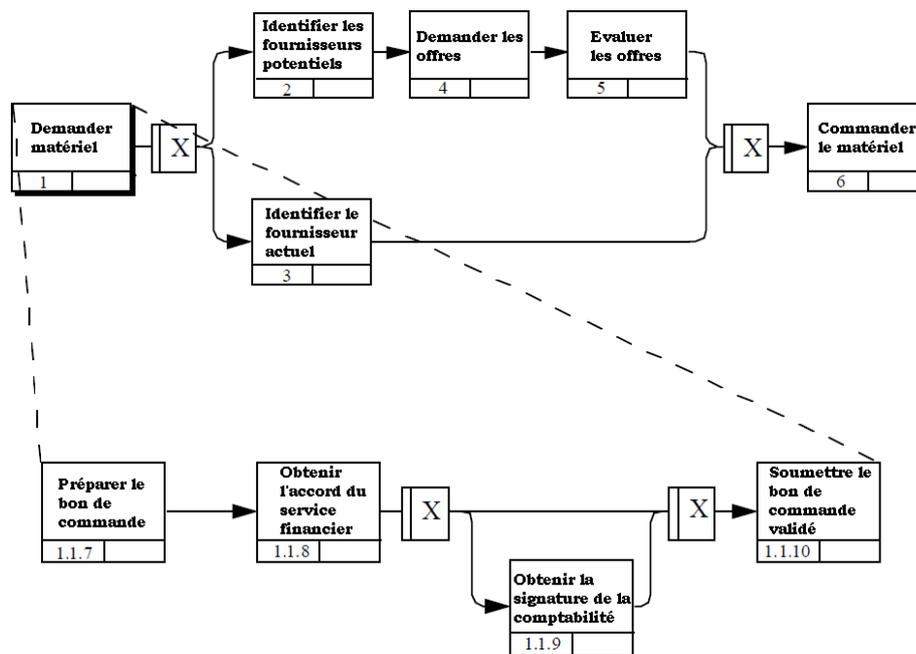


Figure 2.14 : Exemple de diagramme IDEF 3

IDEF0 s'est diffusée dans les services méthodes de nombreuses entreprises européennes, comme l'une des méthodes les plus efficaces d'analyse descendante, permettant de décrire un système de façon systématique et hiérarchique comme sur la Figure 2.15¹⁸.

¹⁷ Tirée de http://www.idef.com/pdf/Idef3_fn.pdf (02/03/2015)

¹⁸ Tirée de <http://www.idef.com/pdf/idef0.pdf> (02/03/2015)

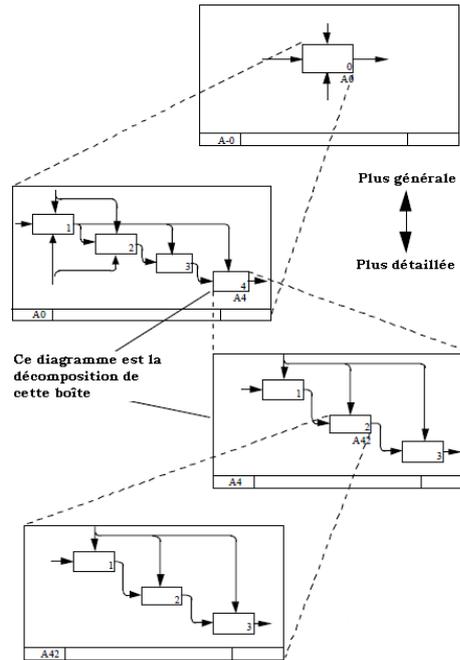


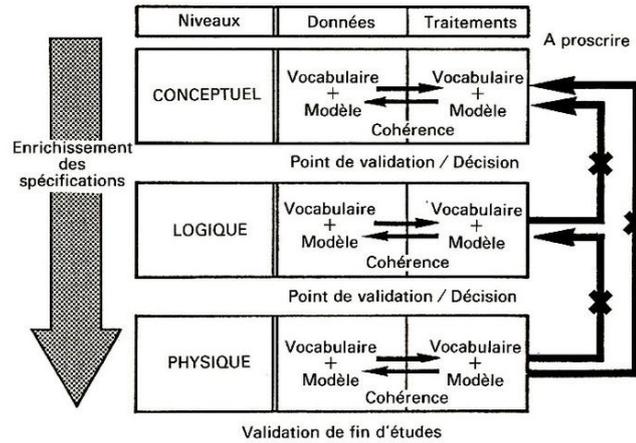
Figure 2.15 : Exemple de diagramme IDEF 0

IDEF0 s'inspire fortement de SADT (Structured Analysis and Design Technique), diagramme très répandu, conçu dans la société Softech par D. Ross [ROSS 77] et permettant une analyse fonctionnelle descendante d'un système complexe ou d'un processus opératoire. Les briques représentent des fonctions elles-mêmes images d'activités, de processus ou de transformations à l'intérieur du système. Une boîte fonctionnelle appelée « Activité » est reliée aux autres boîtes et à l'environnement extérieur par l'intermédiaire de flèches, cette activité pouvant alors être à son tour décomposée à un niveau inférieur. La popularité de cette représentation vient du fait qu'elle constitue une approche cohérente et non-ambiguë de décomposition d'un système permettant une communication claire entre analystes, concepteurs et utilisateurs du système. Des similitudes sont à remarquer avec l'approche par composant, et notamment, le concept de boîte noire pour les processus « parents » et de boîte blanche pour les processus « enfants ».

2.4.2.3 La méthode MERISE

La « Méthode d'Étude et de Réalisation Informatique pour les Systèmes d'Entreprise » (MERISE) [TARDIEU 83] orientée « systèmes d'information », a rencontré un succès certain dans les années 1980. La conception du SI repose sur un cycle d'abstraction composé d'étapes (Figure 2.16¹⁹) afin d'aboutir à un système fonctionnel proche de la réalité. Cette méthode préconise d'analyser séparément données et traitements dans chaque étape, le passage à la suivante correspondant à un point de validation et de décision. Le déroulement standard de cette méthode est donc de valider une à une chaque étape en prenant en compte les résultats de l'étape précédente. MERISE a été très utilisée (et en particulier son modèle entité-relation) avant de voir sa cote décroître au sein des architectes de SI du fait de sa mise en œuvre très lourde et de l'apparition d'UML, MERISE en étant d'ailleurs l'un des piliers.

¹⁹ Source : « CIM, les nouvelles perspectives de la production », Jean-Baptiste Waldner, DUNOD, 1990.



La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.

Figure 2.16 : Niveaux et étapes de la méthode MERISE

2.4.2.4 La méthode OSSAD

Dernière méthode présentée ici, OSSAD (*Office Support Systems Analysis and Design*), permet la modélisation de systèmes d'information et d'organisation (ouvertes et standards) afin de gérer les problèmes organisationnels amenés par l'arrivée massive de la technologie. Celle-ci a été développée dans le cadre d'un projet de recherche européen ESPRIT²⁰ (1985 à 1989) et repose sur deux niveaux de modèles : abstrait (Figure 2.17) et descriptif. Le premier s'intéresse aux objectifs d'une organisation, indépendamment des moyens et des ressources mis en œuvre pour les réaliser, ce niveau se concentrant sur la représentation des fonctions et des informations circulant à l'intérieur de l'organisation offrant ainsi plusieurs paliers de visualisation et la possibilité de décomposer ainsi des processus en sous-processus.

Le second, quant à lui, décrit les moyens humains et les ressources technologiques qui sont présents dans cette organisation, en se concentrant sur la représentation des procédures et des opérations nécessaires à l'accomplissement des fonctions. Il repose sur trois types de formalismes graphiques : les matrices activités-rôles, les graphes de circulation des informations et les graphes des opérations (Figure 2.18).

MODELE ABSTRAIT

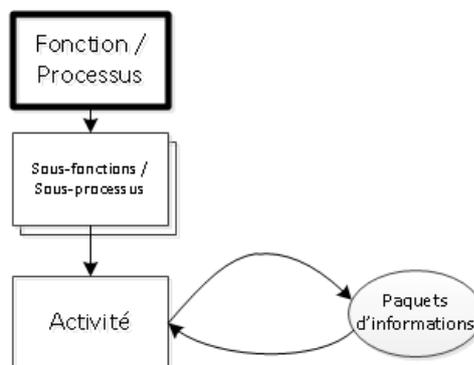


Figure 2.17 : Modèle abstrait de la méthode OSSAD

²⁰ Désormais connu sous le nom des programmes cadres ICT.

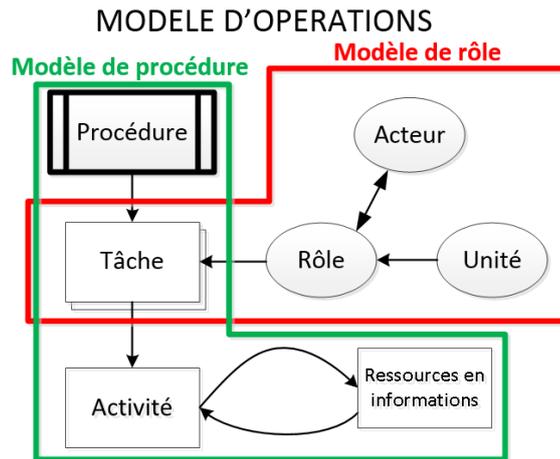


Figure 2.18 : Modèle d'opérations de la méthode OSSAD

2.4.3 Conclusion au niveau des méthodes

L'objectif étant à terme de réaliser et d'utiliser un modèle de simulation, il est constaté que chaque méthode possède bien entendu des inconvénients. Tout d'abord, le langage IDEF se prête difficilement à la traduction de sa modélisation des processus en un modèle de simulation. MERISE quant à elle, a l'inconvénient majeur d'être très lourde à mettre en œuvre, et la simulation étant ici un moyen de valider les hypothèses, chaque changement se traduira par un travail de refonte des modèles MERISE, chose peu envisageable. Pour terminer, les modèles OSSAD imposent de séparer acteurs et ressources des processus, ce qui s'écarte de notre but, à savoir obtenir un modèle complet avec toutes les informations disponibles sur ce modèle. Nous ne sélectionnerons donc aucune méthode en particulier du fait de leurs lacunes, d'autant plus que nous avons choisi d'utiliser la méthodologie ASCI qui nous en dispense.

2.5 LA SIMULATION : POURQUOI ET COMMENT ?

L'utilisation des différents outils précédents ayant pour optique de concevoir un modèle des processus utilisable pour construire un modèle de simulation, il est intéressant de se pencher sur la simulation elle-même pour voir en quoi elle consiste et voir ce qu'elle peut apporter.

2.5.1 Qu'est-ce que la simulation ?

La simulation se retrouve partout, et cela autant dans le monde du divertissement par le biais des jeux-vidéos que dans l'industrie et l'aviation (optimisation de ligne de production, simulation en soufflerie) comme le signale la définition suivante :

Définition 2.8 : « Représentation du comportement d'un processus physique, industriel, biologique, économique ou militaire au moyen d'un modèle matériel dont les paramètres et les variables sont les images de ceux du processus étudié. (Les modèles de simulation prennent le plus souvent la forme de programmes d'ordinateurs auxquels sont parfois associés des éléments de calcul analogique). »²¹

²¹ Tirée du Larousse : <http://www.larousse.fr/dictionnaires/francais/simulation/72824>

Le but de la simulation est donc d'imiter, de copier le comportement d'un système réel avec un modèle de simulation qui réagira de manière similaire, pour en observer l'évolution à travers le temps et générer ainsi un historique artificiel du fonctionnement du système et d'en tirer des conclusions quant à ses caractéristiques. Elle peut donc être vue comme un outil d'analyse pour prédire les effets de changements sur le système ou pour en prévoir la performance.

La simulation s'apparente donc à une aide dans le design, l'émulation, l'exécution de systèmes, pouvant être basée sur un modèle physique ou sur des modèles numériques informatiques, nécessitant dès lors l'utilisation d'un logiciel.

Cependant beaucoup de systèmes réels étant très complexes, insolubles, mathématiquement parlant, l'utilisation de la simulation numérique permet de contourner ce problème, mais celle-ci ne doit pas être l'objectif visé à chaque fois, certaines situations ne la requérant ou ne s'y adaptant pas, comme nous allons le voir.

2.5.2 Doit-elle être utilisée à chaque fois ?

La simulation n'est pas la solution à tous les problèmes existants. Il existe des situations où y recourir peut apporter plus de difficultés que de bénéfices. Voyons quels sont ces cas :

↳ Les situations à éviter

Parmi ces nombreuses situations, nous allons retrouver des cas qui découlent du bon sens mais certaines d'entre elles ne sont pas de prime abord évidente :

- ❖ *Le problème étudié est soluble analytiquement ou par le sens commun* : utiliser la simulation pour faire de la simulation n'a pas d'intérêt. En effet, la construction d'un tel modèle demande de l'énergie et des moyens qui, mobilisés dans une résolution immédiate, se montreraient beaucoup plus pertinents ;
- ❖ *Des essais réels se révèlent plus faciles* : cette situation se recoupe légèrement avec la précédente. En effet, des essais sur le terrain peuvent se révéler plus rapides qu'une simulation dans l'obtention d'un résultat ;
- ❖ *Le coût* : la construction d'un modèle de simulation peut représenter un investissement massif dans certains cas alors que les gains qui en seront tirés ne seront que minimes. Estimer en partie ce rapport dépenses/profits avant de se lancer dans une telle démarche se révèle être pertinent ;
- ❖ *Le manque de temps et de ressources* : tout comme l'aspect économique, les besoins matériels et temporels pour cette démarche sont conséquents et nécessitent donc du personnel et des ressources mobilisées sur cette tâche souvent à temps plein. Initier un tel projet impose donc de vérifier la disponibilité de ce personnel et de ses ressources ;
- ❖ *Les données* : dans la même ligne, le manque de données utiles au bon développement du modèle de simulation et à son exploitation, sera limitant dans l'obtention d'un modèle significativement représentatif de la réalité et donc dans les gains espérés au lancement de cette démarche ;
- ❖ *L'attente vis-à-vis du modèle* : pour terminer, un modèle de simulation reste un modèle de simulation et ne peut résoudre tous les problèmes et anticiper toutes les situations. Des attentes irréalisables vis-à-vis de cette démarche doivent stopper le projet à ses débuts pour éviter toute désillusion.

2.5.3 Quels avantages et inconvénients ?

Comme toute démarche, la simulation apporte son lot d'avantages et d'inconvénients qu'il est bon de mettre en évidence.

2.5.3.1 Avantages

Comme dit précédemment, l'intérêt majeur de la simulation est de voir en direct et de manière efficace les effets d'altérations diverses sur le comportement du système sans en perturber le déroulement réel. L'étude de tels effets passe aussi par la compréhension et l'appréhension du système qui se fait plus précise grâce à la simulation, en mettant en évidence des points clés, des éléments et des composants sensibles aux modifications parfois peu évidents au premier abord. D'un point de vue plus technique, la simulation est donc capable de gérer la complexité de nombreux éléments en simulant leur évolution indépendamment les uns des autres puis en estimant l'impact globale sur la population en synthétisant les effets induits à titre individuel par chaque entité. Ces effets sont une conséquence des différentes choix faits à chaque nœud décisionnel du système, ce qui dans le cadre d'un système stochastique impliquera que ces choix sont issus de distributions probabilistes.

De même, la mise en route et en pratique de nouvelles situations techniques ou organisationnelles peuvent être anticipées et testées par son utilisation, tout comme l'estimation des besoins pour le fonctionnement du système suivant différents scénarios. Pour terminer, la simulation peut tout simplement être utilisée dans le but d'apprendre ou d'illustrer le fonctionnement d'un système.

On peut lister encore parmi les avantages existants :

- ❖ Prédire les futures performances du système ;
- ❖ Eliminer les risques et incertitudes quant à son fonctionnement ;
- ❖ Eviter des coûts inutiles ;
- ❖ Etre plus facile à contrôler qu'une expérience réelle ;
- ❖ Permettre de tester des situations irréalisables ;
- ❖ Permet de tester des situations extrêmes ;
- ❖ Visualiser le système bien avant sa construction ;
- ❖ Tester des hypothèses.

2.5.3.2 Inconvénients & Limites

Bien entendu, la simulation présente aussi des inconvénients et limitations dans ce qu'elle peut faire, son développement et dans son utilisation propre.

Comme déjà dit, la construction d'un modèle de connaissance demande des connaissances et de la pratique, ce qui en cas de défaut, limite les possibilités de modélisation, tout comme elle demande beaucoup de ressources et de temps pas toujours à disposition. La simulation ne permet pas d'obtenir à chaque fois les résultats souhaités ou sous la forme voulue. Il y a alors un travail d'interprétation à faire, exercice qui n'est pas toujours évident.

De plus, la simulation permet d'apporter des réponses et non pas une solution clés en main aux problèmes étudiés.

A cela peut s'ajouter :

- ❖ La simulation n'est qu'une approximation d'un système réel ;
- ❖ L'exécution d'une simulation peut prendre un temps considérable ;
- ❖ La fiabilité des résultats n'est pas toujours assurée et doit donc être suivie avec rigueur.

2.5.4 Les différentes classes de simulation

Parmi les grandes classes de simulation qu'il est possible d'observer à travers les publications, les simulations continues, à événements discrets et par agents sont les plus représentées. D'autres sont utilisées telles que la simulation de Monte Carlo, la simulation comportemental, mais dans les paragraphes suivants seules les deux premières énoncées seront abordées en énonçant quels avantages et inconvénients leurs sont inhérents.

2.5.4.1 La simulation à événements discrets

Globalement, il est possible de dire pour ce type de simulation, que la **DES** (acronyme anglais pour « Discrete Event Simulation ») est transparente, intuitive et surtout supportée par de nombreux outils logiciels. Reconnue pour être « très flexible » si ce n'est la plus flexible, pour la modélisation de systèmes très détaillés prenant en compte des contraintes, les interactions et des conditions de passage par exemple, son approche *stochastique* permet de modéliser par le biais de distributions mathématiques des éléments non linéaires tels que des temps d'attente mais encore l'aspect dynamique d'une population de file d'attente. Le principe est donc de modéliser les opérations s'exécutant dans le système de manière chronologique en suivant une séquence d'événements prédéfinis, chacun d'entre eux arrivant à un instant bien précis en lien avec un changement d'état, de statut du système. Ainsi dans le domaine hospitalier, les modèles DES sont construits de manière à connecter une série d'éléments comme des files d'attentes ou des tâches à travers lesquelles des entités (par exemple les patients) voyagent. Les paramètres clés de ce type de système comprennent donc les règles de gestion telle que le cheminement des entités en son sein suivant leurs attributs mais aussi des informations temporelles telles que la durée des tâches ou les temps de déplacement.

De surcroît, la DES peut être appliquée de nombreuses manières et se trouve être très suivie dans le milieu de la modélisation. Il existe donc un nombre important de logiciels la supportant. L'une des approches permettant de tirer parti au maximum de la DES est la simulation visuelle interactive où facilité d'utilisation et représentation graphique sont mises en relation à travers des animations et des graphiques interactifs pour représenter la dynamique du système étudié.

Cependant comme toute chose, la DES n'a pas que des avantages et possède un certain nombre de limitations. Le fait de pouvoir traduire le comportement de chaque élément du système peut rendre très vite un modèle complexe et difficile à comprendre, tout comme accroître de façon importante son temps d'exécution handicape alors fortement l'utilité d'une telle méthode.

2.5.4.2 Simulation continue

Plus axée sur la modélisation à objectif stratégique, elle se révèle souvent plus intuitive et rapide à mettre en place, et est tout comme la précédente supportée par bon nombre de logiciels où l'interface graphique tient une plus grande place pour faciliter son développement. Ce type de simulation se révèle moins complexe du point de vue codage que la DES et a de plus cette capacité d'approche holistique sur un système. Elle est basée sur l'utilisation et la gestion des stocks et flux

interagissant dans le système avec les différents éléments/entités le constituant, et permet d'intégrer l'idée que toute chose en affecte une autre, notion indissociable du monde hospitalier.

De plus, elle se montre plus utile pour des analyses qualitatives (contrairement à la DES plus quantitative), et permet de comprendre comment un système interagit (modélisation graphique) et de voir si celui-ci est stable ou non en fonctionnement. Contrairement à la DES, celle-ci ne cherche pas forcément à fournir la meilleure solution au problème soumis mais plutôt d'aider à identifier les variables clés ayant une influence sur le comportement du système et quelles solutions seraient envisageables.

Dans le domaine hospitalier, un tel modèle peut montrer l'évolution du patient à l'intérieur d'un réseau complexe (différents services) au fil du temps, avec des incréments de temps si petites qu'elles ne perturberont pas la continuité du flux.

Malheureusement la simulation dynamique est fondamentalement **déterministe**, a contrario de stochastique, et souffre de quelques difficultés dans l'exploitation et l'intégration des données, d'où son utilité plus dans un cadre qualitatif, un paramètre important dans le domaine de la santé.

2.5.5 Les niveaux d'abstraction

Les systèmes étudiés à l'aide de la simulation étant très souvent complexes de par leur fonctionnement, créer des modèles de simulation possédant un niveau de détails élevé (afin de prendre en compte un maximum de cas, de spécificités du système, de scénarios, etc.) se révèle difficile. L'obligation de précision, pour un rendu le plus réaliste, imposera donc une connaissance la plus fine possible du système ainsi qu'un jeu conséquent de données le plus exhaustif possible. Similairement à la modélisation des processus, il sera donc nécessaire de faire un choix quant au niveau d'abstraction du modèle de simulation ainsi qu'à l'horizon visé.

[[SINREICH 04](#)] propose de mettre en relation le niveau d'abstraction des modèles de simulation avec le niveau de modélisation adopté, qui dans notre cas concerne les processus. Ainsi il présente un schéma détaillant le choix des options de simulation en fonction de chaque niveau d'abstraction.

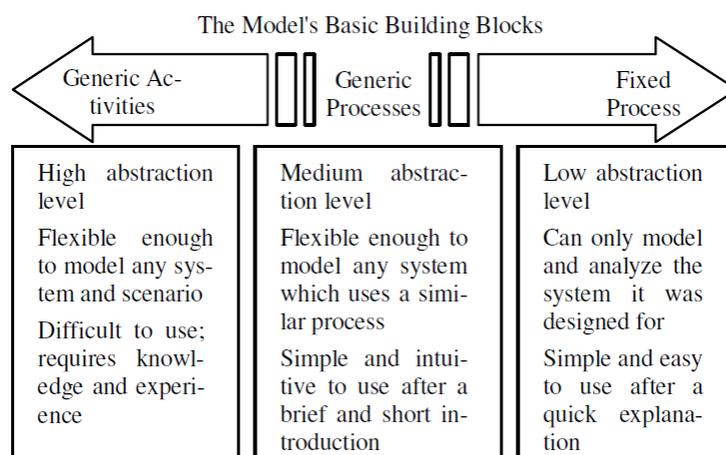


Figure 2.19 : The Range of Modeling Options and the Building Blocks Used in Each Case

Dès lors les travaux de [[RODIER 10](#)] donnent une version plus poussée des liens entre modélisation des processus et simulation :

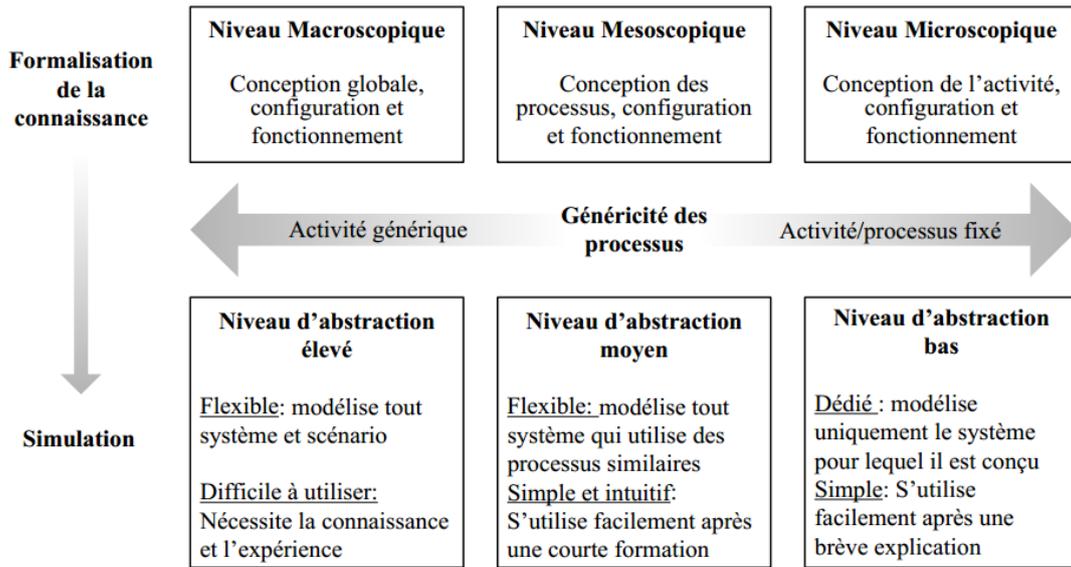


Figure 2.20 : Niveaux de modélisation pour la formalisation de la connaissance

Le niveau d'abstraction est donc bien dépendant du niveau de formalisation de la connaissance. En effet, il est difficile d'envisager un modèle de simulation très détaillé si le niveau de connaissance du système est resté à un niveau macroscopique. L'un ne va donc pas sans l'autre.

Comme énoncé plus haut, l'horizon temporel visé aura aussi son importance quant au niveau d'abstraction du modèle de simulation. Abordons dans un premier temps cette notion d'horizon temporel qui se retrouve dans toutes les questions liées à la modélisation de système que l'on souhaite étudier.

Les trois niveaux de définition de l'horizon temporel sont les suivants : stratégique, tactique et opérationnel. Chacun d'entre eux se situant à plus ou moins long terme comme nous allons le voir.

- ❖ Horizon stratégique : ce premier horizon vise une période généralement supérieure à 5 ans et permet de se projeter dès lors sur l'impact futur de décisions actuelles. Les problèmes d'organisation, de conception, d'optimisation sont généralement traités à ce niveau, tels que les questions de dimensionnement, de localisation des équipements ;
- ❖ Horizon tactique : ce second, prenant en compte généralement les années à venir, c'est-à-dire de 1 à 5 ans, il s'axe plus sur des problèmes liés à la planification ;
- ❖ Horizon opérationnel : le dernier quant à lui vise un horizon bien sûr beaucoup plus proche, des jours suivants à quelques mois, et traite des problèmes liés à la production : le pilotage des ateliers par exemple.

Il y aura donc une analogie entre horizon temporel et modèle de simulation. La distinction à faire entre modèle stratégique et opérationnel s'est révélée souvent très utile pour identifier les différents niveaux d'application de la modélisation et donc le niveau d'abstraction du modèle. Le contenu de cette distinction découle bien sûr de l'explication précédente des horizons. En règle générale, les modèles stratégiques concernent la prise de décision où il faut tenir compte d'un nombre important de paramètres et cela sur un intervalle de temps relativement important (comme les politiques de gestion d'entreprises). Typiquement les décisions stratégiques impactent une grande partie du système étudié et elles sont directement liées aux relations qui existent entre les

organes directeurs et avec les organes sur le terrain. Les modèles opérationnels quant à eux sont liés à la dynamique, au fonctionnement des entités les plus petites du système et qui composent donc la base de celui-ci.

A noter que la distinction entre les deux n'est pas toujours parfaitement claire. Par exemple, un modèle générique à visée opérationnelle peut être appliqué à de nombreux départements et avoir de la sorte de fortes implications stratégiques. De plus, en fonction du contexte, l'interprétation de l'utilité du modèle ne sera pas forcément la même et ceci décidera donc du côté opérationnel ou stratégique de ce dernier.

En conclusion, la distinction entre ces deux horizons de modélisation se doit d'être vue plus comme un spectre de possibilité et non pas sous l'angle de catégories bien définies et séparées. Malgré cela, un choix doit être fait pour choisir le niveau d'abstraction du modèle de simulation, même s'il sera possible de moduler cela pour que le modèle soit dans la capacité de répondre à des préoccupations autres que le choix précédemment fait.

Cependant, de notre point de vue, le niveau d'abstraction d'un modèle de simulation serait dépendant non pas seulement des deux critères présentés précédemment (horizon temporel et niveau de formalisation de la connaissance), mais aussi d'un troisième qui a été énoncé dans la partie 2.2.5 et qui est l'objectif visé par l'étude : identifier, contrôler ou restructurer.

2.6 LES CODES DE LA GÉNÉRICITÉ

Privilégiant la méthodologie ASCI pour nos travaux, cette dernière fait référence, tout comme [RODIER 10] à travers la Figure 2.20, à une notion importante et primordiale pour nous : la généralité. Cette notion a été abordée par un certain nombre d'auteurs comme nous allons le voir, les travaux les plus poussés étant ceux de [FLETCHER 09a] sur lesquels se basera majoritairement ce qui va suivre.

De prime abord, même si le terme « généralité » se comprend facilement, un modèle dit « général » n'est pas pour autant une chose aisée à obtenir. C'est pour cette raison que des auteurs ont défini et énoncé des principes permettant d'atteindre un tel résultat. Pour le premier, [SINREICH 04], il est constaté que tout comme la granularité, trois niveaux ont été mis en évidence par celui-ci :

- ❖ Un équivalent macroscopique : un modèle est fortement général quand il est hautement abstrait et couvre beaucoup de systèmes et scénarios ;
- ❖ Un équivalent microscopique : un modèle n'est pas général si celui-ci ne représente qu'un seul système ;
- ❖ Un équivalent mésoscopique : tous les modèles représentant un processus similaire.

Ces trois niveaux restent donc assez succincts dans leur description. C'est là tout l'importance du travail de [FLETCHER 09a] qui a approfondi l'étude de cette notion. Il commence par classer les modèles dans quatre catégories du plus général au plus spécifique. S'en suivra une traduction de l'opposition spécifique/général à travers cinq thématiques identifiées comme majeures pour [FLETCHER 09a].

2.6.1 Le spectre de la généralité

A travers les nombreux articles que [FLETCHER 09a] a étudiés ce dernier a pu dresser un tableau assez fin du découpage des différents modèles informatiques des hôpitaux qu’il a pu rencontrer. Celui-ci identifie donc quatre niveaux – le menant même à cinq niveaux en affinant – en utilisant comme règle de classification le niveau d’abstraction et de transportabilité de chaque modèle. Ces quatre niveaux (Figure 2.21) sont :

- ❖ **Niveau 1** : un modèle générique dit de principe (une simple file d’attente par exemple) ;
- ❖ **Niveau 2** : un cadre générique de modélisation. Celui-ci pourrait contenir des modules représentant différentes étapes du processus générique. Ce niveau a l’avantage d’être en lien avec les niveaux 1 et 3 à la fois. En effet, adapté par l’utilisateur il pourrait mener à l’obtention d’un modèle spécifique – comme nous allons le voir dans le niveau 3 – mais il pourrait par la même occasion illustrer des principes généraux visibles au niveau 1 ;
- ❖ **Niveau 3** : un modèle générique cette fois-ci dédié. Il se concentre ici sur un service, une entité spécifique. Le modèle ne change pas, ce sont les données qui lui sont fournies qui évoluent et changent ;
- ❖ **Niveau 4** : un modèle dédié à un service ne pouvant être utilisé que par ce service.

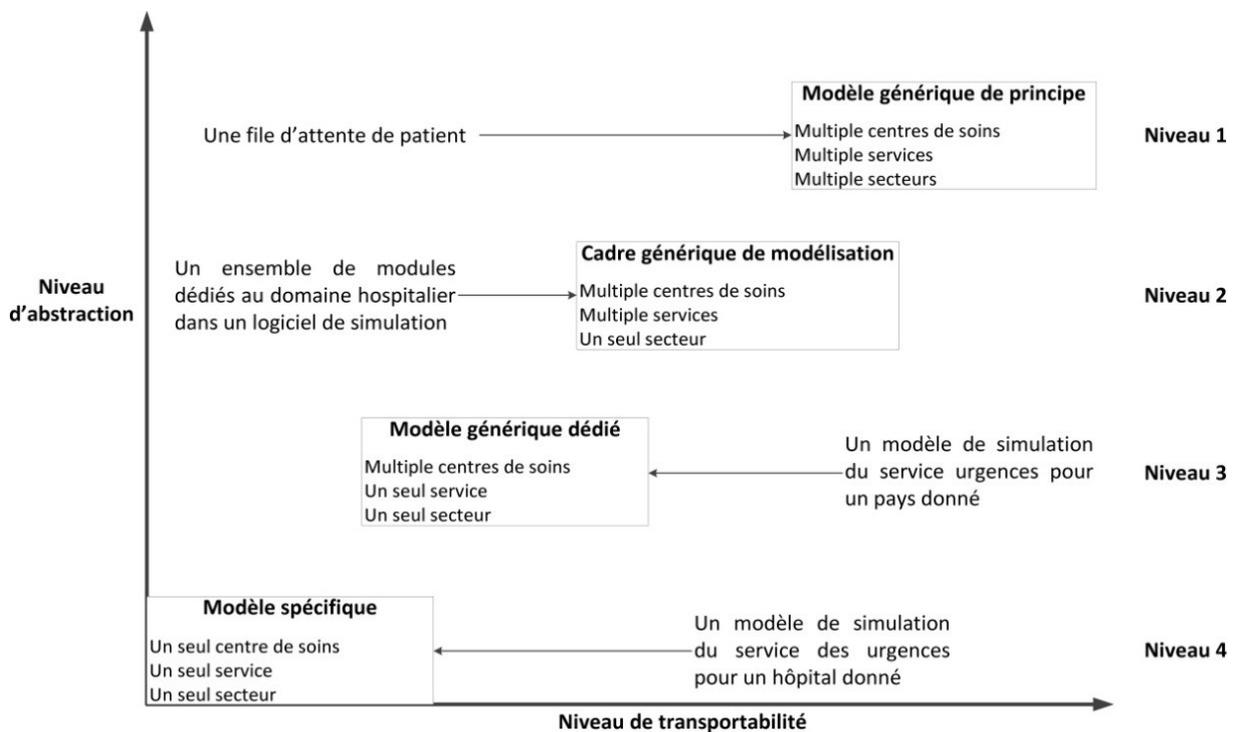


Figure 2.21 : Spectre de la généralité

Comme énoncé précédemment, les auteurs ont ici poussé plus loin cette réflexion en scindant le niveau 3 en deux sous-niveaux. Le premier est le niveau 3A où les modèles génériques ont une dimension « centrale » dans le sens où ce dernier représente/centralise le fonctionnement de nombreuses structures contrairement au niveau 3B qui, bien que représentant lui aussi plusieurs structures, celui-ci commence à avoir une dimension locale et donc à être dédié. Il est donc possible de résumer en disant que le niveau 3A se rapproche plus du cadre de modélisation et le niveau 3B du modèle spécifique ou multi-spécifique dans un sens.

La Figure 2.22 résume ce découpage :

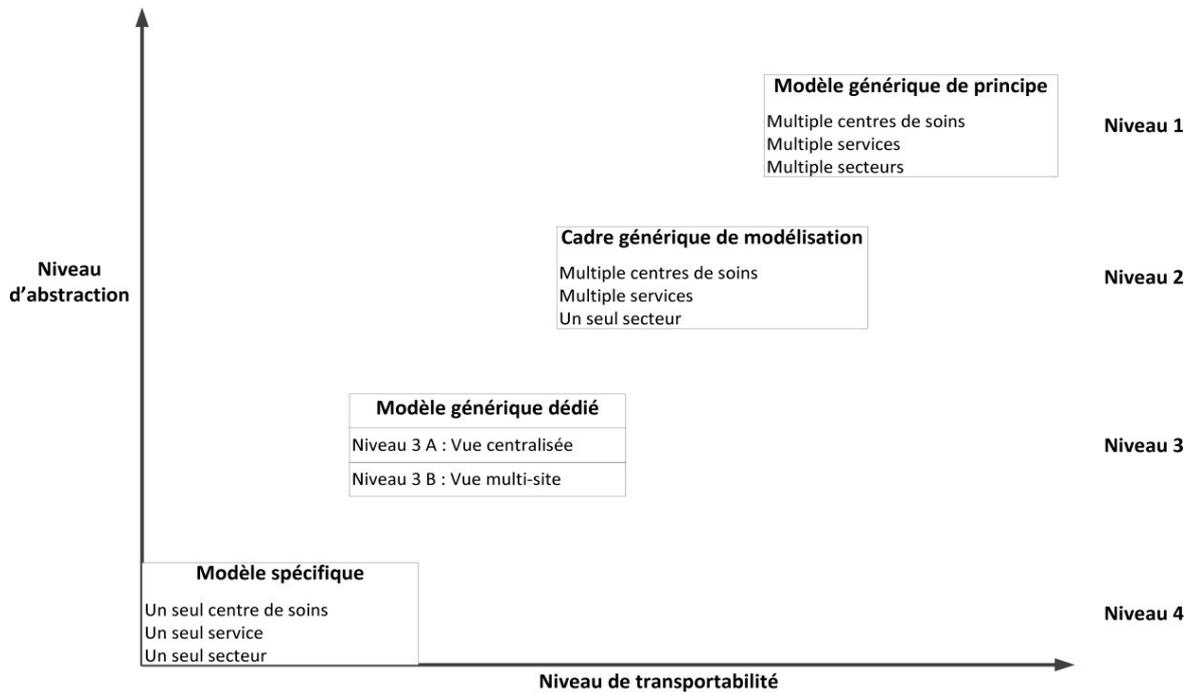


Figure 2.22 : Spectre revu de la généralité

2.6.2 Générique versus Spécifique

En se concentrant sur les trois derniers niveaux à savoir 3A, 3B et 4 du spectre précédemment établi, [FLETCHER 09a] a une nouvelle fois approfondi sa réflexion en scindant ces trois niveaux selon cinq thématiques majeures. Ces dernières représentent en tout 24 critères pour évaluer la généralité d'un modèle et sont issues de l'étude des nombreux articles analysés par ces auteurs. Elles sont :

- ❖ Initiation du projet ;
- ❖ Conception et construction ;
- ❖ Données ;
- ❖ Validation ;
- ❖ Implémentation.

Présenté sous forme de tableaux, nous ne montrerons dans ce qui suit que le tableau 3 dans son entièreté et un exemple de chacune des autres tables qu'il est possible de retrouver entièrement en **Annexe 3**. Le Tableau 2.8 est donc une copie du Tableau 3 de [FLETCHER 09a]. Celui-ci a été choisi car il traite de la thématique la plus en rapport avec nos travaux, à savoir la conception d'un modèle générique de simulation.

Le Tableau 2.9 donne quant à lui les critères « Conflit avec des modèles existants » (Initiation du projet) ; « Données tests de départ » (Données) ; « Représentation » (Validation) et enfin « Volonté d'utilisation » (Implémentation).

Tableau 2.8 : Facteurs clés dans la conception et la construction d'un modèle générique ou spécifique

Critère	Niveau 3A – Générique central	Niveau 3B – Générique multi-site	Niveau 4 – Spécifique
Niveau de réutilisabilité du code/modèle	Dépend des travaux précédemment réalisés par le modélisateur et l'organisation, du degré de partage des connaissances ou encore de la disponibilité du code/modèle générique	Dépend des travaux précédemment réalisés par le modélisateur et l'organisation, du degré de partage des connaissances ou encore de la disponibilité du code/modèle générique	Dépend des travaux précédemment réalisés par le modélisateur et l'organisation, du degré de partage des connaissances ou encore de la disponibilité du code/modèle générique
Conception des processus	Réalisé avec des experts du domaine, et des experts locaux	Réalisé avec des experts du domaine, et des experts locaux	Réalisé avec des experts locaux
Représentation des problèmes locaux	Pas nécessairement requis	En changeant les données d'entrée et non pas la structure du modèle	Potentiellement la meilleure – représentation autant d'un point de vue structure et conception du modèle qu'au niveau des données d'entrée.
Justesse d'utilisation	Clairement définir les systèmes et les problèmes pour lesquels le modèle est utilisable. Le modèle doit être ajustable pour coller à différents problèmes.	Nécessaire de définir clairement quand le modèle sera approprié (par exemple quand les modifications des données d'entrée sont appropriées pour modéliser le problème)	Forte probabilité de devoir faire des ajustements au niveau de la structure du modèle pour que celui-ci colle au problème
Niveau de détails	Inclus des processus significatifs communs à de multiples centres de soins	Inclus des processus significatifs communs à de multiples centres de soins	Inclus des processus significatifs à un centre de soins en particulier
Objectif de la conception	Modéliser des processus usuels communs à de nombreux centres de soins	Modéliser des processus usuels mais assez flexible pour représenter quelques processus locaux	Modéliser un système précis (local). Ce processus « local » doit être similaire à ceux des autres centres de soins, sans pour y être parfaitement ressemblant
Connaissance de la structure et du fonctionnement interne du modèle par l'utilisateur	Si l'utilisateur a été impliqué dans le développement du modèle, alors possiblement élevé	Initialement bas (l'utilisateur local n'est pas impliqué dans le développement)	Les utilisateurs locaux doivent avoir été impliqués dans le développement du modèle, donc probablement élevé

Tableau 2.9 : Exemple de critères pour les quatre autres thématiques

Thématique	Critère	Niveau 3A – Générique central	Niveau 3B – Générique multi-site	Niveau 4 – Spécifique
Initiation du projet	Conflit avec des modèles existants	S'occuper des manques, lacunes perçus/réels des modèles centraux existants	Peut-être en conflit avec des modèles locaux déjà existants, ou peut avoir été conçu pour remplir un manque local	S'occuper des manques, lacunes perçus/réels des modèles locaux existants
Données	Données tests de départ	Probablement des moyennes nationales, complétées d'exemples locaux et d'estimations nationales	Probablement des moyennes nationales, complétées d'exemples locaux	Données historiques locales
Validation	Représentation	Niveau « général » uniquement	Un ensemble de centres de soins	Un seul centre de soins
Implémentation	Volonté d'utilisation	Dépend de la perception de la qualité, des problèmes couverts, de l'existence d'autres modèles	Dépend de la perception de la qualité, des problèmes couverts, de l'existence d'autres modèles et des politiques locales	Dépend de la perception de la qualité, des problèmes couverts, de l'existence d'autres modèles

L'utilité de ces tableaux sera mise en avant lors de la création de notre modèle générique et nous pourrons dès lors voir si nous répondons aux critères énoncés par [\[FLETCHER 09\]](#).

Cette partie traitant plus de l'abstraction à laquelle doit être soumis chaque modèle afin d'atteindre un objectif de généralité, regardons du côté d'une autre notion qui a sa place et qui s'avère parfaitement complémentaire avec la généralité : il s'agit de la réutilisabilité.

2.6.3 La réutilisabilité

Cette nouvelle notion a aussi son importance dans le domaine de la généralité. En effet, un modèle peut être générique autant de par son contenu que par ses fonctions. Prenons un instant pour expliquer cette différence. D'un point de vue utilisateur (en opposition à programmeur/concepteur), la généralité d'un modèle sera établie par sa capacité à s'adapter à de multiples situations, à pouvoir les traiter et à en fournir une analyse (en termes de résultats). L'idée de généralité « de fonction » sera utilisée. D'un autre côté, du point de vue concepteur, un modèle pourra être vu comme générique s'il possède en son sein, une ou plusieurs parties capables de s'adapter, de convenir à d'autres modèles. L'idée est donc cette fois-ci la « réutilisabilité » du modèle pouvant être aussi vue comme la généralité « de contenu ».

C'est l'idée même de l'article de [ROBINSON 04]. A travers cette étude les auteurs établissent différents niveaux de réutilisabilité et interprètent celle-ci de manière différente, comme nous allons le constater.

Dans un premier temps [ROBINSON 04] établit lui aussi un spectre de « sa » genericité en évaluant la réutilisabilité suivant quatre niveaux (Figure 2.23). Chacun de ces niveaux symbolise une part de réutilisabilité plus ou moins importante et a lieu plus ou moins souvent. Détaillons chacun de ces niveaux :

- ❖ *Niveau 1 : Ponction partielle de code* : il s'agit d'une démarche bien connue des développeurs et autres programmeurs qui en reprenant des portions d'un code qui fonctionne tentent d'en faire quelque chose de nouveau. Cette démarche demande une grande confiance dans le code qui sera sélectionné, ce qui explique que majoritairement cette démarche est réalisée par la personne ayant réalisé le code à la base. Comme le note [ROBINSON 04] cette activité est extrêmement répandue et sûrement le moyen par lequel nombre d'entre nous avons appris à coder ;
- ❖ *Niveau 2 : Réutilisabilité de fonctions* : niveau supérieur à la ponction de code, il s'agit ici de prendre des fonctions complètes pour les réutiliser dans un nouveau modèle/code. Usuellement, elles ont un fonctionnement très spécifique et sont très détaillées, permettant de vérifier aisément si ces fonctions s'exécutent comme désiré ;
- ❖ *Niveau 3 : Réutilisabilité de « composants »* : [ROBINSON 04] définit un « composant » comme étant un module (encapsulé) possédant une interface définie, des fonctionnalités limitées et qui s'intègre dans une architecture précise. Ces « composants » sont usuellement plus développés que les fonctions et peuvent avoir un fonctionnement interne très complexe. Ils peuvent donc être vus comme des « super-fonctions », leur réutilisabilité ayant les bénéfices apparents de celle des fonctions et même d'autres avantages. Pour [ROBINSON 04], c'est la zone de son spectre la plus complexe notamment quand les « composants » gagnent fortement en taille et en fonctionnalité ;
- ❖ *Niveau 4 : Réutilisabilité d'un modèle complet* : vu comme le Graal de tout programmeur, la réutilisabilité d'un modèle dans son entièreté pour une nouvelle application reste l'objectif que tout développement devrait supporter dès le départ. Cependant, réutiliser un modèle complet pour une autre raison que celle pour laquelle il a été conçu provoque de gros problèmes de validité et nécessite donc un gros travail de réadaptation/prise en main.

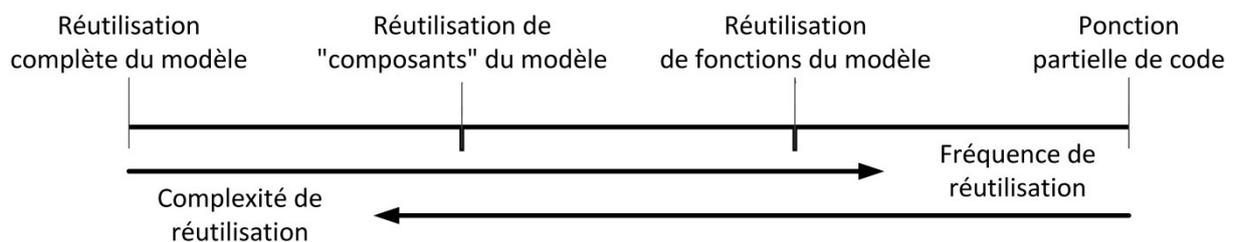


Figure 2.23 : Spectre de réutilisabilité par [ROBINSON 04]

Dans [ROBINSON 04], plusieurs auteurs donnent leurs idées et la conception qu'ils ont de la réutilisabilité. Ainsi nous allons nous attarder quelques temps sur les vues de Simon Taylor.

Pour celui qui se questionne sur ce qui peut être réutilisé dans un modèle, là aussi trois niveaux apparaissent. Fait intéressant sur lequel on pourra se reposer plus tard, Simon Taylor se place du point de vue d'un logiciel de simulation commercial et qui possède donc d'emblée un grand nombre de fonctions et de composants (Commercial-off-the-shelf simulation package). Les trois niveaux sont donc :

- ❖ *Réutilisabilité des composants basiques* : le concepteur réutilise les composants basiques livrés avec le logiciel tels que les stations de travail, files d'attente, serveur, ressources, etc. ;
- ❖ *Réutilisabilité de sous-système du modèle* : il s'appuie cette fois-ci sur des parties « génériques » précédemment développées, auxquelles il peut accéder à travers la librairie du modèle (un convoyeur en est l'exemple parfait), et qu'il peut adapter à un nouveau modèle ;
- ❖ *Réutilisabilité d'un modèle similaire* : le concepteur a développé précédemment un modèle ayant des caractéristiques similaires à celui actuellement désiré. Il suffit donc d'adapter convenablement le modèle ;

L'auteur met en évidence que le gain de temps de réalisation du nouveau modèle, conséquence de la réutilisabilité, est dépendant du niveau auquel l'on se place. Pour le premier niveau, la réutilisabilité des composants ne pourra se faire qu'après avoir testé le fonctionnement de ceux-ci et d'y avoir apporté des modifications. En effet, ce type de composants basiques évolue souvent fortement lors de son utilisation et se spécifie, ce qui implique les tests et modifications.

Pour le second niveau, le raisonnement est similaire. Tester les sous-systèmes nécessaires est une obligation non seulement pour comprendre comment ils fonctionnent (comment ils ont été construits) mais aussi pour voir s'ils modélisent le sous-système étudié convenablement et donc y apporter les modifications en conséquence si utile. Finalement, Simon Taylor conclut que dans la plupart des cas, ce niveau de réutilisabilité sera plus gourmand en temps qu'un développement entier.

Les conclusions sont similaires en ce qui concerne le troisième et dernier niveau, étant donné que tester les fonctionnalités et la validité du modèle prendra plus de temps que pour un sous-système du modèle. De plus pour Taylor, il est rare d'avoir un modèle pouvant être parfaitement fonctionnel pour deux systèmes distincts si cette pluridisciplinarité n'a pas été pensée dès le début du développement.

En conclusion, les attentes derrière la réutilisabilité d'un modèle ne sont pas équivalentes en tout point à celles d'un logiciel (abordée par Richard Nance dans [\[ROBINSON 04\]](#)). La réutilisabilité d'un modèle ne signifie pas pour autant la réutilisabilité du modèle dans sa globalité. Il y a un large spectre associé qui s'étale de la réutilisabilité d'un morceau de code à la complète réutilisation d'un modèle en passant par celle de large partie, la difficulté de réutilisation étant proportionnelle à la portion du modèle réutilisée (Figure 2.24).

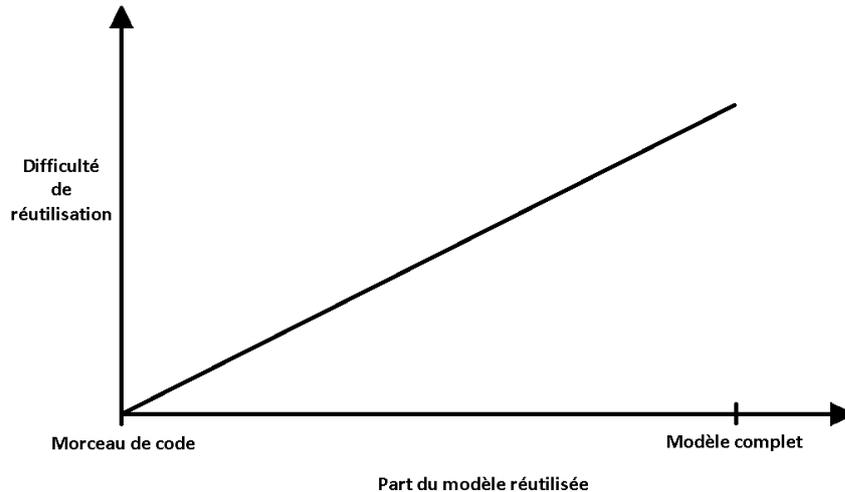


Figure 2.24 : Evolution de la difficulté de réutilisabilité

Cependant, l'économie de temps (et d'argent) pouvant être glanée lors de cette réutilisation est à mettre en avant, malgré les obstacles suivants : il y a encore peu de volonté de la part des développeurs pour faciliter la réutilisabilité des modèles. En effet, cela augmenterait le coût de développement alors que le bénéfice irait aux autres :

- ❖ A cause de problèmes de droits ;
- ❖ Et, car le temps d'adaptation au code d'une tierce personne peut surpasser les bénéfices de la réutilisabilité.

Pour terminer, et afin de mettre en miroir notre genericité « de contenu » avec celle « de fonction », il est possible de constater et notamment dans notre situation que ces deux notions sont interdépendantes. En effet, notre objectif étant le développement d'un modèle générique par le biais d'un logiciel de simulation déjà présent sur le marché, les idées de [\[FLETCHER 09a\]](#) et [\[ROBINSON 04\]](#) viendront à s'entrecouper. Ainsi, bien que le futur modèle BPMN répondra aux spécificités de la genericité, le futur modèle de simulation devra être à la fois capable de simuler un grand nombre de pharmacies différentes (d'où le sens générique) et devra être réutilisable pour des chaînes de productions pharmaceutiques diverses et variées.

2.7 CONCLUSION

Répondre à la demande de nombreuses certifications qui réclament que les processus soient traités, identifiés et analysés, a imposé de comprendre précisément cette notion et a imposé de voir comment il est possible de modéliser ceux-ci et quels moyens (langages, méthodes, méthodologies) se prêtent à cet exercice.

L'étude des processus ici menée nous a donc permis de répondre non seulement à cette question mais aussi de fournir un socle pour démarrer nos travaux en mettant en lumière une liste de langages capables de les modéliser. Malheureusement, le domaine autour duquel s'articulent nos travaux, à savoir le monde hospitalier, limite le nombre de langages pouvant être utilisés du fait des contraintes inhérentes à ce domaine et des capacités et connaissances dont dispose le personnel hospitalier. Ainsi, afin de pouvoir diffuser et dialoguer le plus aisément possible sur les processus, tout en possédant un outil de travail performant, notre choix s'est porté sur BPMN. Ce choix a été

validé après l'évaluation de divers langages et cela selon deux points de vue : un point de vue technique et celui du pharmacien, permettant de prendre en compte les deux domaines liés à notre étude.

De surcroît, pouvoir étudier les processus implique de disposer aussi de méthodes et méthodologies adaptées. Cependant, ayant finalement choisi de travailler avec la méthodologie ASCI, la nécessité d'avoir une méthode pour étudier les processus du CDM n'était plus une obligation, d'autant plus que celles énoncées ici, étant parmi les plus connues, ne se révèlent pas pertinente.

Les difficultés intrinsèques du monde hospitalier, énoncées dans le Chapitre 1, imposent de disposer de techniques permettant d'anticiper les résultats d'un changement de stratégie, quelle qu'elle soit. Le recours aux modèles de simulation s'est donc intensivement développé, ces modèles apportant un point de vue supplémentaire pour analyser et étudier le fonctionnement d'un système en expérimentant de futurs scénarios. De plus, la simulation ne trouve pas une justification que dans le cadre d'une politique de réduction des coûts, des risques, etc. mais aussi dans le fait que le domaine de la santé tend à être de plus en plus complexe de par ses entités multi dépendantes, la grande diversité et les variations dans les demandes, et la rationalisation nécessaire de ces ressources. Et, à cela vient encore s'ajouter la diversité des patients de par leur physiologie, psychologie, milieu social.

Toutefois, l'utilisation de la simulation doit être tempérée, de simples modèles utilisant des méthodes directes pouvant être parfois plus pertinents que celle-ci, cette dernière devant donc être approchée avec un esprit critique afin d'en tirer tous les bénéfices et l'appliquer au mieux. En effet, la simulation est seulement une technique, mais quand elle est utilisée dans le cadre d'un projet structuré (analyse, discussion, design, calibration et autres discussions), cette technique apporte un nouveau jour et des réponses sur des problèmes impossibles à résoudre autrement.

Ainsi pour déployer efficacement la simulation dans les hôpitaux, un accord sur les standards, son intégration et la trame générale la supportant sera donc une des clés pour réussir la mise en place des évolutions organisationnelles, structurelles ou techniques requises.

De plus, il a été constaté que les problèmes majeurs rencontrés lors de son adoption ne sont pas que techniques mais aussi culturels et sociales. Des méthodes doivent donc être trouvées pour rassurer les professionnels de santé quant à l'utilisation de la simulation.

Pour terminer, la généricité et la réutilisabilité telles que nous les avons abordées ici permettent de se faire une idée des impératifs qui pèseront sur nos divers outils. Néanmoins, il est important de noter que pour un modèle de simulation, la validité et la crédibilité sont deux points très importants. En effet, un modèle peu fidèle à la réalité peut-être tout autant utile qu'un modèle très détaillé et entièrement calqué sur la réalité mais qui demande un temps de développement astronomique. La validité et la crédibilité d'un modèle ne s'évalueront donc pas que de par sa taille ou de par ses options mais aussi suivant le nombre de tests auxquels il sera soumis. Bien entendu, plus il en réussira, plus il sera crédible et valide. Il sera donc surtout évalué sur sa capacité à répondre aux questions prédéfinies par le « client ».

Mais, la simulation est-elle intensivement utilisée dans le monde hospitalier ? Voyons cela dans le chapitre suivant.

III

OÙ EN EST LA SIMULATION ?

RESUME DU CHAPITRE 3

Le chapitre précédent ayant placé la simulation comme l'une des solutions les mieux adaptées à l'évaluation des évolutions structurelles et organisationnelles devant gérer le circuit du médicament, nous verrons ici quelle est réellement sa place dans le domaine hospitalier.

Ainsi, ce chapitre se concentrera dans un premier temps à une analyse de la situation de la simulation hospitalière d'un point de vue général, avant de se focaliser plus précisément sur les travaux liés à la simulation de la pharmacie et plus particulièrement à ceux sur le circuit du médicament.

SOMMAIRE DU CHAPITRE

3.1	INTRODUCTION	92
3.2	SITUATION DE LA SIMULATION HOSPITALIERE	92
3.2.1	Une évolution marquée.....	93
3.2.2	Quelles tendances dans la simulation ?	94
3.2.3	Qu'en est-il du CDM ?.....	100
3.3	CONCLUSION	103

3.1 INTRODUCTION

En 1995, [LEAPE 95] avance que les processus de dispensation, de distribution ainsi que leurs homologues ne sont pas assez efficaces dans le monde hospitalier, cela étant dû en grande partie à un manque de formalisme et de représentation. En réponse à cette constatation, des avancées notables ont eu lieu dans le domaine. Cependant il reste de nombreux efforts à faire afin de répondre avec efficacité aux problématiques auxquelles sont soumis ces processus. La modélisation et la simulation permettront dans ce cas de répondre à ces questions en apportant une vision globale de la problématique et en permettant de tester des solutions.

La simulation a été utilisée intensivement afin d'éprouver et d'améliorer les systèmes de distributions dans le monde hospitalier, et ceci notamment en Angleterre et aux Etats-Unis [JUN 99], [ROHLEDER 07], [COOPER 02]. Une revue de la littérature [FONE 03] révèle une prédominance des modèles liés aux problèmes opérationnels portant sur les urgences, la planification de soins, les parcours patient et les capacités de prise en charge. Les autres domaines du monde de la santé ne sont pas pour autant laissés de côté, cette étude classifiant suivant cinq grands axes les recherches qui y sont menées : Organisation et planification ; Evaluation de la performance économique ; Infection et propagation ; Observation, contrôle des performances et enfin, Divers.

Bien que les premières applications de la simulation remontent aux années 1960 [FETTER 65], la plupart des travaux sur la simulation hospitalière sont plus récents. Une grande part de l'explication de ce phénomène trouve sa place dans l'évolution technologique, l'accroissement des performances des ordinateurs et les nouveautés logiciels en découlant. Ces changements ont permis d'agrandir le champ des possibilités dans le domaine de la simulation et de la modélisation, non seulement en termes de puissance de calcul pure mais aussi du point de vue de l'accessibilité, en permettant une plus grande flexibilité pour les professionnels et une plus grande simplicité pour les profanes. [BRAILSFORD 09] a d'ailleurs noté que parmi les 163 papiers sur la recherche opérationnelle analysés, la simulation reste la seconde technique la plus usitée après l'analyse statistique pour résoudre ce type de problème.

Comme nous le verrons par la suite, il y a eu plus qu'une évolution significative du nombre d'études dans ce domaine mais il serait attendu que de nos jours la modélisation soit un élément essentiel et routinier des politiques de management et d'évaluation des systèmes de santé, ce qui n'est malheureusement encore que trop peu le cas.

3.2 SITUATION DE LA SIMULATION HOSPITALIERE

Depuis les années 1960, l'utilisation de la simulation s'est rapidement développée dans le monde hospitalier [BARNES 97], [JUN 99], étant donné que de nombreux modèles de simulations des hôpitaux, ou de certaines de leurs composantes plus exactement, ont été conçus. Les papiers, articles ou autres publications sur ces réalisations ont déjà en grande partie été passés en revue par un certain nombre d'auteurs sur lesquels nous nous appuyerons pour mettre en évidence la situation de la simulation dans le monde hospitalier. Parmi ces nombreuses revues de la situation, sept d'entre elles nous serviront de base pour couvrir un maximum d'articles sur le domaine :

- ❖ [LENT 12] ;
- ❖ [BRAILSFORD 09] ;
- ❖ [FONE 03] ;
- ❖ [JUN 99] ;
- ❖ [GUNAL 10] ;
- ❖ [THORWARTH 09] ;
- ❖ [BRAILSFORD 11].

Le tableau suivant présente le nombre d'articles et la période couverte par ces revues :

Tableau 3.1 : Articles passant en revue la simulation hospitalière

Article	Nombres articles étudiés	Période couverte
[LENT 12]	89	N.C
[BRAILSFORD 09]	342	1952-2007
[FONE 03]	182	1980-1999
[JUN 99]	117	1952-1997
[GUNAL 10]	N.C	N.C
[THORWARTH 09]	62	2003-2007
[BRAILSFORD 11]	233	1975-2009

3.2.1 Une évolution marquée

Au fil du temps le nombre de recherches portant sur cette thématique s'est largement développé. Apparue dans un premier temps aux Etats-Unis – 60% des papiers étudiés par [FONE 03] en proviennent avant 1990 – celle-ci s'est ensuite fortement implantée dans les pays européens. Sur la période étudiée par [BRAILSFORD 09] – 1952-2007 – le résultat est similaire. De ces précédentes études, il est donc possible de tirer un schéma d'évolution du nombre de publications à travers le temps. La Figure 3.1 répartit donc celles-ci sur la période 1952 -2007 :

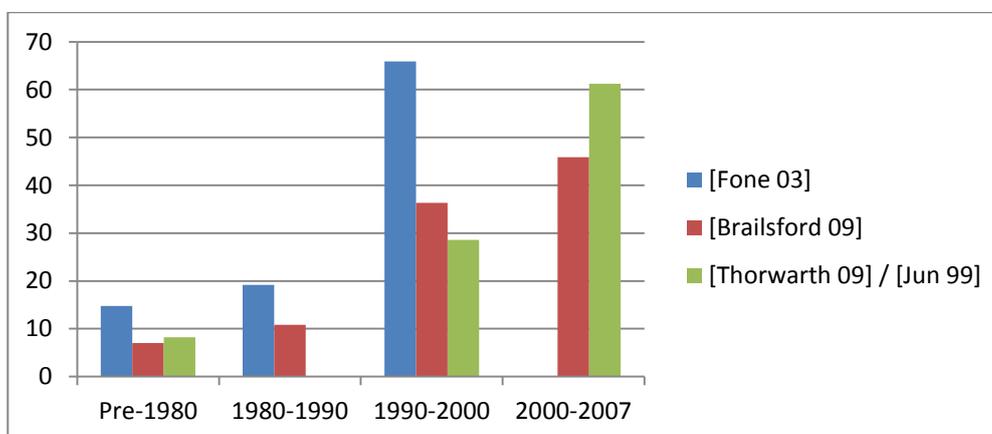


Figure 3.1 : Nombre de publications par période

Pour [LENT 12] qui s'est concentré sur la période après 2000, le nombre de papiers issus de sa recherche thématique est beaucoup plus conséquent. En effet, les termes utilisés pour ces recherches étaient DES, patient, soin et hôpital. Contrairement aux autres qui montrent une évolution au fil des âges, [LENT 12] met en évidence un surcroît d'intérêt pour la simulation après 2005 comme il est possible de le voir :

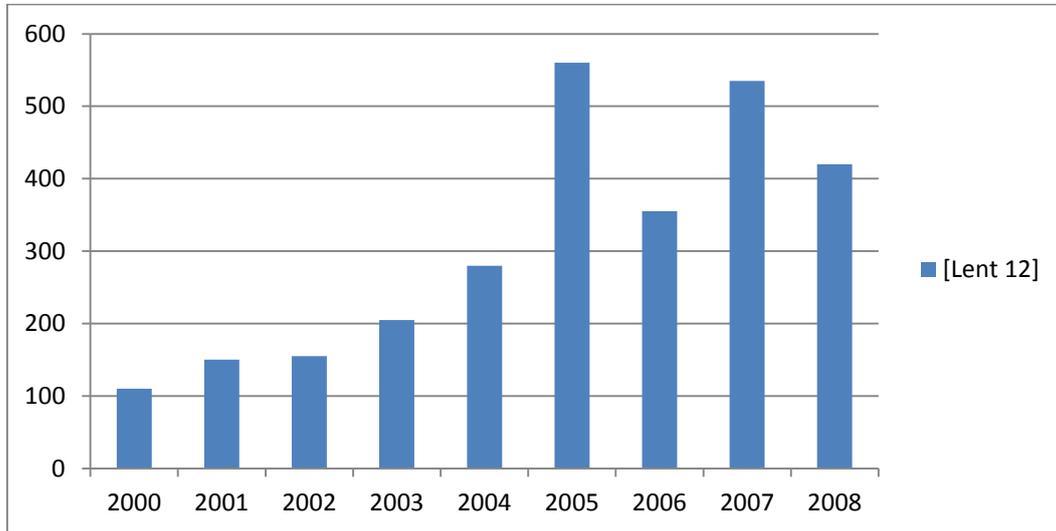


Figure 3.2 : Evolution du nombre de publications pour [Lent 12] après 2000

La simulation a donc trouvé sa place dans le monde hospitalier, mais comme nous allons le voir, celle-ci ne se répartit pas uniformément sur toutes les problématiques connues dans ce domaine.

3.2.2 Quelles tendances dans la simulation ?

3.2.2.1 Existe-t-il des modèles d'hôpitaux ?

En toute logique, une revue des différents modèles de simulation produits sur le domaine hospitalier devrait permettre d'observer autant d'hôpitaux que de services (urgences, obstétrique, etc.) modélisés. Dans un premier temps, il est constaté avec un certain étonnement que cela n'est pas le cas. Une grande majorité des articles publiés se concentre sur un ou plusieurs services comme le met en évidence [FONE 03], mais encore [LENT 12] qui, avec ses 89 papiers étudiés, en rencontre 68 dédiés à un service et donc 21 à de multiples services. Cependant, ces derniers ne portent que très rarement sur plus de deux services à la fois. Il y a donc un fort manque à ce niveau comme le note [JUN 99].

Pour [GUNAL 10], la conclusion est identique. En effet, pour celui-ci, la plupart des études ne se concentrent que sur un unique service et sont donc spécifiques à un hôpital empêchant dès lors une réutilisabilité par d'autres structures.

Mais pourquoi une telle constatation ?

Il paraît clair que la simulation est elle-même en partie responsable de ce manque de modélisation de par ses contraintes inhérentes (voir Chapitre 2), mais elle n'est pas seule responsable. Deux pistes sont avancées pour expliquer ce manque :

- ❖ La complexité des hôpitaux : de par le nombre d'intervenants et des diverses entités en présence, un modèle de simulation complet d'une telle structure est très difficile à concevoir et demande un temps de développement et des besoins financiers conséquents. De plus, un modèle ne peut être qu'une vue simplifiée de la réalité, et contrairement au sens propre de ce terme, il est compliqué d'obtenir la bonne simplification sans omettre des éléments importants [FONE 03]. Il sera donc préférable de se concentrer sur certaines portions de l'hôpital ;
- ❖ L'autre explication avancée par [JUN 99] est due à la collecte des données nécessaires à la création d'un tel modèle, récolte se heurtant souvent à la dissémination à travers tout l'hôpital des informations requises et à la manière dont celles-ci sont stockées.

Au final, obtenir un modèle représentant avec exactitude un hôpital n'est qu'utopie du fait de la difficulté qu'un tel développement rencontrerait mais aussi du fait qu'un modèle est et restera toujours une simplification de la réalité [PIDD 09].

Parmi les quelques exemples qui existent et qui se rapprochent le plus d'une modélisation hospitalière dans sa globalité, reportons nous aux travaux de [BRAILSFORD 04] et [COCHRAN 06].

↪ [BRAILSFORD 04]

Parmi les quelques applications existantes de simulation basée sur les systèmes dynamiques, les auteurs se sont portés ici sur la gestion des patients dans la ville de Nottingham. Prenant en compte autant les patients à caractère urgent (issus des urgences, arrivant par ambulance, ou provenant d'autres structures) que les patients venant sur demande d'un médecin généraliste, la grande particularité de ce modèle est de prendre en considération plusieurs structures hospitalières à la fois et par conséquent plusieurs services. Une dizaine de services divers sont donc gérés comme il est possible de le voir ci-dessous sur la vue conceptuelle du réseau « urgences » de la ville de Nottingham.

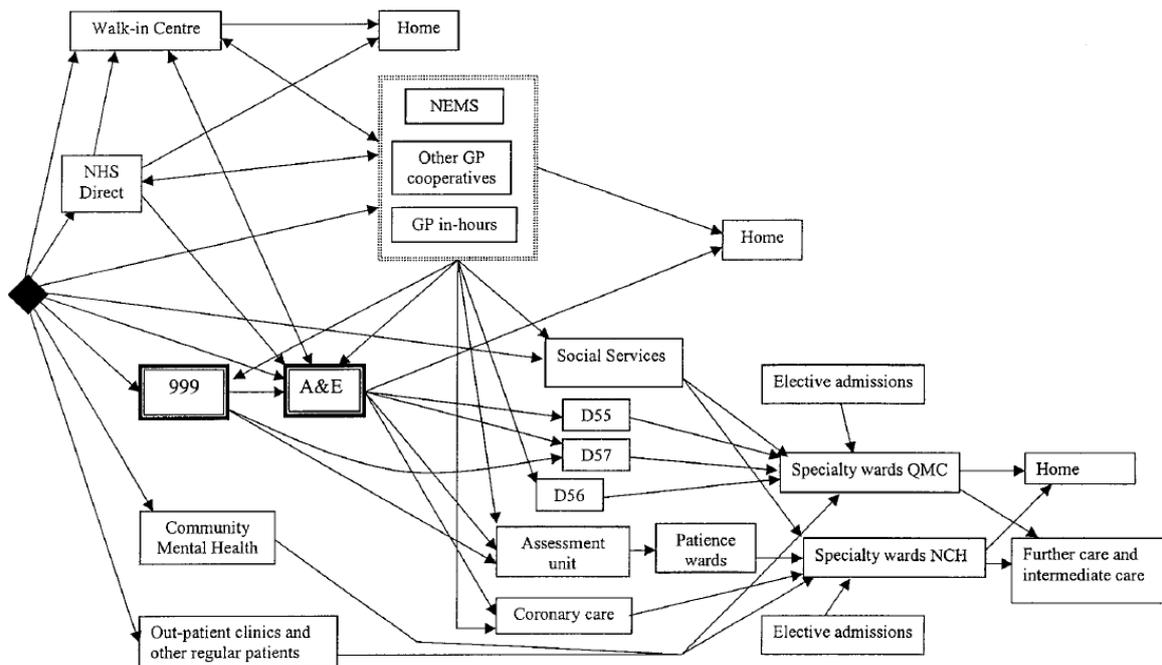


Figure 3.3 : Modèle conceptuel du réseau « Urgences » de la ville de Nottingham

Le modèle de simulation développé pour traiter ce problème se composait donc d'un modèle global se basant sur des sous-modèles pour fonctionner. Chacun d'entre eux se basait sur un ensemble de paramètres (âge patient, nombre de patients, jour de la semaine, etc.) pour faire évoluer sa situation et fournissait des données sur les flux y transitant, la situation des stocks, etc.

↳ [COCHRAN 06]

Se servant de la simulation à événements discrets, le modèle ici développé avait pour objectif d'équilibrer l'utilisation des 400 lits d'un hôpital, afin d'éviter la mobilisation inutile de cette ressource précieuse. Dans le but de résoudre cette problématique, les auteurs ont choisi de coupler à la DES une gestion des files d'attente permettant de compléter l'équilibrage fait.

Ce modèle se concentre donc uniquement sur le taux d'occupation des lits, contrairement au précédent plus large, mais gère l'entièreté des lits d'un seul et unique hôpital en tenant compte des spécificités de chaque service.

3.2.2.2 Des thèmes et services privilégiés ?

En observant les nombreuses contributions, de nombreux auteurs ont constaté que certains services et domaines des hôpitaux se retrouvaient plus souvent étudiés et modélisés que d'autres. Cette constatation est en partie dû au fait que les publications sont majoritairement consacrées à un service ou à un problème en particulier, ce qui est notamment le cas de ceux avec les plus grandes contraintes (exemple les urgences). De plus ces travaux sont souvent menés ou initiés à la demande du service subissant ces problèmes, axant dès lors le travail sur celui-ci et mettant donc de côté l'aspect transversal de l'hôpital.

Parmi les auteurs passant en revue la situation de la simulation dans le monde hospitalier, plusieurs d'entre eux catégorisent les différentes publications.

↳ [FONE 03]

Dans cet article, les auteurs identifient cinq grandes catégories pour classer les papiers :

- ❖ Organisation et planification ;
- ❖ Infection et propagation/dissémination ;
- ❖ Evaluation des performances économiques;
- ❖ Observation, contrôle des performances ;
- ❖ Divers.

Parmi celles-ci, il ressort qu'il y a une prédominance des travaux sur les problèmes de planification et d'organisation (51.6%), et sur l'observation et le contrôle des activités (24.2%).

Du côté des services, les auteurs ne mettent pas évidence un service particulier, mais reconnaissent que la problématique de la gestion des arrivées reste l'une des plus traitées et que la plupart des modèles sont en lien avec les blocs opératoires et, les urgences.

↳ [BRAILSFORD 09] & [BRAILSFORD 11]

Pour ces deux articles, les auteurs ont trié les contributions suivant neuf catégories. Les premiers résultats sont issus des données du « Research Into Global Healthcare Tool » (noté **RIGHT**) et le second compare ceux de l'« Operational Research Applied to Healthcare Services » (noté **ORAHS**) à ceux du **RIGHT** :

- ❖ Finance, stratégie, régulation et gouvernance ;
- ❖ Santé publique, offre de soins ;
- ❖ Patient (comportement, caractéristiques) ;
- ❖ Planification, utilisation des ressources, du système ;
- ❖ Qualité, surveillance des performances et diagnostic ;
- ❖ Management du risque et prévision ;
- ❖ Management du personnel et des effectifs ;
- ❖ Recherche ;
- ❖ Divers.

Cette fois-ci les deux domaines prédominants sont de nouveau la planification et l'utilisation des ressources avec 21.4% dans le cadre du RIGHT et 36% pour ORAHS. Le second thème est quant à lui « Economie, stratégie et gouvernance » pour RIGHT (18.1%) alors que pour ORAHS trois thèmes sont à égalité (environ 14%) : Recherche ; Qualité, surveillance des performances et diagnostic ; Finance, stratégie, régulation et gouvernance.

↪ [THORWARTH 09]

Pour celui-ci, les articles étudiés peuvent être classés en cinq catégories principales, l'une d'entre elles pouvant être découpée en quatre sous-catégories. Ils ont donc obtenu :

- ❖ Hôpital :
 - Unités de soins spécifiques (cardiologie, orthopédie, gériatrie, etc.) ;
 - Procédures chirurgicales ;
 - Hôpital en général ;
 - Unités de soins intensifs.
- ❖ Chaîne logistique hospitalière ;
- ❖ Urgences ;
- ❖ Prévision des besoins en soin (sur un territoire, etc.) ;
- ❖ Hôpital de jour.

Ici deux sujets prédominent. Le premier est celui traitant des prévisions des besoins en soins (20%) tandis que le second concerne les procédures chirurgicales avec 18% des articles étudiés.

↪ [GUNAL 10]

Comme ces prédécesseurs, les auteurs ont ici mis en évidence que certains services ou problématiques sont plus étudiés que d'autres. Ainsi, les sujets les plus abordés par ordre d'importance sont pour ces auteurs les urgences, le processus de soins pour les patients hospitalisés, l'hôpital de jour, les diverses unités de soins spécifiques et enfin les hôpitaux dans leur globalité.

Ils énoncent de plus qu'une nouvelle fois, les problèmes de planification en particulier pour l'hospitalisation de jour restent le sujet le plus majoritairement abordé.

En plus de ces revues sur la situation de la modélisation hospitalière, il est possible d'associer [LAGERGREN 98] qui, tout comme ces différents auteurs, a mis en évidence un certain nombre de grands axes pour classer les modèles hospitaliers : Epidémiologie et prévention ; Conception des systèmes de santé ; Amélioration des systèmes de santé et enfin Aide à la prise de décision médicale.

De même, [LENT 12] indique que les articles traitent en majorité le cas des urgences (20%) suivi des blocs opératoires (17%) et enfin des consultations (15%).

Enfin, [JUN 99] a aussi identifié deux domaines hospitaliers où les travaux se font plus intenses et qui concernent la gestion des flux patients et l'allocation des ressources.

En se basant sur les grands thèmes énoncés par [LAGERGREN 98], [FONE 03] et [BRAILSFORD 09], il est possible de construire une « généalogie » rapide de ces derniers pour obtenir une synthèse des idées émises (Figure 3.4). Un certain consensus apparaît dans les grands thèmes que ces articles abordent (pouvant être retrouvé dans la plupart des revues du domaine), avec quand même une évolution de ceux-ci au fil des parutions et de l'étoffement de la littérature.

Au final, il ressort de ces différentes revues que la planification reste la problématique majeure abordée par la simulation dans le domaine hospitalier et notamment dans le cadre de la gestion des patients aux urgences et en hospitalisation de jour, là où le turnover est donc le plus présent. Ceci s'explique aussi par le fait que ces deux domaines restent assez contenus, facilement observables et ont des périodes temporelles assez courtes.

3.2.2.3 Une variété trop importante

Bien que la simulation soit un puissant outil pouvant répondre à des problèmes divers et variés, il n'en est pas forcément de même pour le monde hospitalier. De nombreuses lacunes apparaissent dans son utilisation.

Tout d'abord, la plupart des approches de modélisation ne portent pas assez leur attention sur la formulation, la structuration du problème, chose la plus importante pour la compréhension de celui-ci. Malgré le fait que cette étape peut être considérée en dehors du cadre pour développer un modèle de simulation, celle-ci en fait pourtant partie intégrante.

En regardant de plus près les problèmes hospitaliers et leur complexité, de nombreuses situations où le modèle de simulation n'est utilisé que pour résoudre un problème bien précis est observé. Ceci rentre quelque peu en contradiction avec l'esprit de la simulation, à savoir l'idée de prospection et de tests. Ainsi comme le note [BRAILSFORD 09], parmi le grand nombre de méthodes présentées dans la littérature, beaucoup d'entre elles sont beaucoup trop spécifiques pour être généralisées et réutilisées. De plus il note comme [JUN 99] que la simulation est majoritairement utilisée pour planifier l'allocation de ressources.

A cela se greffe le problème de récolte des données nécessaires à la simulation. Pour le monde hospitalier, les données récoltées sur le terrain ne sont pas assez fiables pour se permettre de les qualifier de fil conducteur, de base pour le modèle. Les deux raisons principales d'une telle défiance vis-à-vis de ces données sont que celles utilisées pour des prévisions à long terme sont issues d'archives. Depuis l'obtention de ces données, l'environnement hospitalier a évolué autant technologiquement, qu'économiquement ou stratégiquement. Dès lors la valeur de ces données est quasi nulle pour certaines d'entre elles. La seconde raison est humaine, la récolte de données sur le terrain étant souvent anxiogène pour les professionnels de santé et les données livrées ne sont pas toujours totalement objectives.

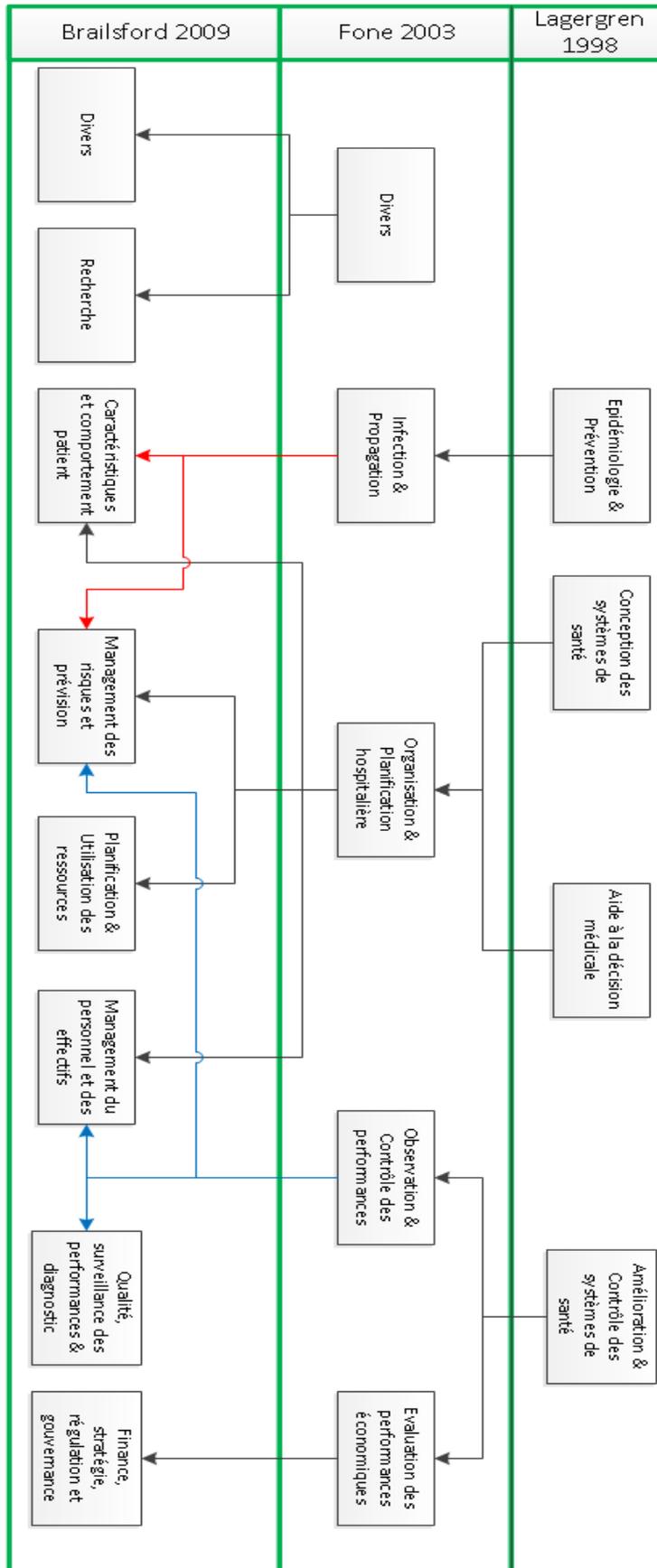


Figure 3.4 : Généalogie des thématiques d'études sur les hôpitaux

D'autres problèmes soulevés par [LENT 12] sont aussi à souligner. Il met en évidence que parmi les nombreuses études en lien avec la simulation, peu de modèles sont réellement implémentés et encore moins étudient l'impact des changements recommandés par les résultats issus des modèles. Ainsi beaucoup trop d'articles se concentrent sur la construction du modèle de simulation et pas assez sur ce que ce modèle pourrait apporter comme amélioration à l'hôpital.

Enfin, il est impératif que les hypothèses faites sur le domaine d'étude soient indépendantes du modèle lui-même, que celles-ci puissent être suivant la volonté de l'utilisateur désactivées ou pas et qu'elles puissent être réévaluées de façon probabiliste si nécessaire. De cette manière, il est possible pour les utilisateurs de tester ces hypothèses au lieu d'en être dépendant pour la construction du modèle et la résolution du problème. [PIDD 09] illustre cette idée en affirmant que le modèle doit superviser, contrôler les données et non pas l'inverse.

Pour illustrer la difficulté d'intégration de la simulation dans le domaine hospitalier, utilisons ce workshop suivant, [MASHNET 2007] qui a mis en évidence une différence flagrante d'attente vis-à-vis des modèles entre managers du « National Health Services » (noté NHS) et les chercheurs. D'un côté les personnes « terrain » comme les managers voudront étudier des problèmes très précis sur des échelles de temps souvent courtes contrairement aux chercheurs qui viseront des généralités et cela sur plusieurs années. Il est donc évident que les modèles de simulation à produire et les résultats attendus seront différents. Un dernier point important à souligner est que les résultats des modèles de simulation sont le plus souvent indicatifs et non pas à prendre au pied de la lettre, difficulté supplémentaire à faire prendre en compte aux non-initiés.

3.2.3 Qu'en est-il du CDM ?

Bien que le sujet qui nous intéresse en priorité concerne le circuit du médicament, nous ne nous limiterons pas à ce seul thème dans un premier temps, mais nous considérerons cette notion dans sa globalité c'est-à-dire la pharmacie hospitalière.

3.2.3.1 Un service laissé de côté ?

Des nombreuses revues sur la situation de la simulation hospitalière, il est possible de tirer certaines conclusions. Comme attendu, le domaine de la pharmacie est peu sollicité dans la simulation en comparaison des autres domaines, chose un peu étonnante, celui-ci étant un domaine offrant de nombreuses perspectives et problématiques variées. En effet, une pharmacie hospitalière est un service complexe qui doit gérer un grand nombre de demandes différentes, qui est constitué d'une palette de rôles très variée et qui utilise de nombreuses solutions technologiques pour améliorer la sûreté et la vitesse de dispensation. C'est notamment une constatation faite dans la revue de [LENT 12], où seul un papier incluait la pharmacie dans le développement d'un modèle de simulation.

De plus, parmi les études réalisées, les problématiques se concentrent assez souvent sur la gestion des patients aux guichets pour la rétrocession des médicaments [HENDERSHOTT 96], [REYNOLDS 11].

Cependant comme nous allons le voir, la pharmacie n'est pas complètement ignorée.

3.2.3.2 Quelle est la situation de simulation du CDM?

Pour celui-ci, les travaux restent encore rares, même si un certain nombre d'articles sont apparus depuis les années 2000 du fait du développement des capacités de simulation. C'est une constatation faite par [BATES 00] pour qui, à l'époque, la plupart des systèmes en place dans les hôpitaux n'ont pas été réellement formalisés, et notamment le circuit de distribution des médicaments.

Afin d'y voir plus clair, découpons la période en deux : avant 2000 et après.

↪ Avant 2000 :

- ❖ [DEAN 97] ;
- ❖ [MUKHERJEE 91] ;

↪ Après 2000 :

- ❖ [WONG 03] ;
- ❖ [OUZAYD 12] ;
- ❖ [GOURGAND 07] ;
- ❖ [HUET 11] ;
- ❖ [YURTKURAN 08].

Chacun de ces papiers abordant la problématique sous un angle différent, nous ne nous intéresserons qu'à trois d'entre eux.

Le premier est celui de [WONG 03], qui a travaillé sur la remise en forme du processus de délivrance des médicaments par l'intégration d'un nouveau SI à l'hôpital « Sunnybrook and Women's College Health Science Center » (SWCHSC) à Toronto. Ce travail traite de chaque partie du CDM, et propose de plus une modélisation « avant » (Figure 3.5 (a)) et « après » (Figure 3.5 (b)) du processus afin d'en faire une comparaison portant sur le temps de réponse (analyse de la prescription, préparation et envoi) et sur le taux d'erreur liés à ces actions. La comparaison des deux modèles permet de constater une réduction de l'ordre de 50% du temps de réponse et d'environ 70% des erreurs commises.

Les deux modèles obtenus sont les suivants, et sont issus de l'utilisation du logiciel PROMODEL, permettant donc d'obtenir un modèle graphique.

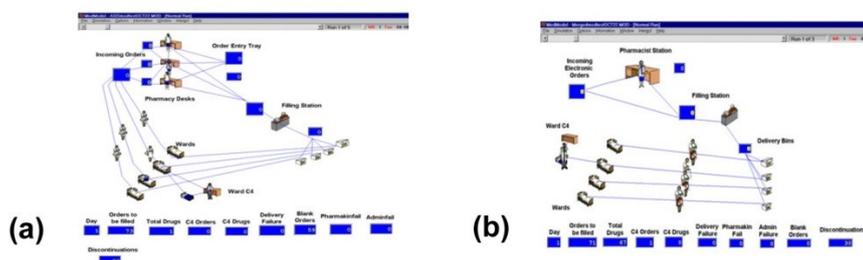


Figure 3.5 : (a) Modèle du circuit actuel (AS-IF) ; (b) Modèle futur du circuit (TO-BE)

Le second papier est celui de [OUZAYD 12] qui traite lui aussi tout le processus du CDM de la prescription à la dispensation. Les auteurs se sont cette fois-ci orientés vers l'utilisation des réseaux de Pétri colorés, et fournissent une analyse des performances de leur modèle visible en partie dessous.

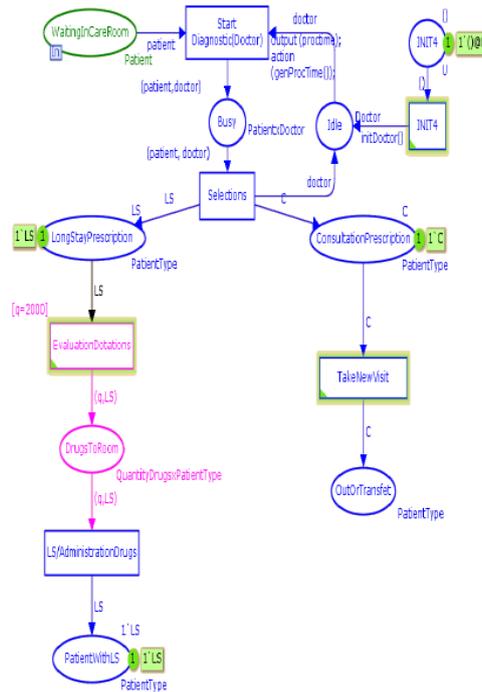


Figure 3.6 : Modèle en réseau de Pétri coloré du CDM

Le dernier est quant à lui un travail de thèse [HUET 11] qui propose une méthodologie de réingénierie pour un contrôle par le produit du CDM. Ce travail se termine par l'application de cette méthode à la pharmacie du CHU de Clermont-Ferrand et le développement d'un modèle de simulation sous AUTOMOD.

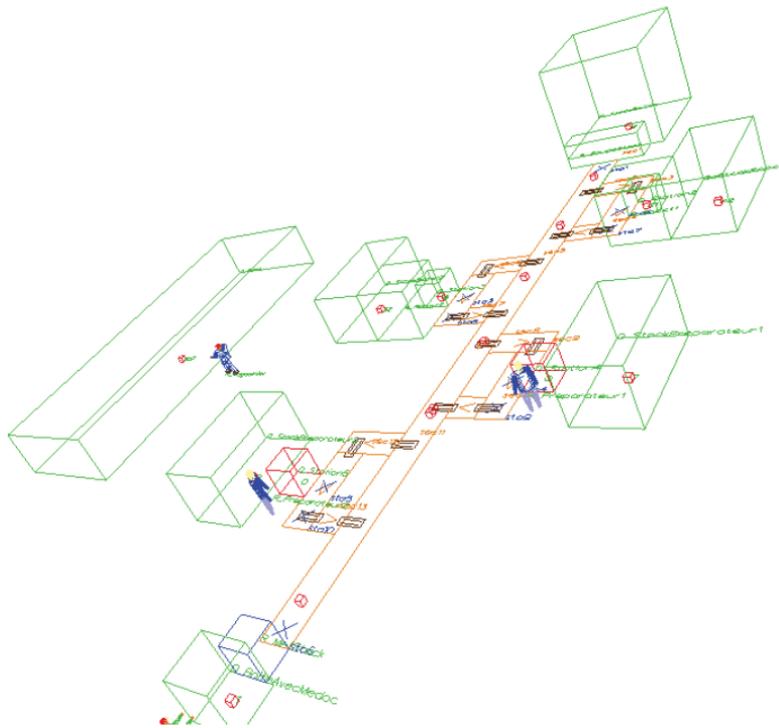


Figure 3.7 : Modèle AUTOMOD de la pharmacie par Huet (2011)

3.3 CONCLUSION

Bien que la simulation soit de plus en plus usitée dans le domaine hospitalier, seules quelques études peuvent se targuer d'étudier avec soin le CDM et ses différentes composantes. Le peu de retour et de résultats sur cette problématique ne permet donc pas de dresser ou de donner des pistes pour simuler de manière efficiente ce circuit.

Parmi les articles traitant des problèmes du CDM de ces dernières années, une grande partie l'aborde avec une approche de modélisation. Cependant, ces articles se concentrent beaucoup sur l'aspect mathématique en adoptant des algorithmes d'optimisation comme les heuristiques, métaheuristiques, modèles mathématiques exacts ou encore graphiques, chose limitante pour nous, ces aspects n'étant pas toujours appréhendables de manière évidente par les praticiens du domaine hospitalier. Il est à noter qu'une partie d'entre eux s'intéresse grandement à la construction d'une méthodologie, appliquée par la suite à un exemple concret, articles ayant donc un point commun avec le travail que nous développons. Mais le défaut majeur de ces différentes contributions est de se placer d'un point de vue ingénieur, concepteur, ne permettant pas aux utilisateurs de comprendre la démarche de conception en les en excluant en très grande partie (voire complètement). Il sera donc nécessaire de fournir une démarche simplifiée aux pharmaciens (utilisateurs dans notre cas) pour que ceux-ci réalisent cette tâche d'eux-mêmes (Chapitre 5), mais aussi d'apporter des outils génériques et simples d'utilisation (Chapitre 6 et 7).

De plus, nous ne pouvons que nous réjouir de voir la simulation prendre une telle place dans un domaine qui en est pourtant tant éloigné dans l'absolu. Une des principales raisons qui a fait que la simulation est devenue populaire pour résoudre des problèmes dans ce domaine est que celle-ci se prête à une analyse dynamique et non statique de la situation.

La simulation semble donc être la solution privilégiée pour tester des scénarios et configurations que l'on ne pourrait faire fonctionner sans moyens conséquents, ce qui valide un peu plus notre objectif de départ, à savoir faire de celle-ci notre outil d'aide à la décision.

Mais les avancées de la simulation ne doivent cependant pas cacher encore les nombreuses déficiences et réticences de la part des établissements qui ne veulent pas se lancer, et cela de manière fort logique, dans des travaux souvent coûteux aux résultats incertains s'ils sont mal conduits. C'est pourquoi par la suite nous proposons une démarche simplifiée et rapide à mettre en œuvre.

IV

ASCI & BPMN

RESUME DU CHAPITRE 4

Avant de se pencher sur les différents travaux que nous avons réalisés, ce chapitre a pour objectif de présenter de manière approfondie, mais sans pour autant être exhaustive, les deux clés de voûte de nos recherches, à savoir ASCI (Analyse, Spécification, Conception et Implémentation) et BPMN (Business Process Model and Notation ou Business Process Modeling Notation), l'objectif final étant de comprendre comment la méthodologie ASCI fonctionne, le pourquoi de BPMN et sa structure.

Alors que, comme nous le verrons, l'utilité d'ASCI n'est plus à démontrer dans le monde hospitalier, ce n'est nullement le cas de BPMN, encore très peu sollicité dans ce domaine. Présentant donc les étapes et les éléments constitutifs d'ASCI dans un premier temps, nous nous arrêterons ensuite sur le BPM (Business Process Management) et la notation liée, BPMN, dont on mettra en avant la dualité : rapidité de compréhension mais une utilisation simple et complexe à la fois.

SOMMAIRE DU CHAPITRE

4.1	INTRODUCTION	107
4.2	ANALYSE, SPECIFICATION, CONCEPTION ET IMPLEMENTATION	107
4.2.1	Etape 1 : Construction du modèle générique de connaissance	110
4.2.2	Etape 2 : Construction du modèle d'action pour un système	113
4.2.3	Etape 3 : Mise en œuvre du modèle d'action	114
4.2.4	Etape 4 : Action sur le système	114
4.2.5	Conclusion	114
4.3	BUSINESS PROCESS MODEL AND NOTATION	114
4.3.1	Le Business Process Management.....	115
4.3.2	BPMN, une histoire récente	115
4.3.3	Les éléments du langage	116
4.3.4	Les activités	117
4.3.5	Les évènements.....	122
4.3.6	Les Portes	125
4.3.7	« Pool » et « Ligne »	129
4.3.8	Les données.....	130
4.3.9	Les connecteurs.....	132
4.3.10	BPMN : son utilisation	133
4.4	CONCLUSION	136

4.1 INTRODUCTION

Comme déjà énoncé dans le Chapitre 2, nous allons utiliser la méthodologie ASCI (du fait de ses précédentes utilisations réussies) et le langage BPMN (ayant été évalué comme le plus adéquat) afin de construire une démarche et des outils permettant de modéliser les processus de la pharmacie et d'améliorer notre système. Cette méthodologie et ce langage étant les points d'orgues de la première phase de nos travaux, nous allons les présenter en détails afin d'en cerner tous les tenants et aboutissants.

Dans un premier temps, nous nous pencherons sur la méthodologie ASCI afin de détailler son contenu et son fonctionnement. Les différentes étapes de celle-ci seront donc présentées en précisant dans chacune d'entre elles quels outils et quelles phases (A, S, C ou I) sont utilisés.

Présenté succinctement dans la partie 2.3.1, le langage BPMN va être ici détaillé étant donné que lors de l'évaluation de différents langages, le résultat fut de sélectionner BPMN pour modéliser les processus de la pharmacie. Ce résultat est d'ailleurs en concordance avec ceux de [RECKER 09], où parmi les nombreux langages évalués, BPMN se classe second (voir premier si l'on ne considère que les langages évalués par nos soins).

Bien que le but premier de BPMN soit d'apporter une notation facilement compréhensible et utilisable par tous les acteurs impliqués dans les processus, de l'analyste business au manager, BPMN pose quelques problèmes d'interprétation et d'utilisation comme nous le verrons.

Ainsi, après la présentation de ASCI, nous aborderons le BPM d'une manière générale afin de comprendre le but de cette démarche, avant de faire un bref historique du langage BPMN et de poursuivre sur la présentation de sa symbolique graphique. Nous discuterons enfin comment BPMN est utilisé par les nombreux acteurs du BPM, ce qui mettra en lumière certaines informations intéressantes.

4.2 ANALYSE, SPECIFICATION, CONCEPTION ET IMPLEMENTATION

Créée dans le but de faciliter le développement d'outils d'aide à la décision, cette méthodologie a été proposée par Gourgand et Kellert au début des années 1990 [GOURGAND 91, GOURGAND92]. Depuis lors, celle-ci a été à de nombreuses fois réutilisée notamment dans notre domaine, le monde hospitalier et parmi ces utilisations il est possible de citer les travaux suivants :

- ❖ [CHAUVET 09] : ASCI est utilisée pour la modélisation de systèmes hospitaliers ;
- ❖ [EL HAOUZI 08] : ASCI sert de cadre méthodologique pour obtenir un nouveau type de pilotage (centralisé et décentralisé) de système dans le monde industriel ;
- ❖ [FENIES 06] : Dans le cadre de l'évaluation des performances d'une supply-chain (d'un point de vue économique et industriel), ASCI sert à la construction d'une suite logicielle ;
- ❖ [HUET 11] : ASCI est couplée à une approche holonique dans le but de construire une méthodologie de réingénierie pour le contrôle par le produit des systèmes manufacturiers. L'auteur met en application ses travaux sur le CDM d'un hôpital ;
- ❖ [RODIER 10] : ASCI sert de base pour le développement d'une méthodologie visant à modéliser les systèmes hospitaliers.

La réutilisation de cette méthodologie dans le monde hospitalier nous a donc incités à poursuivre sur cette voie, d'autant plus qu'ASCI supporte aisément des modifications dans sa structure et les outils qu'elle utilise.

Permettant de concevoir la modélisation d'une classe de systèmes, le modèle générique de connaissance de cette classe, et de réaliser une bibliothèque de composants logiciels exploitée pour générer un modèle d'action (programme informatique) pour un système de la classe, ASCI suit donc un déroulement logique dans sa mise en application. En effet, le cadre conceptuel de cette méthodologie s'appuie sur un processus de modélisation séparant explicitement le recueil de la connaissance de son exploitation. Elle propose une vision systémique qui préconise la décomposition hiérarchique d'un système avec le paradigme objet en trois sous-systèmes communicants et complémentaires (les sous-systèmes logique, physique et décisionnel).

Nous allons donc maintenant décomposer et expliciter chacun de ces objets dans le but de clarifier le déroulement et le contenu d'ASCI.

La Figure 4.2 donne le cadre général de la méthodologie ASCI où il est possible d'observer les quatre grandes étapes de cette dernière ainsi que les entités qui résultent de ces différentes étapes. Du côté gauche nous avons donc la démarche et du côté droit de cette figure la mise en œuvre de celle-ci. Sur cette figure apparaît une notion importante dans le cadre d'ASCI, à savoir le « domaine ». En effet, l'une des particularités de cette méthodologie est son approche des systèmes qu'elle catégorise par « domaine ».

Définition 4.1 : *Un domaine est une classe de systèmes qui regroupe un ensemble de systèmes comparables ayant les mêmes caractéristiques techniques et fonctionnelles. Ainsi, il est possible de décrire la classe des systèmes industriels, la classe des systèmes hospitaliers, etc.*

A noter que la notion de « sous-domaine » peut être utilisée pour caractériser un sous-ensemble d'un domaine, tel que le sous-domaine de la pharmacie, le sous-domaine des blocs opératoires, etc. existants dans le domaine des systèmes hospitaliers (Figure 4.1).

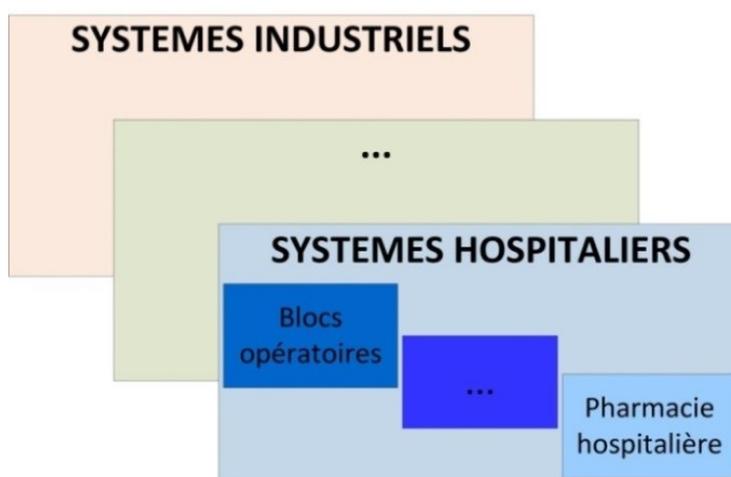
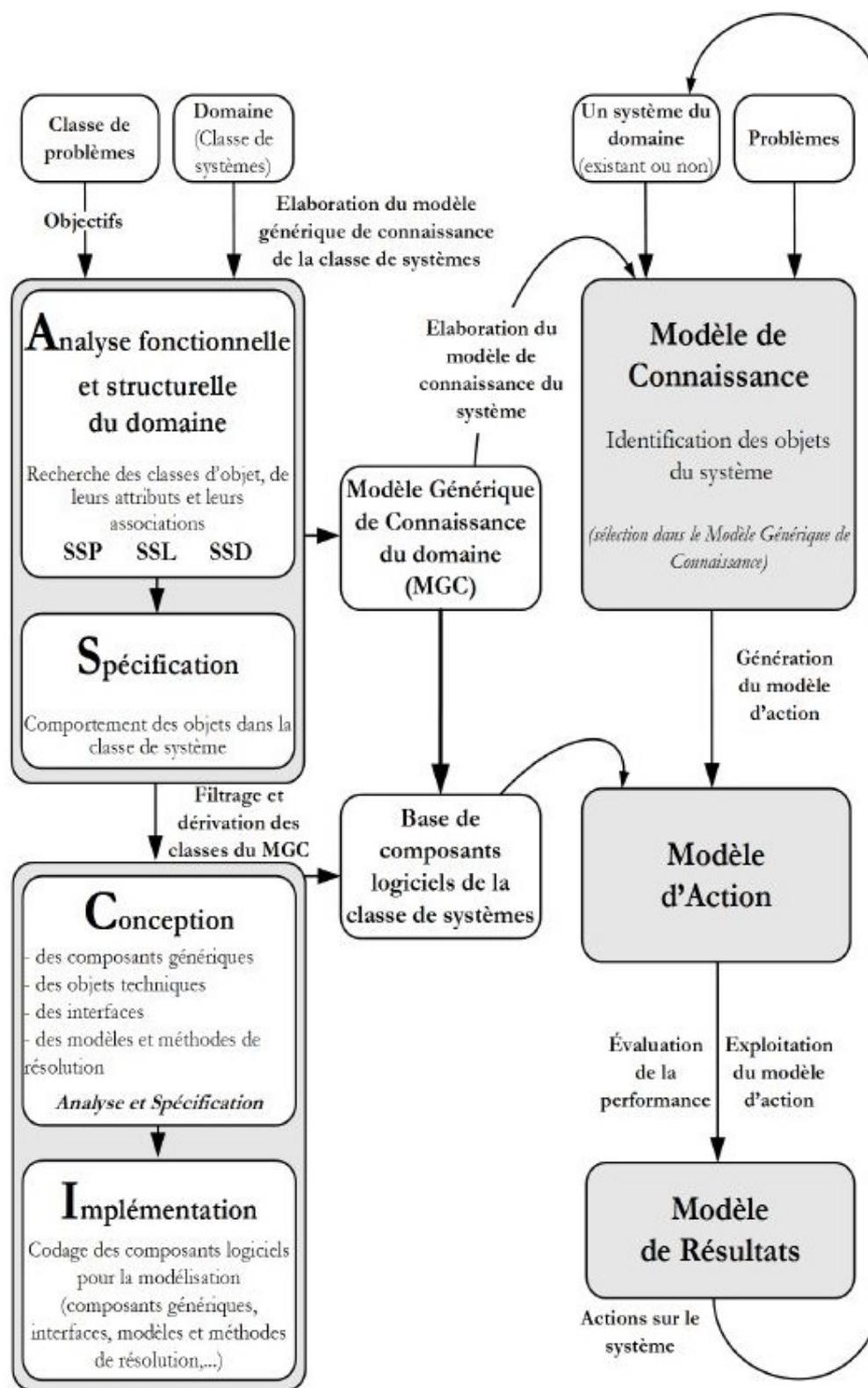


Figure 4.1 : Vue schématique des domaines et sous-domaines



Démarche ASCI proposée pour concevoir une méthodologie de modélisation et son environnement logiciel

Mise en œuvre du processus de modélisation avec la démarche ASCI

Figure 4.2 : La méthodologie ASCI d'un point de vue domaine (gauche) et du point de vue d'un système du domaine (droite)

ASCI s'intéresse donc à l'ensemble des systèmes ayant des caractéristiques structurelles et fonctionnelles communes, et rend alors possible la création et la conception de modèles d'actions pour tout système appartenant à ce domaine.

La Figure 4.3 ci-dessous, tirée de [RODIER 10], permet une vue en « cycle » d'ASCI, ou processus de modélisation, adapté à un système d'un domaine. Celle-ci permet d'appréhender plus facilement le déroulement de la démarche comparativement à la Figure 4.2. La présentation des divers éléments et étapes se fera donc sur une chronologie similaire à celle de la Figure 4.3 tout en faisant référence aux termes génériques de la Figure 4.2.

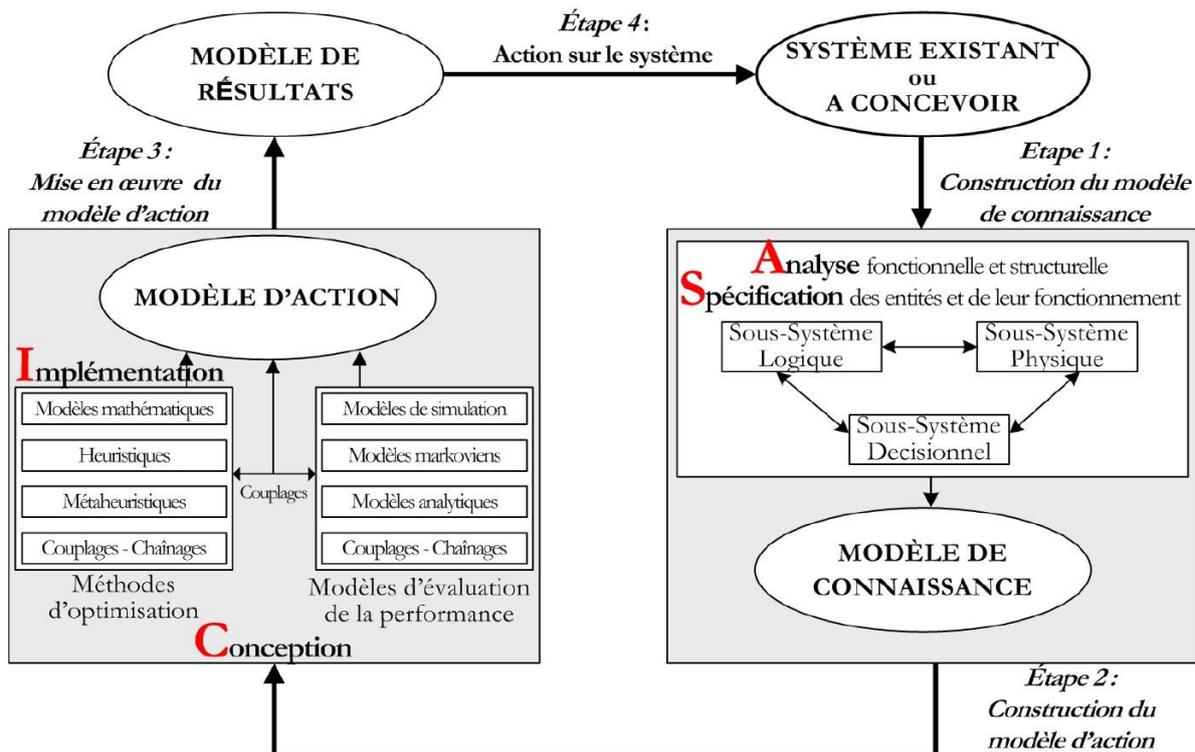


Figure 4.3 : Vue en cycle d'ASCI pour l'étude d'un système du domaine

4.2.1 Etape 1 : Construction du modèle générique de connaissance

Lors de la première étape d'ASCI, l'objectif est de concevoir un modèle générique de connaissance qui sera le résultat des deux premières actions menées dans ASCI à savoir l'analyse et la spécification. Celui-ci pourra dès lors être instancié sur tout système du domaine.

4.2.1.1 L'analyse fonctionnelle et structurelle

Le but de cette première action est de formaliser le domaine sous une forme graphique ou syntaxique. Ici les différents experts du domaine et de la modélisation travaillent de concert pour comprendre le domaine et identifier les différentes entités qui le composent, leurs fonctions ainsi que leurs associations. De plus, c'est à ce niveau que les objectifs de l'étude et les contraintes à respecter sont énoncés. Cette « analyse » donne donc une vue « statique » du domaine, celle-ci pouvant être issue, si nécessaire, de la décomposition du système en trois sous-systèmes

interdépendants permettant dès lors de faciliter la formalisation de la connaissance. Ces trois sous-systèmes seront présentés en partie 4.2.1.3.

4.2.1.2 La Spécification des entités et de leur fonctionnement

Cette seconde action consiste à exprimer le fonctionnement des systèmes du domaine, des flux présents dans le domaine et ainsi venir à les structurer. Il sera notamment fait cas du système de pilotage, ce dernier devant être décrit de manière précise afin de structurer et prendre en compte de nombreux éléments tels que les règles de gestion tout particulièrement mais aussi les règles de fonctionnement. C'est aussi ici que le format des données d'entrée devra être spécifié, toutes ces différentes tâches devant permettre aux experts (du domaine et en modélisation) de s'accorder sur le fonctionnement (réel ou désiré) du domaine.

4.2.1.3 Décomposition systémique des systèmes du domaine

Afin de pouvoir appréhender de manière plus aisée le domaine dans des cas où la formalisation se révélera délicate, il est utilisé une décomposition systémique dans ASCI. Comme déjà énoncé précédemment, celle-ci se caractérise par la création et l'utilisation de trois sous-systèmes interdépendants qui sont un sous-système logique, un physique et un dernier décisionnel.

[[RODIER 10](#)] énonce parfaitement les interdépendances (Figure 4.4) entre ces trois entités que nous allons présenter un peu plus en détails :

- ❖ Sous-système logique : aussi noté SSL, ce premier sous-système est constitué des nombreuses entités que le système doit traiter, de l'ensemble des opérations (élémentaires) concernant les flux, ainsi que des entrées dans le système qui s'y rapportent ;
- ❖ Sous-système physique : ou SSP, celui-ci est composé de l'ensemble des moyens, des entités physiques (actives ou passives) nécessaires à la réalisation des services élémentaires. Il contient de plus leur répartition géographique ainsi que les interconnexions existantes entre eux ;
- ❖ Sous-système décisionnel : Ce dernier aussi appelé sous-système de gestion est structuré en centre de gestion. Il recense donc l'ensemble des règles de gestion, d'allocation des ressources. De surcroît il permet de spécifier l'ensemble des règles de fonctionnement du système et les lois régissant le pilotage du système. Il est identifié sous le nom de SSD. Ce sous-système tient un rôle central, étant donné qu'il gère les interactions entre les 2 autres sous-systèmes. Il a pour rôle entre autres de :
 - Réceptionner les informations du SSP et SSL ;
 - Régir les comportements du SSL ;
 - Régir les actions du SSP.

Bien que la structure interne des sous-systèmes soit généralement hiérarchique, le SSD peut quant à lui posséder d'autres types de structures, telles que centralisée, décentralisée ou encore coopérative (et même des combinaisons de celles-ci). Ce cas particulier s'explique par le fait que le SSD est souvent le représentant de la complexité du système étudié.

Sur la Figure 4.4, il est bien mis en évidence que c'est le SSD qui assure le regroupement des informations, et définit donc les actions du SSP et SSL. Ces deux derniers communiquant principalement entre eux à travers l'activité, composante du SSL qui mobilise les ressources du SSP.

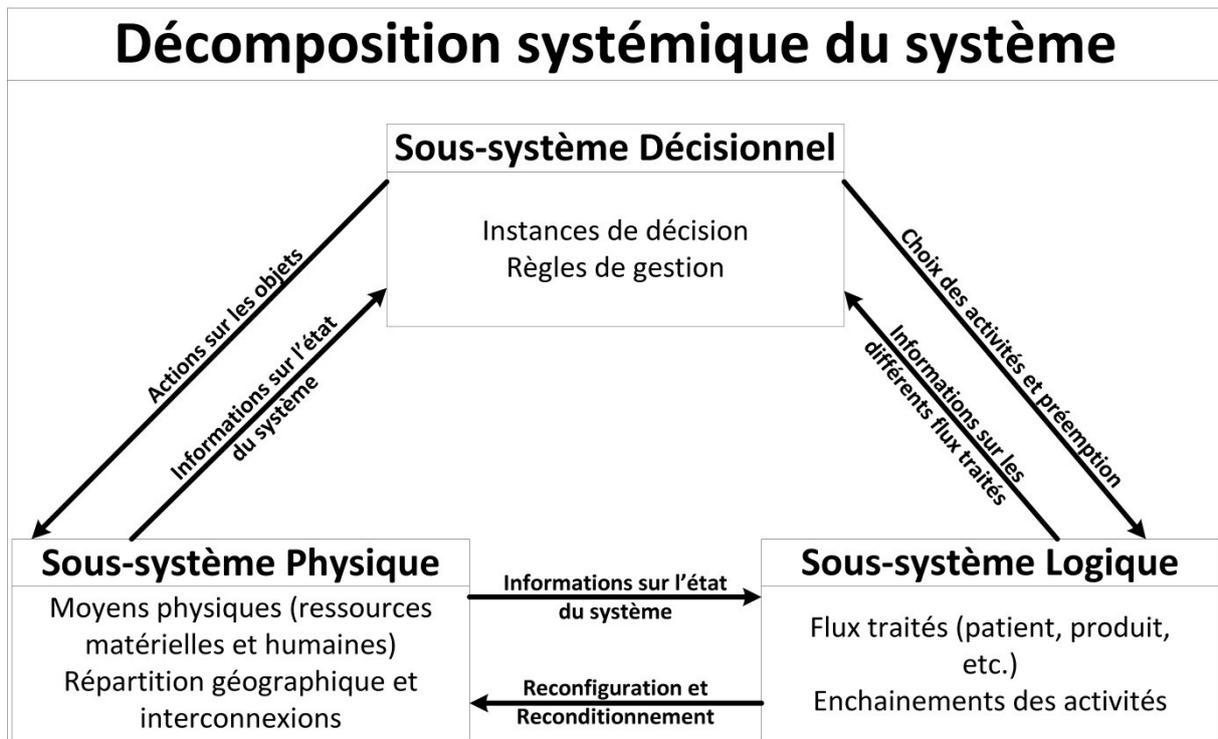


Figure 4.4 : Interactions entre les sous-systèmes d'ASCI

A noter qu'une telle décomposition a bien sûr été utilisée dans les références citées en début de chapitre. A ces deux-là, il est possible de rajouter [[COMBES 94](#)] (orienté hôpital) et [[GOUJON 97](#)] (orienté système industriel).

A travers les différents travaux réalisés, l'analyse et la spécification ont été menées avec divers langages ou méthodes. Pour l'analyse, on pourra citer les diagrammes entités/reliations, le diagramme des classes d'UML, quant à la spécification, il a été utilisé SADT, les RdP, mais encore le diagramme d'activité d'UML.

Maintenant que les deux premières actions menées dans ASCI ont été présentées, penchons-nous sur le résultat de celle-ci, à savoir le modèle de connaissance.

4.2.1.4 Le modèle générique de connaissance

Devant rester cohérent avec le domaine, quelles que soit les modifications et évolutions que celui-ci subira, le modèle générique de connaissance décrit la structure et le fonctionnement du domaine dans un langage naturel, formel ou graphique. La construction de ce modèle consiste en la récolte et la formalisation de la connaissance du domaine étudié, en se servant d'outils et de méthodes adaptés à cette démarche et aux experts du domaine.

A noter qu'un modèle de connaissance instancié (appliqué à un système du domaine) ne comportera pas nécessairement le même type d'informations suivant que celui-ci est la représentation d'un système existant (modèle a posteriori) ou la représentation d'un système en devenir (modèle a priori). En effet dans le premier cas, le modèle contiendra la connaissance acquise

après l'observation du système tandis que dans le second il contiendra les spécifications du futur système.

Le modèle générique de connaissance obtenu, son instanciation sur un système du domaine, permet d'en obtenir un modèle de connaissance, et il est alors possible de passer à l'étape 2 du processus de modélisation.

4.2.2 Etape 2 : Construction du modèle d'action pour un système

Afin de pouvoir exploiter le modèle précédemment développé, la création d'un modèle d'action est nécessaire. Traduction du modèle de connaissance dans un formalisme mathématique ou dans un langage de programmation, celui-ci servira donc à évaluer les critères de performances choisis par les experts, et il sera issu de la mise en application des deux dernières actions d'ASCI, à savoir la conception et l'implémentation que nous allons maintenant aborder.

4.2.2.1 La conception

Cette action permet de mettre en lien le modèle de connaissance préalablement établi et le type d'application (résolution approchée, exacte, par simulation), désiré à la fin, conséquence des choix faits en amont par les experts afin d'atteindre les objectifs et résoudre les problèmes voulus.

Des composants logiciels génériques seront donc formalisés et conçus dans le but d'être utilisés par la suite afin de construire un modèle d'action (programme informatique) pour tout système du domaine.

Contrairement à l'implémentation (partie suivante), la conception ne prend pas en compte les spécificités de tel ou tel langage. En effet, il ne s'agit ici que d'un formalisme général, excluant donc l'utilisation d'un langage de simulation ou de programmation en particulier.

4.2.2.2 L'implémentation

Cette dernière action est celle où il sera décidé quel langage ou logiciel sera utilisé pour coder les composants logiciels précédemment établis. De nombreux choix s'offrent aux utilisateurs à ce niveau, étant donné qu'il est possible de s'appuyer soit sur un langage dit généraliste (C, C++, etc.), sur un langage dit dédié de simulation (SIMAN) sur des solveurs spécialisés (MATLAB), ou directement sur des logiciels adéquats (ARENA, SIMIO).

L'objectif visé est donc d'obtenir une bibliothèque de composants (modulables selon le problème rencontré) afin de construire différents modèles d'actions suivant le système étudié.

4.2.2.3 Le modèle d'action

Servant à l'évaluation des performances du système étudié, ou proposé, ce modèle est comme dit ci-dessus une traduction d'un modèle de connaissance d'un système. Suivant les objectifs, suivant le système, mais encore suivant les hypothèses de modélisation, il peut être traduit, en partie ou complètement, dans divers langages ou par le biais de logiciels. Ce modèle d'action est donc directement exploitable et fournit l'évaluation des critères de performances qui constituent le dernier modèle, à savoir le modèle de résultats, et cela sans recourir à la mesure directe.

A noter que plusieurs modèles d'action peuvent être construits à partir d'un unique modèle de connaissance selon le formalisme choisi, le niveau de représentation permettant de disposer d'une panoplie d'outils d'aide à la décision variés et complémentaires, afin d'aborder un même

problème sous des angles différents, avec des méthodes différentes et de travailler si besoin sur des sous-problèmes précis.

4.2.3 Etape 3 : Mise en œuvre du modèle d'action

L'utilisation du modèle d'action permet comme déjà énoncé d'alimenter le modèle de résultat, que nous allons détailler.

Ce dernier modèle est ce que les utilisateurs et les premiers bénéficiaires de la démarche vont observer et utiliser. Bien qu'il présente une partie des résultats obtenus lors de l'exploitation du modèle d'action, ce modèle doit être majoritairement conçu en amont de cette étape. En effet, celui-ci doit être défini au moment du choix des critères de performances, évalués dans le modèle d'action, et cela en collaboration avec les utilisateurs si possible, afin de pouvoir répondre aux questions et objectifs fixés.

Il regroupera et mettra donc en forme les résultats quantitatifs et qualitatifs fournis par le modèle d'action afin de permettre l'analyse du système et la prise de décision.

Le modèle de résultat obtenu et complété par les informations issu du modèle d'action, il est désormais possible pour les utilisateurs de prendre des décisions et par conséquent d'agir sur le système. Ce sera la dernière étape.

4.2.4 Etape 4 : Action sur le système

Lors de l'interprétation des résultats, la pertinence et la validité des différentes solutions envisagées et prospectées seront évaluées, amenant ensuite à de possibles modifications sur le système existant ou à concevoir. Ces modifications peuvent bien entendu être de différentes natures et pourront se répercuter sur les différents éléments de la démarche ASCI. Ainsi les décideurs pourront choisir de modifier la structure même du système (en lien avec la décomposition systémique), leurs processus (définissant le comportement des entités) ainsi que les règles de gestion.

4.2.5 Conclusion

En conclusion, la démarche ASCI a un objectif clair, celui de permettre une analyse d'un système (autant d'un point de vue conception qu'amélioration), qui soit modulaire et évolutive, celle-ci apprenant de ses propres résultats pour s'améliorer si nécessaire. Les cycles de réutilisation des différents éléments conçus lors de l'analyse (modèle de connaissances) et dans la bibliothèque de composants (pour le modèle d'action), font de la réutilisabilité d'ASCI, un de ses atouts.

4.3 BUSINESS PROCESS MODEL AND NOTATION

Présenté succinctement dans la partie 2.3.1, ce langage va être ici présenté plus en détails, celui-ci ayant été sélectionné comme le langage qui sera utilisé pour modéliser les processus de la pharmacie. Le but premier de BPMN est d'apporter une notation facilement compréhensible et

utilisable par tous les acteurs impliqués dans les processus, de l'analyste business qui crée les premières représentations des processus au manager qui va contrôler et évaluer ces processus.

BPMN a de nombreux avantages mais pose quelques problèmes d'interprétation et d'utilisation comme nous le verrons, bien que Jan Recker considère BPMN comme l'une des notations les plus abouties pour représenter les processus.

Dans un premier temps, nous aborderons le BPM d'une manière générale afin de comprendre le but de cette démarche, avant de faire un bref historique du langage qui nous intéresse et de poursuivre sur la présentation de sa symbolique graphique. Pour terminer, nous discuterons de l'utilisation et de l'impact de BPMN d'une manière générale et sur les systèmes hospitaliers.

4.3.1 Le Business Process Management

Les changements, radicaux ou incrémentiels, ayant lieu afin d'améliorer la compréhension des processus métiers sont depuis toujours une partie de l'histoire de l'évolution des organisations. Cependant, la compréhension de ce domaine a réellement pris son envol durant les années 90, celui-ci devenant l'une des priorités des organisations et notamment des managers, de nombreuses recherches étant menées pour comprendre le BPM afin d'être capable de l'utiliser, de l'implémenter [HLUPIC 05]. Permettant de modéliser les bonnes pratiques au sein de l'organisation et d'organiser le recueil de connaissances par le biais des processus, le BPM peut être défini comme étant :

Définition 4.2 : *Le processus d'entreprise utilisant des méthodes, outils, concepts et logiciels pour concevoir, formaliser, contrôler et analyser les processus de l'entreprise.* [WESKE 04]

Le BPM peut donc être résumé comme étant le processus permettant de modéliser les processus d'entreprise. En 2009, le BPM continue d'être une priorité, et pouvoir construire des processus métiers reste un des challenges majeurs pour les cadres supérieurs. Ainsi, le groupe LANNER²² a identifié trois types d'utilisateurs potentiels pour le BPM :

- ❖ Le concepteur : conception et formalisation des processus ;
- ❖ L'exécuteur : déploiement et mise en application des processus;
- ❖ Le manager : contrôle et surveillance des processus.

On retrouve donc une utilité du BPM à tous les niveaux d'un projet visant à identifier et formaliser les processus de son organisation.

4.3.2 BPMN, une histoire récente

« Business Process Model and Notation » est une technique de modélisation des processus proposée récemment par l'« Object Management Group²³ » et qui se base sur la révision de nombreuses notations tel qu'UML, IDEF, ebXML, RosettaNet, LOVeM ou EPC. Le développement de BPMN provient de la demande d'avoir une notation graphique pour compléter le standard BPEL4WS (permettant d'exécuter les processus métiers) et s'appliquant au BPM.

²² <http://www.lanner.com/fr/>

²³ Object Management Group : <http://www.omg.org/>

Proposé dans un premier temps par le groupement BPMI²⁴ (2004 – Version 1.0), ce dernier fusionna avec l'OMG afin de développer de manière plus poussée BPMN et fourni en janvier 2008 une version 1.1, puis 1.2 un an plus tard. La dernière version en cours est la 2.0, fournie et validée par l'OMG en janvier 2011, ce qui souligne d'autant plus la jeunesse de ce langage. La version 1.2 de BPMN connut un très grand succès et il est à noter que la plupart des outils reste encore à ce jour en version 1.2 (bien que la 2.0 soit de plus en plus implémentée).

4.3.3 Les éléments du langage

BPMN possédant divers types de représentation en son sein, nous ne présenterons ici que les éléments du langage utiles pour modéliser les processus métiers dans les **diagrammes de collaboration** et non pas ceux utiles pour les deux autres (à savoir diagramme de conversation et chorégraphie).

L'utilisation de BPMN mène à la création d'un BPD (Business Process Diagram), sorte de logigramme, conçu pour représenter graphiquement les processus métiers.

La spécification de BPMN différencie les éléments composant ce langage en deux groupes : un ensemble d'éléments graphiques de base (corps de base) et un ensemble d'éléments spécialisés (corps étendu). Ces deux ensembles sont composés de 38 éléments de langage pour construire les processus (en plus des attributs) réparti en cinq catégories:

- ❖ Les objets de flux ;
- ❖ Les données ;
- ❖ Les connecteurs ;
- ❖ Les lignes ;
- ❖ Les artefacts.

Chacune de ces catégories est elle-même composée de divers éléments. Ainsi on retrouve pour les objets de flux, permettant de formaliser le comportement d'un processus métier :

- ❖ Evènement ;
- ❖ Activité ;
- ❖ Porte décisionnelle.

Pour les données :

- ❖ Objet de donnée ;
- ❖ Donnée d'entrée ;
- ❖ Donnée de sortie ;
- ❖ Entrepôt de données.

Il y a trois connecteurs possibles pour relier les objets de flux entre eux ou avec d'autres informations :

- ❖ Flux de séquence ;
- ❖ Flux de message ;
- ❖ Associations.

²⁴ Business Process Management Initiative : <http://www.bpmi.org/>

Parmi les « lignes » il est possible de grouper de deux manières différentes les éléments du modèle :

- ❖ Piste (ou « Pool » en anglais) ;
- ❖ Ligne (ou « Lanes » en anglais).

Enfin les artefacts sont utilisés pour apporter des informations supplémentaires sur les processus. Il y a actuellement deux artefacts standardisés, mais il reste possible aux divers utilisateurs ou développeurs de rajouter autant d'artefacts qu'ils le désirent. On retrouve actuellement :

- ❖ Groupe ;
- ❖ Annotation de texte.

Ces divers éléments vont être présentés dans ce qui va suivre en suivant le même ordre que listés dans cette partie.

4.3.4 Les activités

Le premier élément l'« **Activité** », est un terme générique représentant le travail qui est réalisé par une organisation et qui prend place dans un processus métier. Les diverses sous-classes d'**Activité** existantes sont les Sous-processus, les Tâches, les Activités appelantes et les Transactions réutilisables à souhait dans le modèle et pouvant être atomiques ou non. L'**Activité**, symbolisée par un rectangle aux coins arrondis (Figure 4.5), nécessitera normalement un certain temps pour être réalisée, une ou plusieurs ressources de l'organisation pour fonctionner, certaines entrées (physiques, informations, etc.) et produira bien entendu des sorties (physiques, informations, etc.).



Figure 4.5 : Représentation basique d'une Activité

La classe **Activité** possède un certain nombre de paramètres permettant de jouer sur son fonctionnement, son comportement, sa réalisation comme nous allons le voir. Cependant, tous ces paramètres ne s'appliquent pas aux sous-processus et aux transactions.

4.3.4.1 Le comportement de la classe Activité

BPMN donne la possibilité de spécifier par des marqueurs le comportement de l'**Activité** lors de son exécution. Il est donc possible de la boucler (Loop), de l'instancier mais aussi de la compenser. Détaillons rapidement ces comportements :

- ❖ Loop : il est possible de faire boucler une tâche/activité/sous-processus afin de répéter son exécution et cela en mode « while-do » (tant que, faire) ou « do while » (faire, tant que). BPMN permet de paramétrer le nombre maximum de boucles à réaliser, la/les condition(s) de sortie, etc.

- ❖ **Multi-Instance** : pouvant être séquentielle (traits verticaux, Figure 4.7) ou parallèle (Traits horizontaux), ce comportement permet de réaliser plusieurs fois une tâche/activité/sous-processus avec des jeux de données différents. Il est important de noter qu'il n'y pas ici de notion de cycle. Chaque exécution de l'entité est indépendante des autres exécutions. La tâche/activité/sous-processus soumis à ce comportement sera en quelque sorte clonée.
- ❖ **Compensation** : ce comportement permet de revenir sur l'exécution de tâches/activités/sous-processus (réalisés avec succès), parce que les résultats ou les effets de ces entités ne sont plus désirés ou doivent être annulés. Attention, si une tâche/activité/sous-processus est toujours en exécution, aucune compensation ne sera possible. La mise en application de ce comportement est particulière car elle nécessite un évènement spécifique pour être déclenchée et est en quelque sorte coupée du processus normal. L'exemple de la Figure 4.6 montre cela, avec l'évènement compensation (sur « Book Hotel ») et l'action de compensation « Cancel Hotel ».

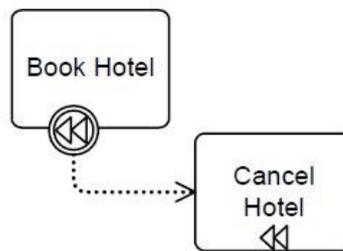


Figure 4.6 : Exemple d'une Activité de compensation liée à son activité de base

Les symboles utilisés pour la Tâche avec ces comportements sont les suivants (Figure 4.7) :



Figure 4.7 : Illustration des marqueurs sur une Tâche pour les trois comportements

4.3.4.2 Les sous-classes d'Activité

La première sous-classe et la plus fondamentale est la Tâche, qui est une **Activité atomique** et qui est utilisée quand le travail réalisé dans un processus ne peut être porté à un niveau de détails plus fin. La symbolique de la Tâche est la même que pour l'**Activité** (Figure 4.5).

La seconde est l'Activité appelante, en quelque sorte entre la Tâche et le Sous-processus (présenté ci-dessous). Cette sous-classe agit donc comme une activité d'emballage par laquelle un processus global ou une tâche globale est exécuté. Sa symbolique est la même que pour l'activité de base (Figure 4.8 (a)) ou le Sous-processus non étendu (Figure 4.8 (b)) mais avec un trait de contour grossi.

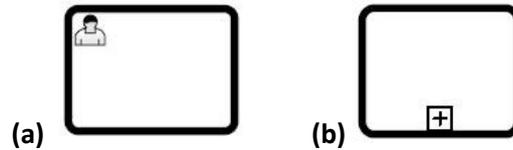


Figure 4.8 : « Activité appellante » pour une Tâche (a) ou un processus (b)

Le troisième, et non moins important, est le Sous-processus qui peut être vu comme un agrégat de divers objets de flux (activité, porte, évènement, flux de séquence). C’est donc une vue macroscopique d’un ensemble d’activités. Le Sous-processus est un objet graphique intégré à un processus mais pouvant si besoin être étendu pour en voir le contenu. Les détails d’un sous-processus ne sont pas visibles d’emblée, mais le seront par l’utilisation du symbole « + » encadré (Figure 4.9 (a)), indiquant que ce sous-processus possède une décomposition. A noter qu’aucun flux ne peut couper les bords du rectangle symbolisant le processus. Seuls des évènements peuvent se positionner dessus (Figure 4.10).

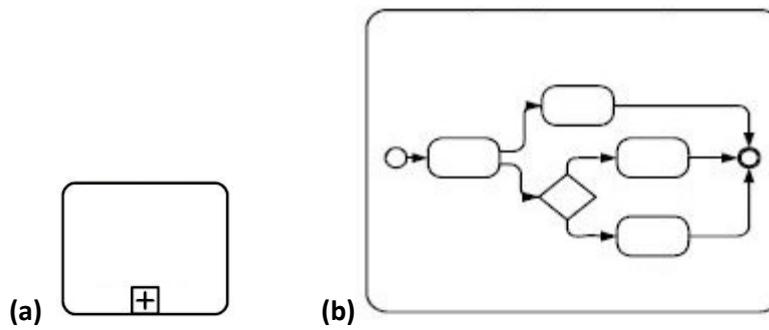


Figure 4.9 : Sous-processus réduit (a) ou étendu (b)

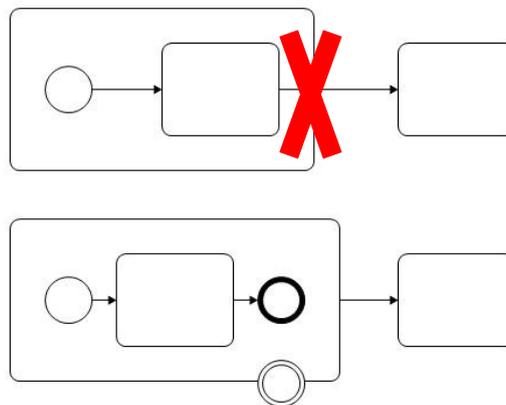


Figure 4.10 : Sous processus non valide (en haut) et valide (en bas)

En plus des comportements définis précédemment (Partie 4.3.4.1), on trouve pour les sous-processus un type supplémentaire de comportement à savoir « Adhoc ». Dans ce type de sous-processus, un ensemble de tâches/activités peut être défini, mais l’enchaînement, la séquence et le nombre d’exécutions de chaque entité sont déterminés par la ressource exécutant la tâche/activité. De plus ce comportement peut être couplé à un autre comme la Compensation ou la Multi-Instance (Figure 4.11).

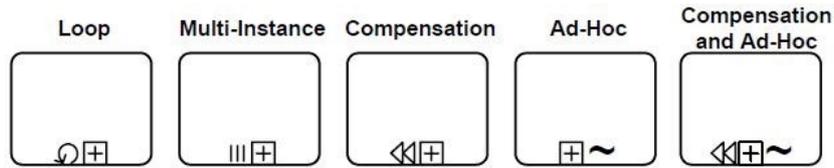


Figure 4.11 : Ensemble des comportements pour un Sous-processus

La dernière sous-classe est la Transaction (Figure 4.12), **Activité** particulière, celle-ci étant un type spécial de Sous-Processus. Le fonctionnement de celui-ci est ici contrôlé par le biais d'un protocole de transaction, qui s'assure que toutes les parties prenantes aient complété leur rôle et aient atteint un point de concordance (prédéfini). Dans le cas contraire, la transaction est annulée et chaque partie doit défaire ce qu'elle vient de réaliser. La symbolique est celle du sous-processus avec un contour en double ligne.



Figure 4.12 : Symbole utilisé pour la Transaction

A noter le caractère particulier d'un sous-processus à savoir le sous-processus évènementiel. Celui-ci se différencie par son symbole (le contour est fait de points et non d'un trait plein, Figure 4.13 (a)) et par le fait qu'il n'a ni de flux d'entrée, ni de sortie. Il se positionne à l'intérieur d'un autre sous-processus (Figure 4.13 (b)) et traduit des fonctionnalités ou comportements particuliers de celui-ci.

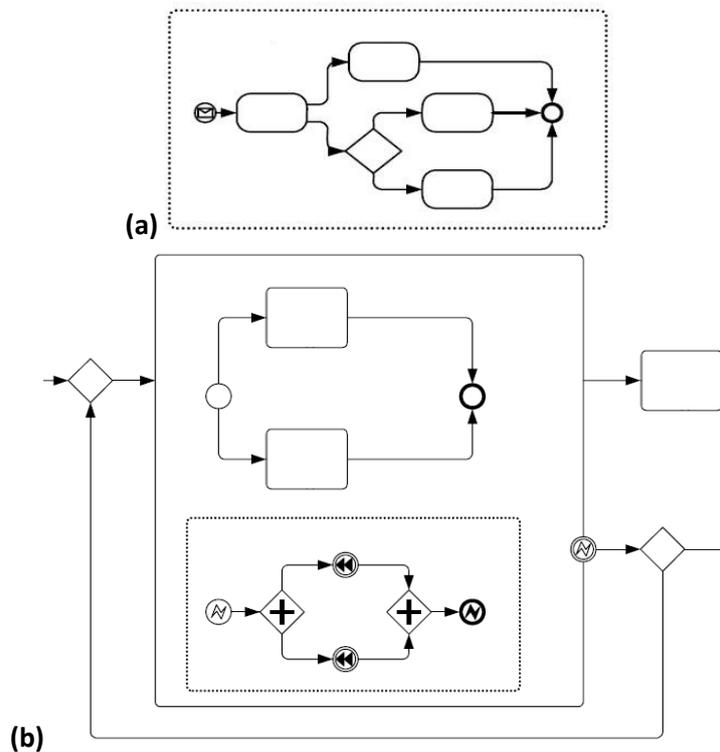


Figure 4.13 : Sous-processus évènementiel seul (a) et incorporé dans un autre sous-processus (b)

4.3.4.3 Le type d'Activité

S'appliquant uniquement à la Tâche et à l'Activité appelante, il est possible, comme pour le comportement, de définir des types d'**Activité**. Etant au nombre de sept, nous n'en présenterons ici que cinq, les deux derniers (Business et Script) étant utiles dans le cas de l'utilisation d'un moteur de simulation des processus métier, moteurs que nous n'utiliserons pas, ces derniers atteignant rapidement leurs limites. Les tâches/activités sont donc de type :

- ❖ Abstraite : il s'agit de la tâche/activité de base où aucun type n'est spécifié. Le symbole utilisé est celui de la tâche/activité de base (Figure 4.5) ;
- ❖ Service : ce type de tâche est une tâche où il sera utilisé un service qui peut aussi bien être un service web qu'une application automatisée. La symbolique est la suivante :

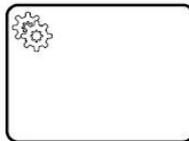


Figure 4.14 : Illustration d'une Tâche de type « Service »

- ❖ Envoi : comme son nom l'indique, il s'agit d'une simple tâche/activité désignée pour envoyer un message à un participant externe (par rapport au processus en cours). Une fois cela réalisé la tâche est complétée ;



Figure 4.15 : Illustration d'une Tâche de type « Envoi »

- ❖ Réception : pendant de la précédente, la tâche/activité réceptionne ici un message provenant d'un participant externe. Une fois la réception finie, la tâche est terminée ;



Figure 4.16 : Illustration d'une Tâche de type « Réception »

- ❖ Utilisateur : il s'agit de la tâche typique qui prend place dans un processus de type « workflow ». Celle-ci représente l'exécution de la tâche/activité par une personne avec l'assistance d'un logiciel ;

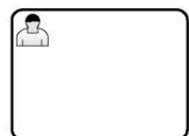


Figure 4.17 : Illustration d'une Tâche de type « Utilisateur »

- ❖ Manuelle : ce type de tâche/activité doit être réalisé sans l'aide d'aucun support business ou informatique.



Figure 4.18 : Illustration d'une Tâche de type « Manuelle »

4.3.5 Les évènements

Un **Evènement** est quelque chose qui se produit durant la réalisation du processus. Cet **Evènement** affecte le flux du processus et a usuellement une cause et un impact. Ce terme couvre un large spectre de situations (début, fin du processus, changement d'état, message, etc.). La symbolique d'un **Evènement** est un cercle, qui selon le cas aura un contour simple ou double et pourra contenir un symbole. Ces différents cas seront détaillés ci-dessous. Il sera trouvé néanmoins trois types d'évènements, dépendants de quand ils affectent le processus, à savoir l'Evènement de Début, Evènement de Fin et l'Evènement Intermédiaire. Le rôle des deux premiers est évident, et le troisième quant à lui intervient entre ces deux-là affectant le déroulement du processus.

Etant donné le grand nombre d'évènements différents, nous ne les présenterons pas tous ici. Seuls les plus communs le seront (la liste de tous les évènements est en **Annexe 4**).

4.3.5.1 L'Evènement de Début

Comme son nom le suggère, il indique où le processus commence. Cet évènement ne possède pas de flux entrants mais peut posséder plusieurs flux sortants. Sa symbolique est un simple cercle vide visible sur la Figure 4.19. Un Evènement de Début n'est pas obligatoire mais est fortement recommandé dans le cas de processus complexe, et il doit être impérativement présent si un Evènement de Fin est présent (et réciproquement).

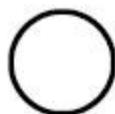


Figure 4.19 : Symbole d'un Evènement de Début

Pour ce type d'**Evènement**, quatre sont à mettre en avant. Ceux de type « Message », « Timer », « Conditionnel » et « Signal ». En plus bien entendu du type « Aucun » qui est l'évènement généraliste (Figure 4.19)

Le premier « Message », signifie qu'un message arrive d'un participant et que celui-ci déclenche le processus associé à cet évènement. Sa symbolique est celle de base dans laquelle est incorporée une enveloppe (Figure 4.20 (a)).

Le second « Timer », est bien entendu lié au temps. Pouvant représenter une date spécifique (le jeudi à 8h), ou un cycle (toutes les 12 heures), il est symbolisé par une horloge dans un cercle (Figure 4.20 (b)).

Le suivant est le « Conditionnel ». Comme son nom l'indique, ce type d'évènement se déclenchera sous des conditions prédéfinies (par exemple : nombre de produits en stock inférieur au

seuil de sécurité). Ces conditions doivent référer à des attributs (statiques) du processus ou l'état des entités du système. Son symbole est une feuille dans le cercle de base (Figure 4.20 (c)).

Le dernier présenté ici est l'Évènement de Début de type « Signal ». Permettant de déclencher un processus par réception d'un signal provenant d'un autre processus, il est à noter que cet évènement ne peut être considéré comme un « Message ». En effet, plusieurs processus peuvent être enclenchés par un même signal alors qu'un message ne transite qu'entre deux entités (et donc deux processus). Son symbole est le dernier de la Figure 4.20, à savoir un triangle dans un cercle.

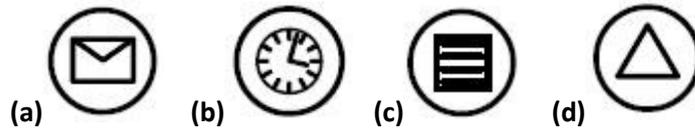


Figure 4.20 : Évènement de Début : (a) Message, (b) Timer, (c) Conditionnel, (d) Signal

A noter que ces évènements peuvent intervenir dans les Sous-processus, et agissent de la même manière. La seule différence est que cette fois-ci, ils peuvent interrompre ou non le déroulement du sous-processus. Prenons l'exemple du « Message ». Dans le cas où le cercle serait continu, le processus associé pourrait interrompre le déroulement du sous-processus dans lequel il se trouve dès qu'une réception de message aurait lieu. Si le cercle est en pointillés alors le processus associé au « Message » ne pourra se déclencher qu'à la fin du sous-processus et donc ne peut interrompre son déroulement. En Figure 4.21, la représentation des deux cas.

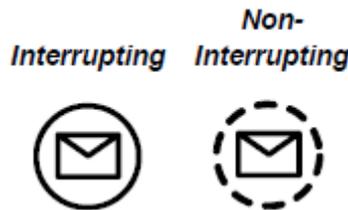


Figure 4.21 : Évènement de début Interruptible et Non-Interruptible

4.3.5.2 L'Évènement Intermédiaire

Ce deuxième type d'Évènement prend place comme nous l'avons dit précédemment dans le processus et agit sur le déroulement de celui-ci. Sa symbolique suit la même logique que celle de tous les évènements, à savoir un cercle vide, excepté que cette fois le contour de celui-ci est fait de deux lignes comme représenté sur la Figure 4.22.



Figure 4.22 : Évènement Intermédiaire de base

Un Évènement Intermédiaire peut être placé n'importe où sur un processus ou sur une **Activité** de type Tâche ou Sous-processus. Pour le second cas il sera néanmoins placé sur la bordure de l'**Activité**, comme il est possible de le voir pour la compensation (Figure 4.6).

Tout comme pour l'Évènement de Début, il y a 12 types possibles pour cet **Évènement**. Ces différents types peuvent être utilisés de deux manières différentes : comme déclencheur (Emission) ou comme récepteur (Réception). La logique de fonctionnement entre les deux catégories est donc

différente. Pour le premier (Emission), dès que le processus arrivera sur cet évènement, celui-ci se déclenchera (par exemple, l'envoi d'un message), tandis que pour le second (Réception) le processus devra attendre que l'évènement déclencheur (un autre évènement, une activité, une heure précise suivant le type de l'Évènement Intermédiaire) ait lieu. A noter que des évènements placés en bordure d'une Tâche ou d'un Sous-processus (Figure 4.6 et 4.10) ne peuvent être que de type « Réception » et permettent d'exprimer un comportement ou une réaction exceptionnelle de la Tâche ou du Sous-processus auquel ils sont attachés.

Parmi les différents types d'Évènement Intermédiaire, tous ne peuvent être utilisés en « Réception » ou en « Emission ». La Figure A.9 en **Annexe 4** permet de visualiser cela. Nous n'allons présenter ci-dessous que les types « Lien » et « Multiple Parallèle ».

Le premier est uniquement utilisable sur le processus (et par conséquent ne peut être placé en bordure d'une **Activité**), et représente un lien entre deux portions du processus. Il peut donc être utilisé pour créer des boucles dans le processus ou pour éviter de longues séquences du dit processus. Parmi les autres applications possibles du « Lien », il est possible de citer la connexion de plusieurs pages représentant un unique processus afin de l'imprimer ou l'utilisation comme objet générique « Go-to ». La symbolique de cet **Évènement** est le cercle dans lequel est placée une flèche horizontale pleine ou vide suivant qu'elle soit d'« Emission » ou de « Réception » (Figure 4.23 (a)).

Le second type d'Évènement Intermédiaire ici présenté, est utilisable uniquement en « Réception », ce qui signifie que pour être déclenché, validé, tous les évènements entrants qui lui sont liés doivent avoir eu lieu (contrairement à l'évènement Multiple que nous détaillerons dans les Évènement de Fin). Sa symbolique est le signe « Plus » dans le cercle d'**Évènement** (Figure 4.23 (b)).



Figure 4.23 : Évènement Intermédiaire de type « Lien » (a) ou « Multiple Parallèle » (b)

A noter que tout comme l'Évènement de Début, l'Évènement Intermédiaire peut être utilisé dans les sous-processus pour interrompre ou non son déroulement. De nouveau ce type d'évènement sera interrupteur si les cercles sont pleins, sinon les cercles seront en pointillés (Figure 4.24).

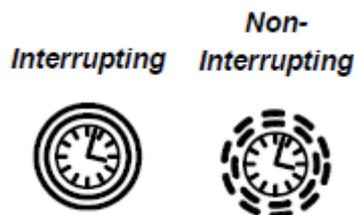


Figure 4.24 : Évènement Intermédiaire de type « Timer » Interruptible et Non-Interruptible

4.3.5.3 L'Évènement de Fin

La dernière catégorie d'**Évènement** est bien entendu l'Évènement de Fin. Concluant les processus, celui-ci ne peut avoir que des flux entrants (uniques ou multiples). Sa symbolique est de nouveau similaire aux autres, sauf que cette fois-ci le contour du cercle est épais. Tout comme l'Évènement de Début, celui-ci n'est pas obligatoire dans le processus, mais doit être présent si un Évènement de début l'est.



Figure 4.25 : Symbole de base d'un Evènement de Fin

Il existe neuf types d'Evènement de Fin. Quatre d'entre eux seront présentés : Erreur, Multiple, Arrêt et Aucun.

L'Evènement de Fin de type « Aucun » est celui de base et permet de représenter toutes les situations de manière générique. Sa symbolique est celle de la Figure 4.25 (ci-dessus).

Le second, « Erreur » est comme sous-entendu la conséquence d'une erreur ayant eu lieu et il sera à l'origine de l'émission d'un signal d'erreur. Cela signifie qu'une fois tout le processus terminé, une erreur sera signalée. Ce type de fin est souvent retrouvé dans les sous-processus en cas de dysfonctionnement lors de l'exécution de celui-ci. Sa symbolique est un « éclair » dans le cercle de fin (Figure 4.26 (a)).

Le « Multiple » donne la possibilité de déclencher plusieurs évènements en une fois. Par exemple pour l'envoi de messages multiples (un Evènement de Fin de type « Message » ne peut être utilisé car celui-ci n'enverrait qu'un seul message), ou pour l'envoi d'un message, d'un signal et d'une erreur en même temps. Il permet donc de composer une fin de processus avec les éléments voulus. Sa symbolique est un pentagone dans le cercle de fin (Figure 4.26 (b)).

Le dernier, l'« Arrêt », signifie qu'à l'exécution de celui-ci toutes les activités du processus doivent cesser immédiatement. Cela inclu bien entendu toutes les instances pouvant exister dans le processus. Sa symbolique est un cercle plein à l'intérieur du cercle de fin (Figure 4.26 (c)).

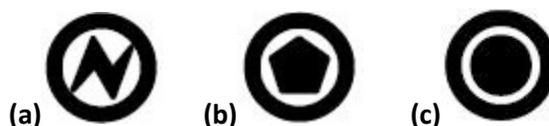


Figure 4.26 : Evènement de fin de type « Erreur » (a), « Multiple » (b) et « Arrêt » (c)

4.3.6 Les Portes

La **Porte** est utilisée pour contrôler comment les flux de séquence se comportent à l'intérieur du processus. Si le flux ne nécessite aucun contrôle, alors aucune **Porte** ne sera nécessaire. Le terme **Porte** sous-entend qu'il y a des mécanismes pour autoriser le passage du flux de séquence à travers celle-ci comme nous le verrons dans ce qui suit. La symbolique de la Porte est un losange vide (Figure 4.27).

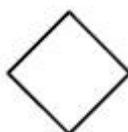


Figure 4.27 : Symbole de base d'une Porte décisionnelle

Bien que contrôlant le processus, la **Porte** ne représente pas, contrairement à l'**Activité**, un travail, une tâche, ce qui signifie qu'elle n'a aucun impact physique sur le processus (en termes de temps, de coût, etc.).

La **Porte** peut définir tous les types de comportements permettant d'agir sur les processus métiers à savoir l'aspect décisionnel (inclusif, exclusif et complexe), la fusion, la scission d'un processus, etc. Ainsi pour préciser le type de **Porte**, il sera rajouté à l'intérieur du losange de base un icône permettant d'en signifier le fonctionnement. Les icônes pouvant être trouvées sont :

- ❖ Un « X » ou aucun symbole pour les portes de type exclusives ;
- ❖ Un « + » pour les portes de type « parallèle » ou le symbole d'évènement « Multiple parallèle » pour les portes « parallèle » basées sur un évènement (de début uniquement) ;
- ❖ Un cercle pour les portes de type inclusives ;
- ❖ Un astérisque pour les portes complexes ;
- ❖ Le symbole de l'évènement « Multiple » pour les portes basées sur ce type d'évènement (de début ou intermédiaire cette fois).

Détaillons chacun de ces types. Pour chaque cas la réflexion sera portée à la fois sur le côté convergent et divergent d'une porte.

4.3.6.1 La Porte Exclusive

Une porte divergente de type Exclusive est utilisée pour créer des chemins divergents à l'intérieur d'un processus mais où un seul et unique choix peut être fait parmi eux pour poursuivre le processus. Ce type de **Porte** se caractérise souvent par la présence d'une question permettant de faire un choix parmi les sorties possibles, chacune de ces sorties possédant une réponse différente. Les deux représentations de cette porte sont les suivantes :

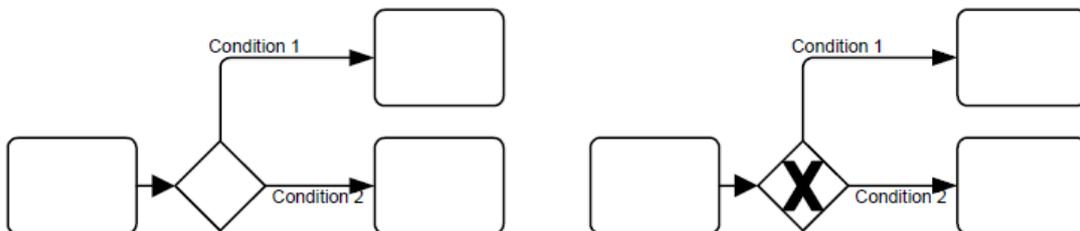


Figure 4.28 : Représentation d'une porte Exclusive divergente avec les deux symboliques

Une porte convergente de ce type sera passante dès lors que l'un de ses chemins entrants sera terminé, c'est-à-dire pour la Figure 4.29, qu'il suffit que l'une des deux situations soit complétée.

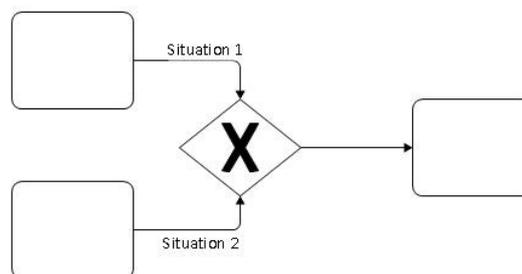


Figure 4.29 : Représentation d'une porte exclusive convergente avec la marque

Exclusive = Un seul chemin possible (divergente) ou nécessaire (convergente) à la fois

4.3.6.2 La Porte Inclusive

Une porte divergente de type Inclusive (Figure 4.30) permet de créer des chemins alternatifs mais aussi parallèles dans un processus. Contrairement à la porte précédente il est possible d'avoir plusieurs chemins en sortie si la condition de la porte est vérifiée pour chacun d'entre eux. Il est donc possible de n'avoir aucun chemin de sortie après cette porte à moins qu'il soit précisé un chemin « par défaut », chose que nous détaillerons dans la partie sur les flux.

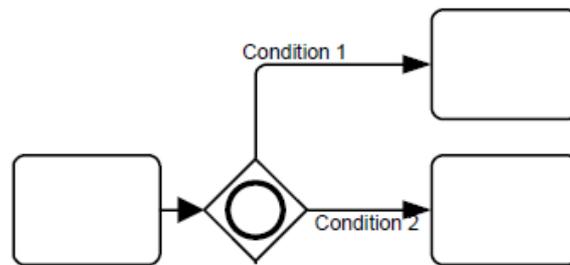


Figure 4.30 : Représentation d'une porte Inclusive divergente

Une porte Inclusive convergente est donc utilisée pour fusionner plusieurs chemins alternatifs ou parallèles d'un processus. Il peut donc être nécessaire d'attendre plusieurs chemins si ceux-ci sont synchronisés pour pouvoir passer la porte.

Inclusive = Un ou plusieurs chemins possible(s) en même temps

4.3.6.3 La Porte Parallèle

Une porte Parallèle (Figure 4.31) sert à synchroniser des chemins entre eux. Que ce soit en tant que porte convergente, permettant de combiner plusieurs chemins, qu'en tant que porte divergente, permettant de créer des chemins, tous ces chemins entrants ou sortants doivent être ou seront exécutés. Contrairement aux deux portes précédentes, ici, aucune condition de passage n'a besoin d'être vérifiée, excepté pour la porte convergente où tous les chemins doivent avoir été exécutés pour que la porte s'active.

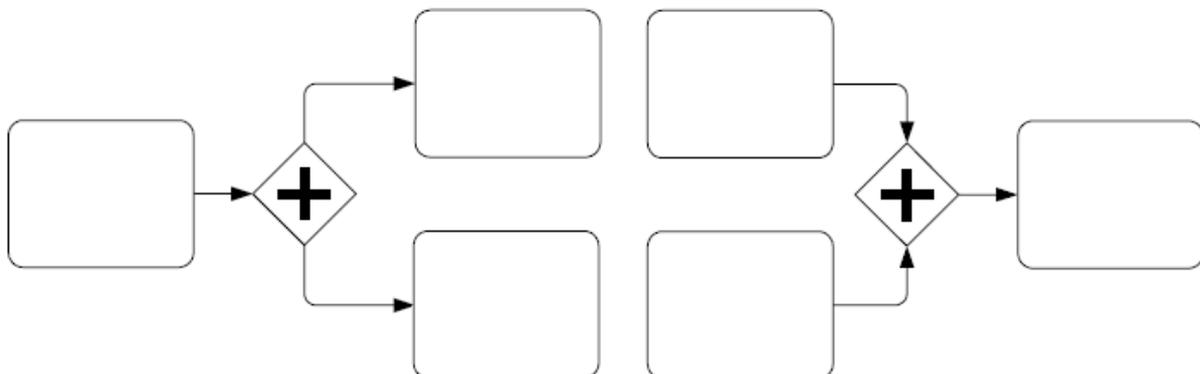


Figure 4.31 : Représentation d'une porte Parallèle divergente (à gauche) et convergente (à droite)

4.3.6.4 La Porte Complexe

Très spécifique, cette porte (Figure 4.32) s'utilise pour modéliser des comportements de synchronisation complexes. La condition de passage de la **Porte** doit donc être spécifiée avec précision et de manière claire pour la meilleure exécution possible de cette porte. Dans le cas d'une porte convergente (Figure 4.32 (a)), cela peut être par exemple l'exécution de 3 tâches sur 4 possibles (soit 3 chemins sur 4 possibles) permettant alors de franchir la porte et de passer à la suite. En ce qui concerne les chemins possibles de sorties (pour une divergente), chacun d'entre eux sera évalué individuellement (comme pour la porte Inclusive en quelque sorte) pour savoir s'ils doivent être enclenchés.

Il est donc possible de faire de nombreuses choses avec ce type de porte, mais comme déjà dit il est important d'être très rigoureux pour que son utilisation et son exécution ne pose pas de problème.

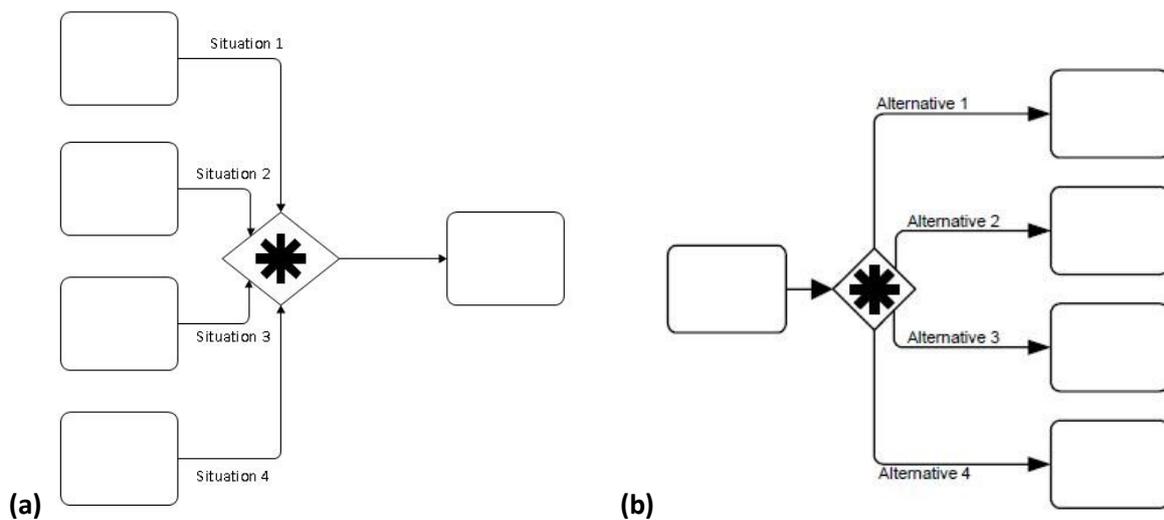


Figure 4.32 : Représentation d'une porte Complexe convergente (a) et divergente (b)

4.3.6.5 Les Portes basées sur un évènement

Celles-ci ont la particularité d'avoir des conditions de sortie qui ne sont plus des expressions mais des évènements comme on peut le voir sur la Figure 4.33. Ce sera donc le premier évènement qui aura lieu qui décidera quel sera le chemin emprunté pour poursuivre le processus. Etant exclusivement des portes divergentes, chacune des sorties possibles doit donc être composée en premier d'un Evènement Intermédiaire, ou bien d'une Tâche de « Réception de message ». Parmi les évènements intermédiaires, ne sont possibles après cette porte que ceux de types « Message », « Signal », « Timer », « Conditionnel » et « Multiple ».

Ce type de porte peut être de « Début » ou « Intermédiaire » mais seule la porte basée sur un évènement « Multiple » pourra être du dernier type. Seule la porte « Intermédiaire » basée sur un évènement « Multiple » sera décrite dessous.

Cette porte requiert au minimum deux sorties, et ces sorties ne doivent pas avoir de condition d'exécution (contrairement aux autres portes). De plus, une fois une sortie activée les autres deviennent obsolètes. Le comportement est alors identique à une porte divergente Exclusive.

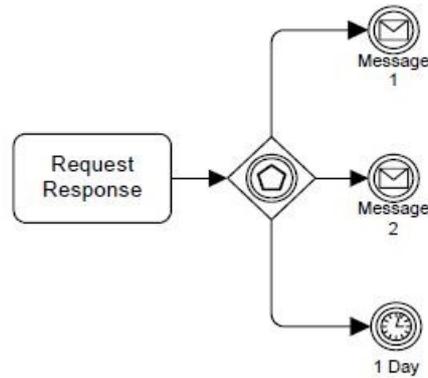


Figure 4.33 : Représentation d'une porte divergente basée sur un Evènement Intermédiaire « Multiple »

A noter qu'une **Porte** peut avoir plusieurs flux entrants et sortants. Cependant il est désormais usuel de découper cela en une porte convergente et une porte divergente afin de gagner en lisibilité du processus.

4.3.7 « Pool » et « Ligne »

Afin de visualiser plus aisément les parties prenantes dans les processus, BPMN utilise les éléments « Pool » et « Ligne » afin de les représenter. Voyons cela.

4.3.7.1 Le « Pool »

Le « Pool » est la représentation graphique d'un Participant dans un diagramme de collaboration. Un Participant peut aussi bien représenter une personne morale qu'une personne physique. Un « Pool » ne contient pas nécessairement un processus et peut donc être utilisé comme une « boîte noire » afin de représenter une entité extérieure à notre processus dont le fonctionnement ne nous intéresse guère.

Sans restriction de taille, le « Pool » est représenté par un rectangle de contour plein sur lequel apparaît le nom dans une case individuelle (Figure 4.34).



Figure 4.34 : Illustration d'un « Pool »

Le processus présent dans un « Pool » ne peut en franchir les limites. Un processus est entièrement contenu dans un seul et unique « Pool ». Seuls les flux de messages peuvent transiter entre différents « Pool ».

4.3.7.2 La « Ligne »

La ligne est une sous-partition d'un « Pool » et couvre toute la longueur ou la hauteur de celui-ci suivant sa forme. Ces « Lignes » sont souvent utilisées pour organiser les activités ayant lieu

dans un « Pool » et qui généralement sont exécutées par différents participants. Par exemple, les différents acteurs dans un service de médecine : Médecin, Infirmière, Interne en médecine, etc.

De plus, la « Ligne » peut être de nouveau scindée pour apporter de nouvelles précisions sur le processus en son sein (par exemple pour le médecin, avant et après visite du patient).



Figure 4.35 : Illustration d'un « Pool » contenant deux « Lignes »

4.3.8 Les données

Il est usuel de retrouver dans la modélisation des processus les différents objets qui sont modélisés et manipulés par le processus. BPMN ne fournit pas une structure pour les nombreuses données pouvant être ainsi manipulées mais fournit néanmoins quelques éléments permettant de travailler avec une structure de données de base. Nous allons présenter pour cela les six items suivants : Objet de donnée, Entrepôt de données, Donnée d'entrée, Donnée de sortie, Message et Associations.

4.3.8.1 L' « Objet de donnée »

Ce premier élément est la base de la structure de données de BPMN, et représente une structure d'informations pouvant être utilisée, modifiée par de nombreuses activités présentes dans le processus. Possédant un nom, des états, il peut provenir de l'extérieur du processus, tout comme n'« exister » que dans celui-ci. L' « Objet de donnée » est symbolisé par une feuille de papier (Figure 4.36 (a)) et peut représenter un seul objet ou une collection (par exemple une liste de commandes), son symbole devenant une feuille avec trois petites lignes verticales parallèles (Figure 4.36 (b)).

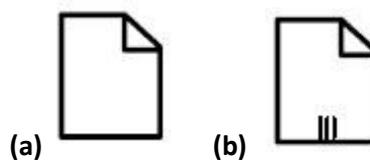


Figure 4.36 : Représentation d'un objet de donnée (a) et d'une collection de données (b)

4.3.8.2 L' « Entrepôt de donnée »

Comme son nom l'indique, cet objet permettra de stocker les différentes informations utiles pour le processus et qui peuvent transiter entre différents processus. Il est représenté par un symbole de base de données commun comme ci-dessous :



Figure 4.37 : Représentation d'un Entrepôt de données

Pour y faire parvenir des informations, en prélever ou mettre à jour certaines d’entre elles, il est utilisé une « Association » qui sera présentée plus bas. Une « Association » dirigée vers l’entrepôt signifiera que l’on y déposera des informations tandis qu’une « Association » qui en sort symbolisera le prélèvement de données.

4.3.8.3 « Donnée d’entrée » et « Donnée de sortie »

« Objet de donnée » particulier, ces deux éléments représentent pour le premier des données provenant de l’extérieur et nécessaires au bon déroulement du processus (pour une Tâche, un Sous-processus), tandis que le second représente des données qui seront fournies par celui-ci à son environnement extérieur. La symbolique est proche de celle de l’ « Objet de donnée », celle-ci étant simplement complétée d’une flèche horizontale vide pour « Donnée d’entrée » (Figure 4.38 (a)) et pleine pour « Donnée de sortie » (Figure 4.38 (b)).

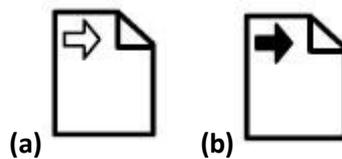


Figure 4.38 : Représentation d’une Donnée d’entrée (a) et d’un Donnée de sortie (b)

4.3.8.4 Association

L’ « Association » est utilisée pour lier les informations et les différents éléments de données avec les éléments graphiques de BPMN. Elle est aussi sollicitée pour lier les annotations de textes ou divers artefacts aux processus, ainsi que pour préciser l’**Activité** utilisée dans le cadre d’une compensation (Figure 4.6). L’ « Association » est une flèche en pointillé, dont la tête permet de signifier la direction d’un flux (de données) entre deux entités (Figure 4.39 (a)) ou une simple ligne en pointillé permettant de préciser l’objet lié au flux (Figure 4.39 (b)).

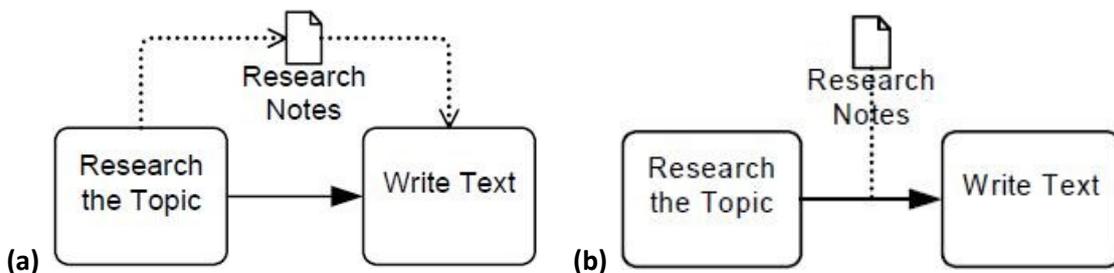


Figure 4.39 : Représentation d’une Association suivant la situation

4.3.8.5 Le « Message »

Cet item représente le contenu d’une communication entre deux participants. Graphiquement représenté par une enveloppe, celui-ci est toujours placé sur un flux de message qui sera détaillé dans la partie suivante.

4.3.9 Les connecteurs

Afin de relier les différents éléments graphiques de BPMN, il existe bien entendu de nombreux types de connecteurs. Ces derniers permettent donc de symboliser l'évolution des différents types de flux au sein et entre les processus. Détaillons-les.

4.3.9.1 Les flux de séquences

Ce premier type de flux permet de symboliser l'évolution du processus à travers les différentes activités qui le composent. Chacun de ces flux ne possède qu'une seule source et qu'une seule cible, et ces deux éléments doivent impérativement être de type **Evènement**, **Activité**, ou **Porte**. Le flux de séquence de base est une simple flèche (Figure 4.40 (a)).

Cette figure montre par la même occasion les deux autres catégories de flux de séquence existantes : conditionnelle (Figure 4.40 (b)) et par défaut.

Alors que la première est représentée par une flèche commencée par un losange, la seconde est complétée à son début par un trait oblique (Figure 4.40 (c)). En ce qui concerne le flux de séquence conditionnel, celui-ci est généralement précédé d'une **Porte** ou d'une **Activité**. Il ne peut en aucun cas l'être par un **Evènement** ou une porte de type « Parallèle ». Comme son nom le sous-entend, cette catégorie de flux peut avoir une condition qui permettra de déterminer si oui ou non le flux sera parcouru/utilisé.

Le flux de séquence par défaut intervient quant à lui dans un cas bien précis, à savoir après une porte « Inclusive » ou basée sur un évènement « Multiple ». Ce flux sera alors utilisé uniquement si les autres flux sortant de la porte ne sont pas choisis.



Figure 4.40 : Flux de séquence standard (a), conditionnel (b) ou par défaut (c)

4.3.9.2 Le Flux de message

Le second type de flux le plus utilisé après le flux de séquence est le flux de message permettant de représenter la transmission d'un « Message » entre deux participants de deux processus distincts (c'est-à-dire de deux « Pool » différents). Ce flux est représenté par une flèche en pointillés dont l'origine est un petit cercle prenant place sur la bordure de l'élément d'origine (un « Pool » ou une **Activité**) et dont la tête est vide (Figure 4.41).



Figure 4.41 : Représentation d'un Flux de message

Il existe encore au sein de BPMN de nombreux éléments graphiques permettant de faire de nombreuses choses. On peut citer le « Groupe », les annotations de textes et bien entendu les nombreuses variantes des évènements que nous n'avons pas présentés ici mais qui sont disponibles en **Annexe 4**. Cependant nous nous arrêterons là, ce langage étant facile à prendre en main mais complexe quand on cherche à en tirer toutes les possibilités et subtilités, comme nous allons le voir dans la partie suivante.

4.3.10 BPMN : son utilisation

Pour terminer, nous allons nous concentrer sur l'utilisation de BPMN et le ressenti que les utilisateurs peuvent avoir sur lui. L'objectif est de montrer que non seulement BPMN est utilisé de manières diverses et variées et cela dans de nombreux domaines, mais aussi que celui-ci pose quelques problèmes quant à son utilisation.

4.3.10.1 Comment BPMN est-il utilisé?

L'étude de [RECKER 08] se base sur le retour d'expérience de 590 utilisateurs de BPMN. Cet auteur constate que l'on a étudié avec précision les tenants et les aboutissants de BPMN, mais qu'un élément n'a pas été encore scruté avec attention, l'utilisateur de BPMN lui-même. Le but de cette étude est donc d'évaluer comment BPMN est utilisé. Un fait marquant concernant cette étude est qu'elle a été soutenue et valorisée par la communauté du BPM, accréditant et donnant plus de poids à ses résultats.

Les 590 réponses obtenues pour cette étude proviennent du monde entier (30 pays différents en tout) et notamment d'Europe, d'Amérique du Nord et d'Océanie. La répartition des réponses est globalement uniforme en termes de provenance (60% proviennent du privé), de filière (40% dans des entreprises de plus de 1000 employés), d'utilité (49% utilisent BPMN pour des problématiques techniques et non business), et d'utilisation (36% utilisent uniquement les corps de base, 37% le corps étendu et enfin 27% toutes les fonctionnalités de celui-ci).

Le premier constat majeur fait par [RECKER 08] est qu'une majorité des utilisateurs ayant répondu (70%) se sont formés d'eux-mêmes ou à travers leur emploi sur BPMN, ce qui l'amène à dire que l'engouement pour ce langage est encore limité (notamment au niveau universitaire). Il préconise donc du fait de la complexité de BPMN de renvoyer les utilisateurs « à l'école » afin de former ceux-ci sur ce langage.

Le second constat est que Visio est l'outil le plus utilisé pour dessiner, construire ses processus BPMN (18,2% des utilisateurs s'en servent). Cependant cela montre aussi que les utilisateurs ne plébiscitent pas un outil en particulier et que parmi ces utilisateurs une grande partie n'utilisent donc BPMN que d'une manière graphique (VISIO ne pouvant exploiter le processus BPMN conçu).

[RECKER 08] a ainsi fait estimer la fréquence d'utilisation des différents éléments graphiques de BPMN, chose visible sur la Figure 4.42. Cette figure montre que presque 50% des éléments de BPMN ne sont pas utilisés. Seules les fonctionnalités basiques de ce langage sont donc exploitées pleinement, ce qui souligne bien le problème majeur de BPMN, à savoir la grande diversité de ces éléments graphiques et la relative complexité de certains d'entre eux.

[ZUR MUEHLEN 08] pousse même la décomposition plus loin, ce qui permet d'affiner encore le constat sur l'utilisation des éléments de BPMN (Figure 4.43). Les résultats proviennent du même set de données que [RECKER 08], ceux-ci ayant juste été analysés avec plus de finesse.

A travers ce nouveau graphique, le constat est encore plus accablant, les éléments de BPMN étant ici mieux explicités. Seuls cinq éléments du langage sont utilisés par plus de 50% des utilisateurs, à savoir le flux de séquence, la tâche, l'évènement de fin, de début et intermédiaire et enfin le « Pool ». Même la porte décisionnelle de base ne rentre pas dans ce pourcentage. Le corps de base du langage semble donc bien maîtrisé par la communauté, contrairement à ceux du corps étendu dont le degré d'utilisation diminue rapidement au fur et à mesure que l'élément graphique étudié se spécialise.

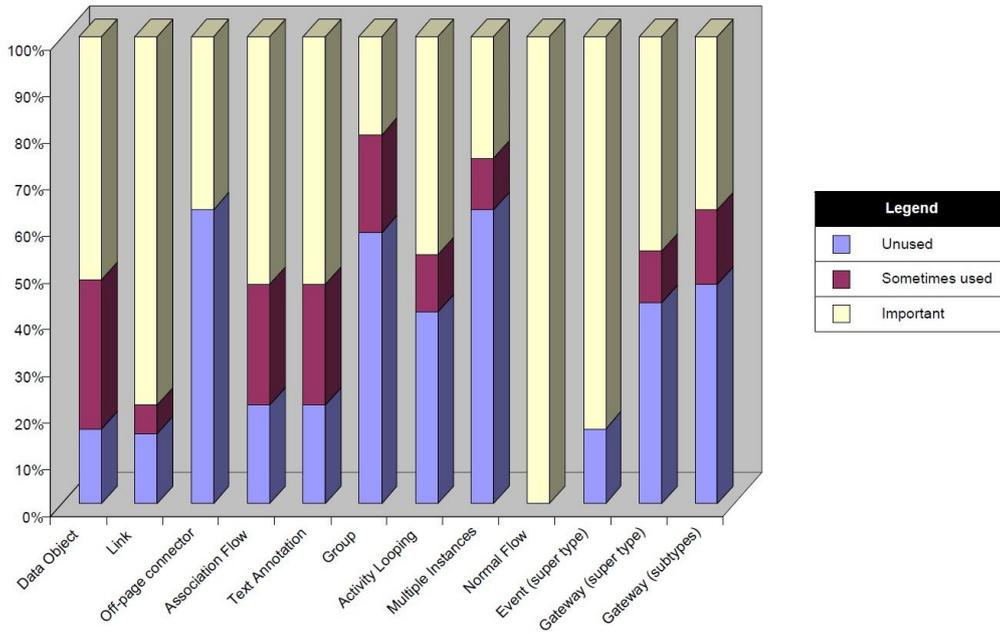


Figure 4.42 : Degré d'utilisation des éléments de BPMN

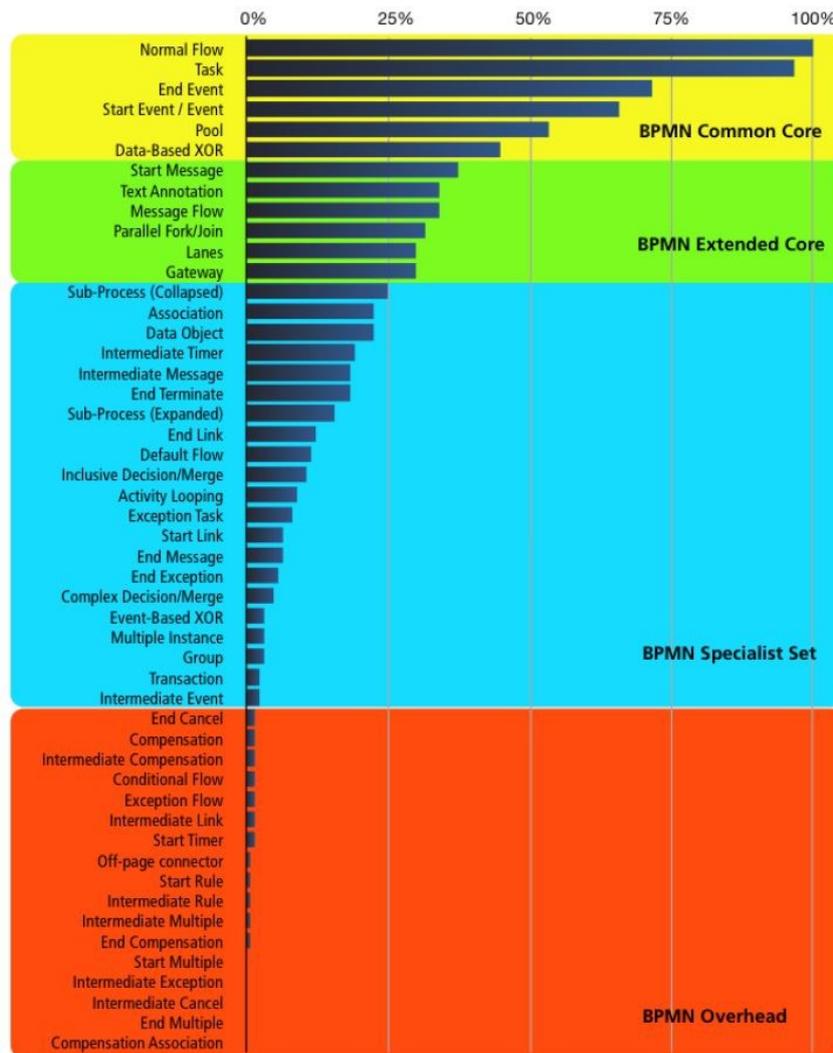


Figure 4.43 : Degré d'utilisation des éléments graphiques de BPMN par groupe

Pour compléter ses recherches, Jan Recker a évalué un peu plus finement, dans [RECKER 10], le domaine d'utilisation de BPMN comme il est possible de le voir sur la Figure 4.44. Ces résultats proviennent en partie des nombreuses interviews utilisées pour [RECKER 08].

Objectifs de la modélisation des processus

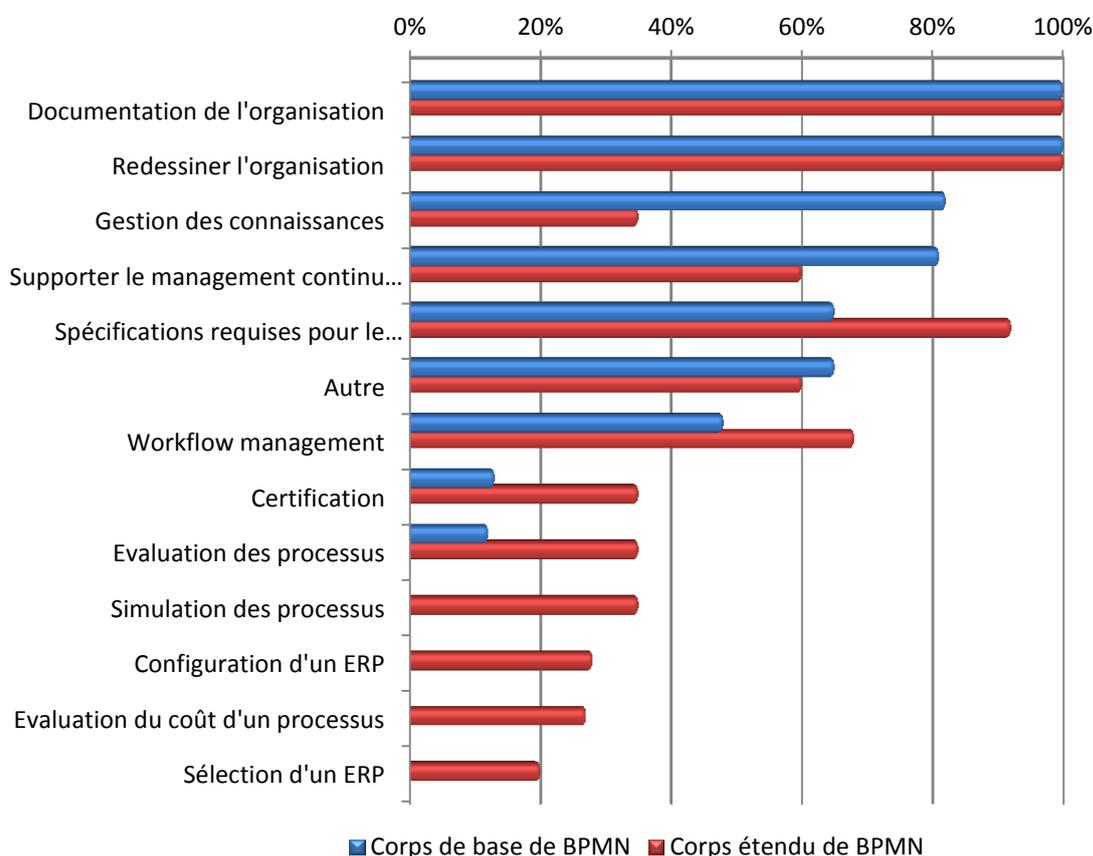


Figure 4.44 : Domaine d'utilisation de BPMN

BPMN est donc très sollicité pour mener une approche classique du management des processus, c'est-à-dire pour documenter, reconstruire, faire de l'amélioration continue, etc. alors que celui-ci est moins utilisé dans le cadre d'application technique telle que du développement logiciel ou de la simulation. Il est aussi constaté que le corps étendu est le plus utilisé quand il s'agit de traiter des problèmes plus complexes (en lien avec des ERP et pour du développement logiciel notamment).

4.3.10.2 BPMN, son utilisation dans le monde hospitalier

Bien que BPMN se soit imposé comme l'un des outils les plus performants pour la modélisation des processus métier dans le monde de l'industrie et du conseil, l'hôpital est une nouvelle fois en retard quant au traitement de cette problématique, qui pourtant, du moins en France, est désormais une obligation légale. Même si une modélisation complète des processus d'un hôpital sous BPMN n'a pas été trouvée, il existe néanmoins des exemples d'applications de ce langage au monde hospitalier. Parmi ces exemples il est possible de trouver :

- ❖ **[MULLER 11]** : les auteurs utilisent le domaine hospitalier pour son côté multidisciplinaire et intègrent dans BPMN une colorimétrie permettant d'identifier les acteurs dans un « Pool » sans nécessiter différentes « Lignes ». Ils soumettent même l'idée de colorer de plusieurs couleurs une tâche/activité réalisée par différents protagonistes pour gagner encore en clarté. Ils identifient certaines lacunes notamment à cette idée de tâche/activité réalisée par divers participants. Leur application concerne le suivi d'un patient pour une opération chirurgicale ;
- ❖ **[ROJO 08]** : application plus poussée de BPMN, les auteurs se penchent eux aussi sur la prise en charge d'un patient pour le traitement chirurgical d'une pathologie cette fois-ci. Ils en viennent donc à modéliser tout le processus de l'admission au bilan pathologique du patient en incluant toutes les tâches réalisées dans le service de pathologie. Pour cela ils ont construit deux modèles BPMN, l'un portant sur la planification de l'opération (niveau macroscopique) et le second sur la réalisation du bilan pathologique (niveau microscopique). Cet article se concentre majoritairement sur l'application de BPMN et n'étudie en rien le langage ;
- ❖ **[BUJANOWSKA 2011]** : bien que concernant les hôpitaux, cet article traite surtout des problématiques liées au matériel utilisé pour prodiguer les soins. Les auteurs se penchent donc sur une modélisation BPMN de la prise en charge des réparations par les services techniques pour divers équipements, ce qui se traduit par deux exemples de processus au niveau macroscopique/mésoscopique ;
- ❖ **[STRASSER 11]** : de nouveau en lien avec le domaine chirurgical, les auteurs ont construit un modèle BPMN en lien avec IHE²⁵. Ceci permet à BPMN de récupérer directement les informations nécessaires par le biais des composants IHE et donc de standardiser et automatiser l'exécution des processus cliniques ;
- ❖ **[PUUSTJÄRVI 06]** : plus ancienne, l'application faite ici de BPMN est basée sur la version 1.0 de ce langage. Les éléments graphiques utilisés sont donc assez basiques et le processus de petite taille. Néanmoins, il démontre que l'intérêt pour ce langage existait déjà quand il en était à ses débuts. Les auteurs l'ont donc utilisé pour construire le processus lié à la prescription informatique dans les hôpitaux.

Les diverses applications de BPMN sont donc assez variées en terme de domaine, et remontent en grande majorité à ces trois, quatre dernières années. Cependant, la plupart des études consultées ont été trop souvent menées à un niveau macroscopique et sur des processus de petite taille ne donnant pas de visibilité à ce langage dans le monde hospitalier et donc ne permettant pas à celui-ci de s'y imposer.

4.4 CONCLUSION

Ce chapitre nous a permis de présenter en détails deux des piliers de nos travaux à savoir la méthodologie ASCI ainsi que le langage BPMN.

ASCI propose une démarche globale enchaînant différentes étapes depuis la formalisation de la connaissance jusqu'à la conception d'outils d'aide à la décision, ce qui nous permet ainsi de

²⁵ Integrating the Healthcare Enterprise : <http://www.ihe.net/>

répondre à l'une de nos problématiques, à savoir disposer d'une méthodologie pour réaliser l'analyse du CDM.

Comme nous l'avons vu, ASCI a été utilisée à plusieurs reprises, dans le monde hospitalier notamment, et cela avec succès, permettant donc de souligner l'efficacité de cette méthodologie. La trame même de celle-ci rend possible la dissociation de la récolte de la connaissance avec son exploitation, ce qui se révèle être un atout indéniable, l'ensemble des informations obtenues pouvant être vu comme un socle solide et complet autorisant et facilitant la construction de l'outil désiré. La vue en cycle d'ASCI a permis d'en expliquer les grandes étapes, celles-ci étant au nombre de quatre. Débutant par l'analyse et la spécification du domaine étudié, dans le but de construire un modèle générique des connaissances du domaine, cette étape est suivie par la conception et l'implémentation, ponctuant alors la formalisation de la connaissance. Cette seconde étape a pour but de traduire le MGC en un format exploitable pour l'utilisateur, à savoir un modèle d'action. Les deux dernières étapes consistent alors en l'exploitation de ce modèle d'action dans le but d'obtenir un modèle de résultat orientant et validant les actions à mener sur le système étudié.

Le second pilier de nos travaux, BPMN, est un langage de modélisation graphique régi par de nombreuses règles offrant une approche centrée sur les processus, ce qui permet de les exploiter et de travailler dessus avec plus d'aisance. BPMN est un langage récent, à l'initiative de nombreuses structures, telles que l'OMG, et qui tente d'apporter des solutions pour travailler sur le BPM. Bien que nous ayons détaillé dans ce chapitre bon nombre des éléments composant ce langage, cela n'en reste qu'une partie, ce qui permet de souligner la grande diversité et la grande flexibilité de celui-ci, notamment au niveau des événements ponctuant les processus, qui ne sont pas moins qu'au nombre de 63. Le détail ici fait de BPMN reste bien entendu très succinct, et afin d'en comprendre toutes les possibilités, nous ne pourrions que recommander de consulter la documentation fournie par l'OMG sur son propre langage²⁶.

BPMN a suscité un véritable engouement ces dernières années, ce qui s'est traduit par une implantation de plus en plus importante dans des suites logicielles ainsi qu'à travers des outils dédiés. Par conséquent, il y a eu une véritable explosion de la population utilisant ce langage. En effet, ces derniers viennent désormais d'horizons très différents et le sollicitent pour des tâches très diverses. Cette propagation a même touché le monde hospitalier, où des travaux en lien avec ce langage ont été menés ces dernières années [[MULLER 11](#)], [[ROJO 08](#)].

Cette évolution ne doit cependant pas masquer les lacunes que BPMN subit. Tout d'abord, le fait que [[RECKER 08](#)] ait identifié Visio comme l'outil le plus utilisé pour construire les processus BPMN, prouve bien qu'il y a encore un manque sur le marché d'outils puissants (pour la construction graphique uniquement) et gratuits (tout du moins compris dans une suite). Notre choix d'utiliser Visio comme outil n'est donc pas illogique, du fait de sa disponibilité au CHU et de ses capacités de diffusion. Bien que cette constatation date de 2008 et 2009, elle est encore en partie d'actualité. De nombreux logiciels sont certes apparus depuis, mais ils s'avèrent être souvent payants et leur prise en main n'est pas toujours adaptée pour des personnes extérieures à ce domaine.

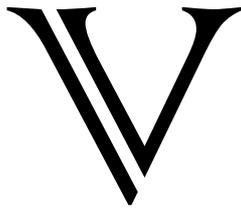
L'autre problème de BPMN est lié à son potentiel graphique. Constatée dans [[RECKER 08](#), [RECKER 09](#)], il est non seulement mis en avant que seuls cinq items sont utilisés par plus de 50% des utilisateurs (les portes décisionnelles et les lignes ne l'étant qu'à hauteur de 29%), mais aussi

²⁶ BPMN 2.0.2 : <http://www.omg.org/spec/BPMN/2.0.2/>

que même si BPMN est un des langages les plus aptes pour modéliser les processus métier (en termes de possibilité), l'auteur concède que cet avantage de diversité dans les possibilités de modélisation devient un inconvénient, le degré de complexité et de redondance de ce langage étant élevé. Il s'agit donc là d'une des explications du pourquoi de la sous-utilisation de ce langage, qui en vient donc à limiter son potentiel et son impact.

Ces constats ne doivent cependant pas décrédibiliser l'utilisation de BPMN qui reste grandement intuitive et facile à mettre en œuvre, l'emploi répété de ce dernier permettra de compenser la complexité de ce langage.

ASCI ayant été présenté d'un point de vue technique dans ce chapitre, le chapitre suivant se concentrera sur la présentation de la démarche adaptée de cette méthodologie et à destination du personnel hospitalier, notamment pharmacien dans notre situation.



ASCI POUR LES PHARMACIENS

RESUME DU CHAPITRE 5

Un pharmacien veut modifier le/les processus de sa pharmacie. Regardant avec un œil attentif nos divers outils, il se pose alors la question : « Intéressant, mais comment puis-je l'utiliser ? ». Là est toute la question de ce chapitre dont le but est de vulgariser la méthodologie ASCI sans pour autant en perdre son essence, l'objectif final étant l'obtention d'une démarche facilement utilisable et compréhensible par les pharmaciens.

SOMMAIRE DU CHAPITRE

5.1	INTRODUCTION	140
5.2	CONSTRUCTION DE NOTRE DEMARCHE.....	140
5.2.1	Les adaptations.....	140
5.2.2	Spécification des phases.....	142
5.2.3	Agencement des phases.....	143
5.2.4	Bilan outils & Comparaison ASCI/Démarche.....	144
5.3	CONCLUSION	148

5.1 INTRODUCTION

Jusqu' alors nous avons réfléchi à comment il serait possible de traiter, d'étudier et de modéliser le CDM en vue de le simuler et de l'améliorer. Durant cette phase de recherche et de réflexion, nous avons travaillé en commun avec les pharmaciens afin de choisir des outils pouvant leur convenir, tels que BPMN notamment. Mais il reste une question en suspens. Même s'ils seront capables d'utiliser les différents outils séparément, il faut encore construire un cadre d'utilisation de ces derniers pour éviter aux pharmaciens de s'éparpiller et de perdre du temps lors de leur exploitation. Nous avons vu dans le chapitre précédent, que la méthodologie ASCI se prête bien non seulement à ce type de démarche (amélioration et simulation d'un système) mais aussi aux systèmes hospitaliers. D'un point de vue ingénieur, concepteur, il n'y a rien à en redire, cependant d'un point de vue pharmacien, le ressenti n'en est pas le même.

Nous proposons donc une nouvelle démarche de modélisation exploitable par les pharmaciens. Cette démarche reprend les concepts d'ASCI et, suite à certaines modifications, les rend d'avantage explicites et exploitables pour cette profession. Par conséquent, le pharmacien n'aurait à intervenir que pour la modélisation de sa pharmacie.

Dans un premier temps, nous construirons notre propre démarche de modélisation en nous focalisant sur les aspects opérationnels, puis nous comparerons ASCI et notre démarche afin de mettre en lumière leurs points communs, tout en y incluant les différents outils que nous avons construits précédemment.

5.2 CONSTRUCTION DE NOTRE DEMARCHE

Etant donné que nous avons dans le Chapitre 4 présenté en détails le fonctionnement de chacune des étapes de la méthodologie ASCI, il est désormais possible de travailler sur un équivalent permettant au pharmacien de mettre en œuvre le processus de modélisation d'ASCI pour son système.

L'étude du déroulement d'ASCI a permis d'identifier neuf actions « majeures » sur lesquelles nous allons nous baser afin de construire une version adaptée aux pharmaciens :

1. Identifier les différents problèmes et les formaliser ;
2. Analyser les entités du système ;
3. Modéliser un/des flux d'objets ;
4. Obtenir un modèle de connaissance de son système (souhaité ou réel) ;
5. Concevoir « sur le papier » son futur modèle de simulation et ses composants ;
6. Coder et implémenter les différents éléments conçus ;
7. Générer le modèle de simulation ;
8. Faire fonctionner ce modèle et obtenir des résultats ;
9. Communiquer/Utiliser les résultats.

5.2.1 Les adaptations

Le pharmacien devant limiter son intervention à la modélisation de sa pharmacie, la phase de construction du modèle de simulation ne le concernera en aucun cas. Cela sera la première

adaptation majeure. Ensuite, étant donné que nous fournissons aux pharmaciens les outils adéquats pour travailler, les phases d'Analyse et de Spécification seront donc amenées à évoluer. Celles-ci ne consisteront plus en le développement d'un modèle des processus et d'une cartographie des interactions, mais en une traduction des évolutions (organisationnelles, structurelles, etc.) voulues, sur le modèle générique de connaissance fourni. Il sera donc impératif de retravailler sur la phase d'acquisition de la connaissance qui a lieu au début de la démarche.

Afin de mieux visualiser ces neuf actions majeures, nous les avons traduites sur le logigramme de la Figure 5.1.

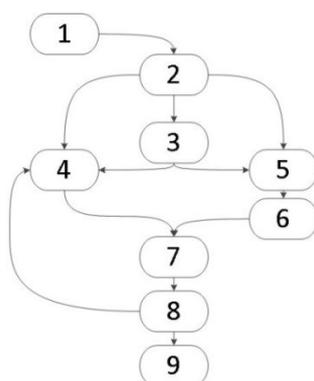


Figure 5.1 : Représentation numérique de l'équivalent d'ASCI

Lors de la première adaptation, les actions « 5 » et « 6 » vont disparaître. L'intitulé de l'action 7 en sera alors changé, ce dernier n'étant plus de « Générer le modèle de simulation » mais de « Paramétrer le modèle de simulation ». Bien que les deux intitulés soient similaires, « générer » pouvant être vu comme un paramétrage particulier du modèle de simulation, nous garderons le second intitulé afin que l'action devant être réalisée en « 7 » soit claire. Nous obtenons alors la démarche de la Figure 5.2 (a).

La seconde adaptation va elle aussi amener la suppression de deux actions, la « 2 » et la « 3 », celles-ci fusionnant avec la « 4 » afin d'obtenir une action « Construire le modèle du système » et non plus « Obtenir le modèle du système ». La Figure 5.2 (b) traduit cette évolution.

A noter qu'une dernière adaptation (Figure 5.2 (c)) est possible en vue de synthétiser la phase d'utilisation du modèle de simulation. L'action « 8 » pourrait être rapprochée avec la « 7 », celle-ci devenant l'action « Simuler le fonctionnement du système ».

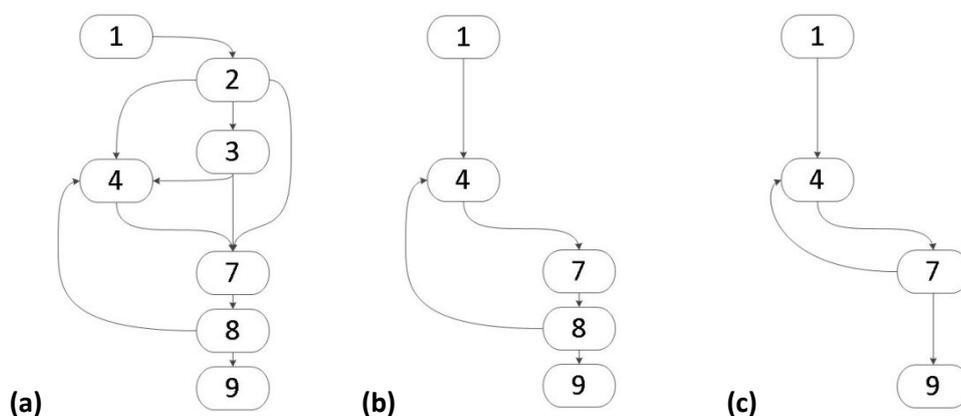


Figure 5.2 : Etapes d'adaptation d'ASCI en vue de l'obtention de notre démarche

Au final, il restera quatre actions de « synthèse » pour traduire ASCI de manière opérationnelle pour un pharmacien :

1. Identifier les problèmes du système ;
2. Construire le modèle du système ;
3. Simuler le fonctionnement du système ;
4. Communiquer/Utiliser les résultats.

Il n'est bien sûr pas pertinent de garder en l'état cette décomposition celle-ci omettant diverses tâches, actions qui sont essentielles au bon déroulement de la démarche. Il va donc être nécessaire de retravailler ces intitulés, afin d'en faire ressortir le véritable sens.

5.2.2 Spécification des phases

Désormais, nous ferons référence aux quatre actions précédentes sous le terme de phases. Parmi elles, la dernière ne nécessitant pas de grand changement, nous nous contenterons de la renommer « Proposition des évolutions aux décisionnaires ». Le travail va donc être axé sur les trois premières.

5.2.2.1 Phase 1 : Identifier les problèmes du système

L'identification des problèmes d'un système ne peut se contenter d'être un simple listing des différents problèmes que l'on y relève. Même s'il s'agit là du début, il est nécessaire d'identifier l'origine du problème, ou celui-ci se produit, etc. Un bilan de la situation de la pharmacie s'avère donc nécessaire. Ce bilan pourra notamment se faire sur deux lignes, une organisationnelle et l'autre structurelle, celles-ci permettant d'expliquer et de mettre en lumière le pourquoi des problèmes.

Celui-ci étant réalisé et en vue de passer à la phase 2, il faudra observer quelles sont les possibilités d'évolutions qui s'offrent aux pharmaciens pour faire progresser leur pharmacie. Cependant une telle démarche demandera de disposer de l'outil adéquat rendant une telle action possible. C'est donc à ce niveau que notre modèle générique des processus, conçu avec BPMN, remplira son rôle. Cet outil permettra de faire le lien entre le bilan de la situation et la prospection des évolutions envisageables, en permettant d'identifier le/les processus actuel(s) de la pharmacie et en donnant l'opportunité d'observer un/des processus parallèle(s), qui pourrai(en)t être implanté(s) dans la pharmacie. Au final, trois actions sont à mener durant cette phase (Figure 5.3 (a)) :

1. Faire un bilan de la situation organisationnelle et structurelle de la pharmacie ;
2. Identifier le(s) processus actuel(s) de la pharmacie ;
3. Etudier les possibles évolutions des processus de la pharmacie.

5.2.2.2 Phase 2 : Construire le modèle du système

Plutôt que de parler de « construction » au sens premier, ici, ce terme sera entendu plus dans le sens de la traduction des différentes évolutions choisies sur le système à étudier. Ces évolutions se traduiront par un impact plus ou moins important sur la structure du processus et donc sur la possible arrivée de nouvelles entités dans le système. Ces nouvelles entités pourront alors être caractérisées, grâce à notre second outil, le diagramme des classes UML, contenant dans l'absolu tout ce qu'il est possible d'avoir dans un CDM. La fin de la Phase 1 étant le moment où les différentes évolutions possibles sont étudiées, il sera nécessaire d'en sélectionner une ou plusieurs durant cette

seconde phase, avant de pouvoir identifier les nouvelles entités qu'elles apportent (ou qu'elles modifient). Il y aura donc deux actions dans cette phase (Figure 5.3 (b)) :

1. Choisir une ou plusieurs évolutions (structurelles et/ou organisationnelles) ;
2. Identifier et étudier les nouvelles entités à intégrer.

5.2.2.3 Phase 3 : Simuler le fonctionnement du système

A ce niveau, il est supposé que toutes les entités devant prendre part à la simulation et leurs caractéristiques sont connues. Nous avons mentionné plus tôt de paramétrage du modèle de simulation et de scénarios de simulation (5.3.2). Afin d'utiliser le modèle de simulation qui lui sera fourni, le pharmacien devra donc « paramétrer » des scénarios dans le modèle afin de tester les évolutions qu'il aura choisies. Ces scénarios seront donc directement dépendant des évolutions sélectionnées. Les concevoir sera donc la première action à mener dans cette phase. Il sera bien sûr ensuite nécessaire de les simuler, ce qui en résumé donnera deux actions dans cette phase (Figure 5.3 (c)) :

1. Constitution des différents scénarios de simulation ;
2. Simulation de ces différents scénarios.

5.2.2.4 Phase 4 : Communiquer/Utiliser les résultats

A ce niveau comme nous l'avons énoncé précédemment, deux options s'offrent au pharmacien. S'il est décideur, il peut mettre en application directe les évolutions qu'il a testées avec notre démarche. Sinon, il devra communiquer les résultats aux décisionnaires concernés. La seule action à effectuer est visible sur la Figure 5.3 (d).

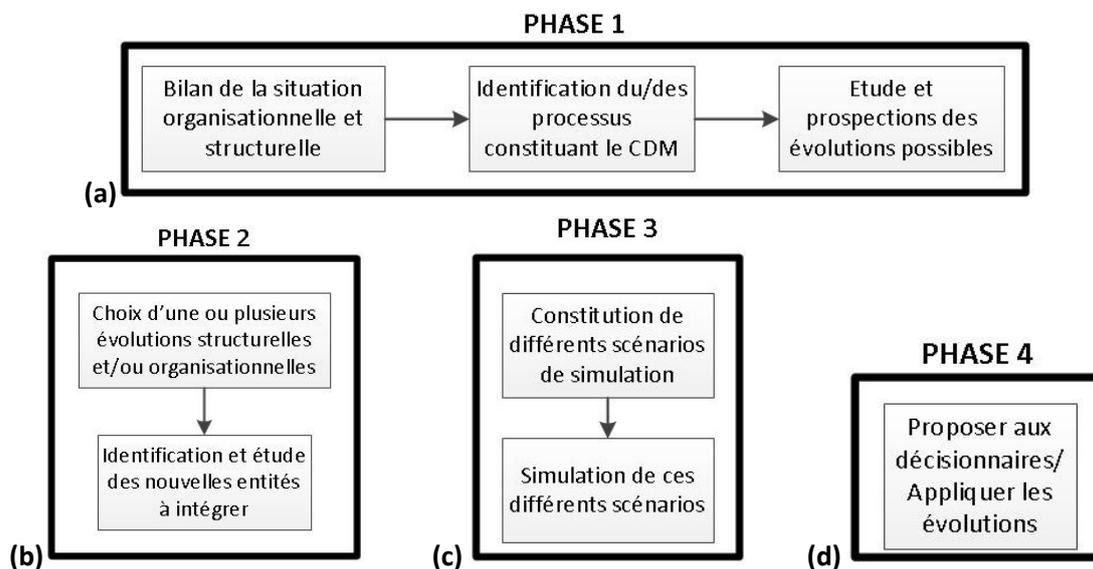


Figure 5.3 : Contenu des phases 1 (a), 2 (b), 3 (c) et 4 (d) de notre démarche

5.2.3 Agencement des phases

Désormais nous avons quatre phases ordonnées, mais entre lesquelles il manque encore des liens, des informations (Figure 5.4 (a)). En les liant suivant la logique de la Figure 5.2 (c), représentant ASCII, nous obtenons la Figure 5.4 (b).

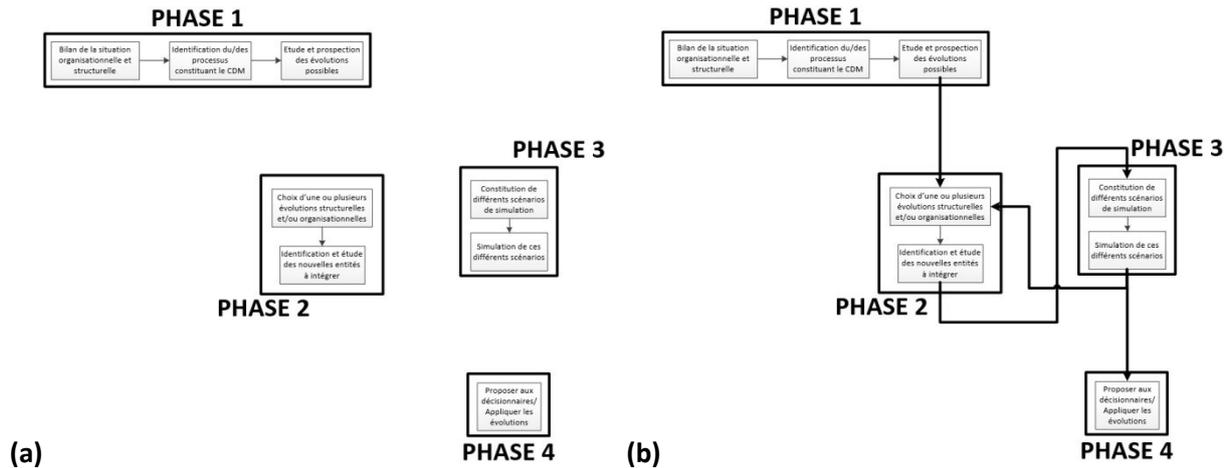


Figure 5.4 : Vue graphique de notre démarche sans (a) et avec (b) les liens hérités d'ASCI

Une telle représentation de la démarche limite cependant son potentiel étant donné qu'il n'est pas possible de modifier les évolutions choisies avant la fin de la simulation, ou de modifier uniquement le contenu du scénario simulé (son paramétrage) sans devoir revenir au choix des évolutions. Deux items pourraient donc être rajoutés dans cette démarche afin de lui apporter plus de flexibilité, à savoir après la Phase 1 et la Phase 3. En effet, après la première phase il serait de bon ton de regarder si les évolutions choisies sont réalistes et intégrables. Il n'est pas question ici de faisabilité, mais de logique. Et pour cause, se demander si augmenter l'équipe de préparateurs en pharmacie de 100 personnes n'est pas pertinent, aucun hôpital n'acceptant une telle évolution de ses équipes. De même, intégrer un robot de dispensation dont le coût cumulé serait de plusieurs millions d'euros dans une structure de 100 lits serait aberrant. La question est donc de « trier » déjà à un premier niveau les évolutions envisagées, mais tout en laissant le choix de ces critères de passage au pharmacien.

De même, après la Phase 3, les résultats tirés de la simulation peuvent être interprétés de deux manières. Soit la simulation nécessite juste un petit ajustement (augmenter le nombre de préparateurs de 1, faire varier légèrement les horaires), soit il est nécessaire de revoir complètement les choix précédemment faits et donc de modifier potentiellement tout le travail réalisé en Phase 2. Cela se traduit sur la Figure 5.5 par 3 possibilités en sortie de la Phase 3.

Maintenant que nous avons notre démarche dédiée à destination des pharmaciens, et afin de vérifier que celle-ci suit bien la trame de la méthodologie originelle, nous allons les comparer. De plus, les différents outils cités précédemment seront rajoutés sur les trames d'ASCI et de notre démarche.

5.2.4 Bilan outils & Comparaison ASCI/Démarche

Lors des précédentes parties, nous avons plus ou moins indiqué où les différents outils construits prendraient place dans notre démarche. Cependant, résumons dans un premier temps les besoins de chaque phase et étapes (Tableau 5.1).

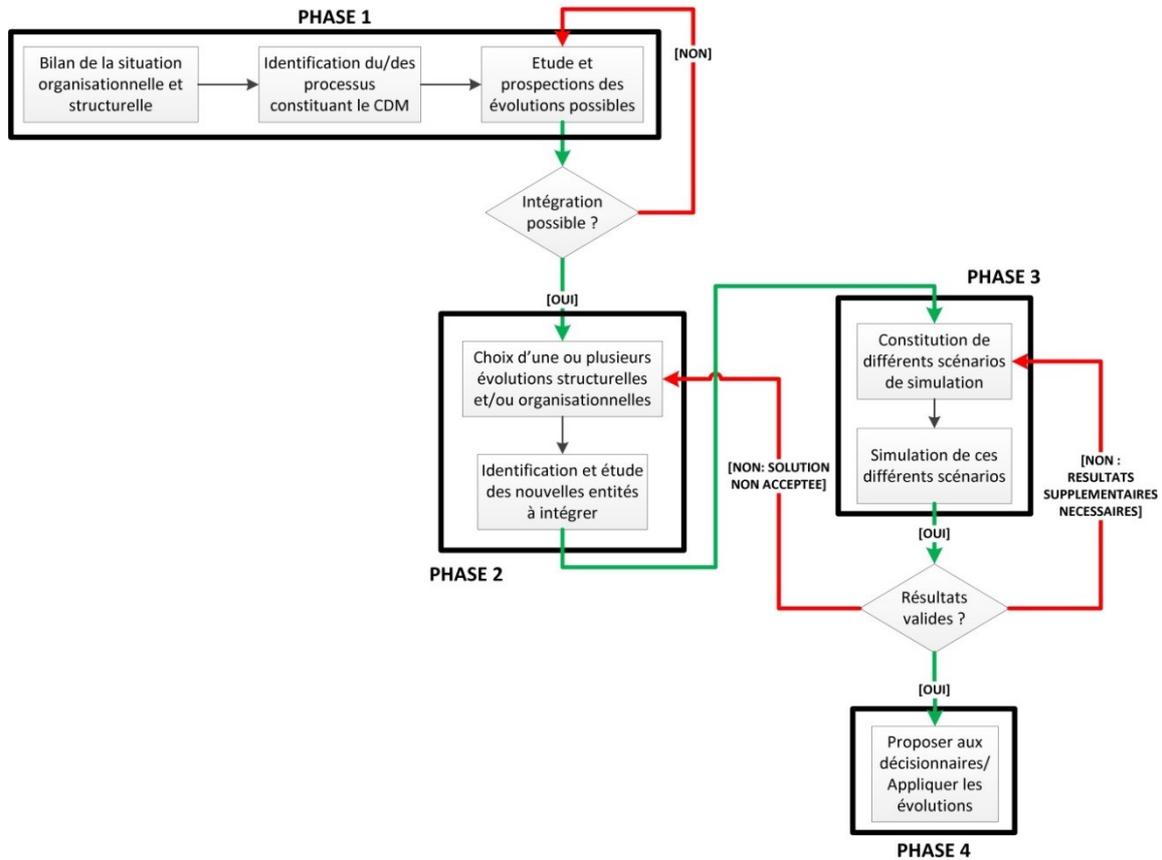


Figure 5.5 : Vue graphique complète de notre démarche

Tableau 5.1 : Bilan des besoins pour chaque phase de notre démarche

Phase	Etapes	Besoin
1	Bilan de la situation	<i>Cahier des charges</i>
	Identification des processus	&
	Etude et prospection	<i>Cartographie des processus</i>
2	Choix des évolutions	<i>Informations relatives aux nouvelles entités</i>
	Etude des nouvelles entités	
3	Constitution des scénarios	<i>Modèle de simulation</i>
	Simulation des scénarios	
4	Proposition des évolutions	

De tous ces besoins, seul le cahier des charges restera à l'entière responsabilité du praticien et des décisionnaires. Voyons rapidement le but et le pourquoi de chacun.

5.2.4.1 La cartographie des processus

Ce premier outil doit permettre de faire un bilan de la situation du système. L'idée sous-jacente est de permettre à cette cartographie d'être à la fois un outil de synthèse et un outil de prospection. Dans un premier temps, le circuit du médicament du système étudié pourra donc être

identifié sur cette cartographie, puis celle-ci permettra aux décisionnaires de voir quelles possibilités s'offrent à eux. Cette carte devrait donc contenir le présent et le futur. Cependant, ceci ne saurait être possible pour tous les hôpitaux à la fois, excepté dans le cadre de la construction d'une cartographie générique regroupant toutes les possibilités envisagées (à un certain degré de représentation bien sûr). Cette cartographie pourrait donc être vue dès lors comme un modèle générique.

5.2.4.2 Un recueil d'informations

Fournir aux décisionnaires un document permettant de voir quelles seront les informations importantes caractérisant les différentes nouvelles entités intégrant le système (par exemple, que faut-il comme données pour pouvoir évaluer un robot ?) simplifiera encore la démarche en limitant le recueil de donnée à ce qui est vraiment nécessaire.

De surcroît, ce document pourrait aussi être un complément au modèle générique des processus en se concentrant par la même occasion sur les interactions entre toutes les entités du système.

5.2.4.3 Le modèle simulation

Le rôle de cet outil est clair. Une fois toutes les données environnementales recueillies, il permettra de les exploiter afin de fournir l'impact des décisions qui seront prises sur le système. Celui-ci devra être de plus aussi simple que possible en termes d'utilisation et de compréhension (une partie graphique serait donc un plus).

Il va s'en dire que nous parlons donc ci-dessus des trois outils suivants : le modèle générique BPMN des processus, le diagramme générique des classes UML et le modèle de simulation générique sous SIMIO. Ainsi, dans le but de souligner les points communs entre ASCI et notre démarche, le Tableau 5.2 va présenter où ces différents outils s'insèrent dans chacune d'entre elles.

Tableau 5.2 : Lieu d'utilisation de nos outils dans ASCI et notre démarche

DEMARCHE		OUTILS	ASCI	
Phase	Etapes		Modèle généré	Phase
1	Bilan de la situation	Modèle générique BPMN des processus	Modèle générique de connaissance	Analyse & Spécification
	Identification des processus			
2	Etude et prospection	Diagramme générique des classes UML	Modèle d'action	Conception & Implémentation
	Choix des évolutions			
3	Etude des nouvelles entités	Modèle générique de simulation SIMIO	Modèle de résultats	
	Constitution des scénarios			
4	Simulation des scénarios	Rapport de simulation		
	Proposition des évolutions			

5.2.4.4 Représentation graphique du parallèle ASCI/Démarche

Sur la Figure 5.6, il est possible de voir une comparaison graphique des deux méthodologies permettant de situer un peu mieux les outils et les différentes étapes. Pour un souci de clarté, nous n'avons pas placé nos trois outils sur cette figure.

Il est donc possible d'y voir les similitudes entre elles, notamment le même point de départ, à savoir le « Modèle générique de connaissance du domaine » constitué dans notre cas du modèle générique BPMN des processus et du diagramme générique des classes UML. Les phases 1 et 2 de notre démarche concernent donc l'analyse et la spécification du système étudié, tandis que la Phase 3 consiste en l'utilisation du modèle d'action / de simulation construit.

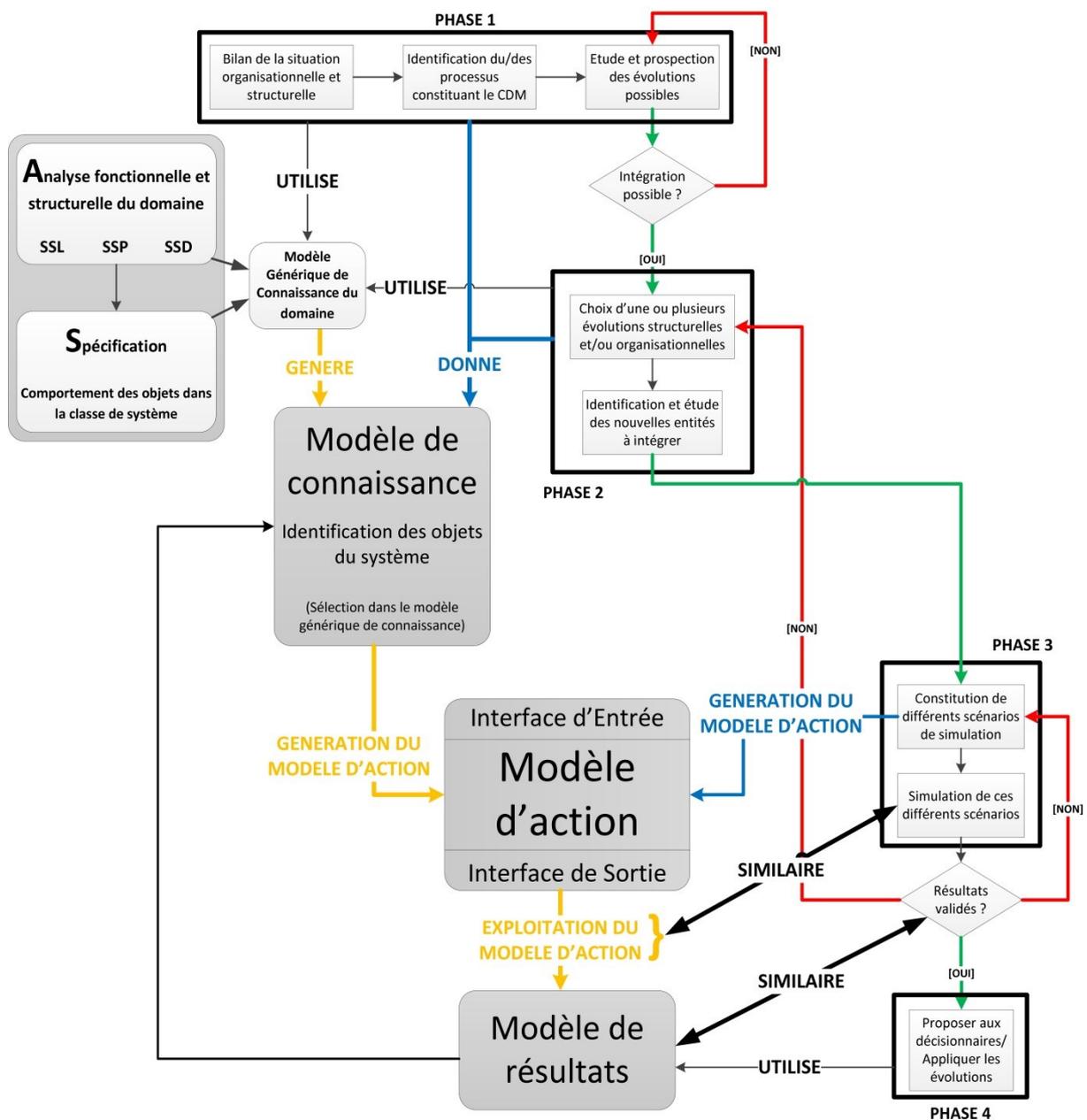


Figure 5.6 : Comparaison graphique entre ASCI et notre démarche

5.3 CONCLUSION

Bien que l'ensemble de nos travaux tournent autour de la méthodologie ASCI, nous ne pouvons demander et attendre des pharmaciens une compréhension et une mise en application parfaite de cette méthodologie et son utilisation. Ce chapitre avait donc pour objectif et a permis de concevoir une démarche adaptée d'ASCI, qui de par son contenu, peut trouver un écho auprès du personnel hospitalier et notamment les pharmaciens.

Au travers de ce chapitre, nous avons pour commencer, résumé la méthodologie ASCI en neuf étapes, dont il a été possible de tirer par la suite la trame principale de notre nouvelle démarche. Cela fait, et possédant désormais une trame constituée de quatre phases majeures, il a fallu spécifier le contenu de chacune d'entre elles, tout en gardant comme ligne de réflexion que, la formulation des différentes actions à mener dans ces phases devra être claire et sans ambiguïté.

Le contenu connu, nous avons alors identifié quels sont les outils ou les supports nécessaires à chacune de ces phases pour pouvoir être opérationnelles. Ainsi, la première d'entre elles requerra un cahier des charges et une cartographie des processus, tandis que la seconde demandera les informations relatives aux nouvelles entités, et que la troisième sollicitera un modèle de simulation. En expliquant le rôle et le pourquoi de chacun de ces besoins, il nous a été possible de faire le lien avec la méthodologie ASCI et ainsi identifier la teneur et la forme que ces divers besoins allaient prendre. Dès lors, il apparaît que les deux premières phases de notre démarche nécessiteront un modèle générique de connaissance du domaine, se trouvant sous la forme d'un modèle générique des processus du domaine – sous BPMN – et d'un diagramme générique des classes du domaine – sous UML. Enfin, le besoin de la troisième phase, le modèle de simulation, concorde avec le modèle d'action d'ASCI, et comme nous le verrons dans les chapitres 7 et 8, ce dernier sera développé avec le logiciel SIMIO.

La comparaison finale entre notre démarche et ASCI, a permis de mettre en avant les nombreuses similitudes entre les deux, qui en somme sont toutes logiques, mais surtout que notre démarche se révèle être une adaptation plus détaillée de la partie droite d'ASCI.

Possédant désormais une démarche susceptible d'être exploitée par les pharmaciens pour travailler d'eux-mêmes sur l'évolution de leur CDM, il est dorénavant temps de concevoir les outils nécessaires à la bonne marche de celle-ci. Le chapitre suivant présentera par conséquent comment le modèle générique des processus et le diagramme générique des classes ont été construits.

VII

CONCEPTION DU MODÈLE GÉNÉRIQUE DE CONNAISSANCE

RESUME DU CHAPITRE 6

L'objet de ce chapitre est de présenter les deux premiers outils que nous avons construits, à savoir le modèle générique des processus BPMN et le diagramme générique de classes UML, nécessaires à l'étude du CDM actuel et à la prospection des possibles évolutions de ce dernier. Dans la méthodologie ASCI, ces deux outils sont issus des phases d'analyse et de spécification du domaine, et constituent à eux deux, le modèle générique de connaissance du domaine.

Dans un premier temps, nous aborderons le modèle BPMN mais sans pour autant pouvoir en donner la représentation graphique complète. En effet, ce modèle nécessite actuellement trente feuilles A4 pour être lisible. Ainsi, afin de palier à ce manque, nous aborderons ici les nombreux acteurs et entités qui sont représentés dans le modèle, les diverses tâches réalisées par eux, comment ce modèle a-t-il été conçu, quelles simplifications ont été faites, etc. Cette partie sur BPMN sera de plus complétée par une présentation de quelques évolutions qui pourraient être appliquées à BPMN afin de le rendre plus facile à utiliser dans certaines situations.

Viendra ensuite la présentation du diagramme de classes UML qui comme nous le verrons a été basé sur les travaux de [[RODIER 10](#)] et [[HUET 11](#)]. La présentation de ce diagramme sera assez rapide du fait de ses origines et nous ne détaillerons pas de nouveau son contenu.

SOMMAIRE DU CHAPITRE

6.1	INTRODUCTION	151
6.2	MODELE GENERIQUE BPMN DES PROCESSUS	151
6.2.1	Les acteurs et entités du CDM.....	153
6.2.2	Les activités	155
6.2.3	Les points clés du processus.....	157
6.2.4	L'impact des points clés sur les actions des acteurs	159
6.2.5	Dernier pas vers un modèle final	161
6.2.6	Le modèle final	163
6.2.7	Les simplifications et hypothèses faites	171
6.2.8	Proposition d'évolution pour BPMN	172
6.3	DIAGRAMME GENERIQUE DE CLASSES UML.....	176
6.3.1	Le diagramme SSPDL de [RODIER 10]	176
6.3.2	Le diagramme SSPDL de [HUET 11]	176
6.3.3	Notre diagramme des classes.....	180
6.4	CONCLUSION	182

6.1 INTRODUCTION

Maintenant que nous avons fait le tour des prérequis pour l'élaboration de nos outils, nous pouvons commencer à construire ceux-ci afin de les mettre en application plus tard. Il va donc être nécessaire de construire un modèle BPMN générique des processus pouvant être trouvés dans une PUI, un diagramme générique des classes UML des entités existantes dans une PUI et enfin un modèle générique de simulation pouvant simuler virtuellement le fonctionnement d'une PUI (objet des Chapitres 7 et 8). Rappelons que ces deux premiers outils permettront de faire le bilan des processus actuellement présents dans une PUI et d'en étudier les évolutions possibles.

La conception du modèle BPMN va être détaillée dans ce qui suit en apportant au fur et à mesure de ce chapitre de plus en plus d'informations sur le contenu de ce modèle. Ainsi, nous commencerons dans un premier temps par présenter un « Macro-processus » et les informations relatives proposées par une des institutions françaises du domaine de la santé. Par la suite, les nombreux acteurs qui composeront ce modèle seront détaillés, en précisant leur place dans le système, ainsi que les activités de chacun. Il sera abordé ensuite les points-clés qui régissent et structurent le processus du CDM, ceux-ci pouvant être de deux natures différentes : des événements ou des décisions. Cette présentation des points-clés sera de plus complétée par une étude de leur impact sur le processus.

Ces divers éléments présentés, il sera alors possible de commencer la construction du processus afin d'obtenir au final un modèle complet des processus. Celui-ci ne pourra malheureusement pas être présenté dans son entièreté du fait de sa taille. Pour contourner ce problème, le modèle sera alors présenté sous forme de blocs mésoscopiques en détaillant ces différents blocs « pool » par « pool » pour y faire apparaître les nombreuses activités qui ont lieu dans chacun d'entre eux.

Pour terminer avec BPMN, il sera détaillé les nombreuses simplifications et hypothèses que nous avons faites pour construire le modèle BPMN, ces choix se justifiant autant d'un point de vue purement organisationnel que pratique. Enfin, nous proposerons diverses évolutions afin d'améliorer le fonctionnement de BPMN, en comblant ce qui semble être pour nous des lacunes majeures.

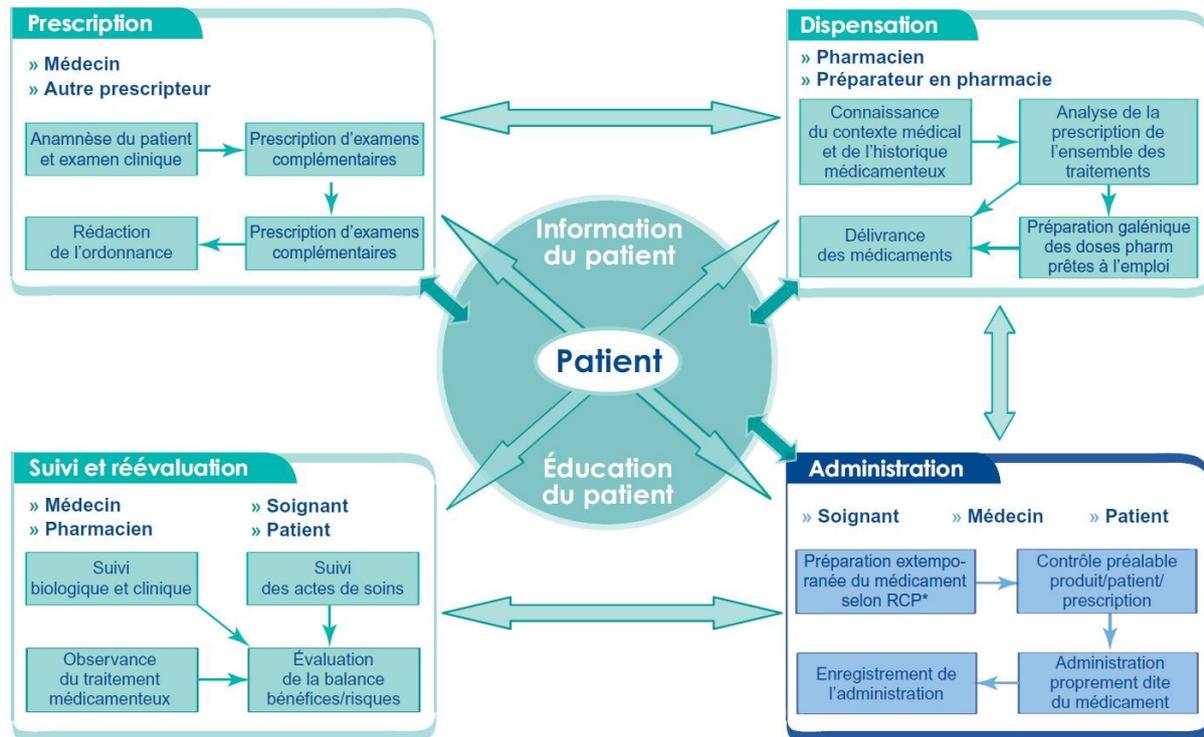
Le deuxième outil présenté ici sera le diagramme des classes UML, qui comme nous le verrons se base sur les travaux de [[RODIER 10](#)] et ceux de [[HUET 11](#)]. La présentation de notre diagramme sera précédée par celles des diagrammes SSPDL des deux auteurs précédents, et sera suivie par un détail rapide des modifications que nous avons apportées sur celui-ci et son contenu.

6.2 MODELE GENERIQUE BPMN DES PROCESSUS

BPMN va nous permettre de construire un modèle générique des processus, qui comme son nom le sous-entend intégrera autant d'entités et d'acteurs qu'il est possible d'en trouver dans le CDM. Par conséquent, cette partie va présenter comment nous sommes parvenus à construire ce modèle.

Il existe à ce jour des modélisations du CDM, mais elles sont malheureusement majoritairement macroscopiques. L'une d'entre elles sera notre point de départ, notamment car elle a été construite par la HAS (Haute Autorité de Santé), une structure gouvernementale dont le travail

a toute légitimité dans ce domaine. Cette vue macroscopique du CDM, ou « Macro-processus » comme elle le désigne, est visible sur la Figure 6.1 et présentée dans [HAS 11].



* RCP : Résumé des caractéristiques du produit - © Société française de pharmacie clinique 2005

Figure 6.1 : Macro-processus du CDM issu de [HAS 11]

Sur ce Macro-processus, il est donc possible de trouver les principaux acteurs intervenants sur le CDM, ainsi que leurs tâches majeures et une vue agrégée de leurs diverses activités. La HAS a réparti suivant quatre thématiques ces différentes tâches comme nous pouvons le voir, celles-ci pouvant être ordonnées de la manière suivante : Prescription, Dispensation, Administration et enfin Suivi et réévaluation. En plus de ce Macro-processus, la HAS fournit dans le même document, [HAS 11], une liste (Figure 6.2) apportant des précisions quant aux tâches réalisées par les acteurs cités dans la Figure 6.1. Sur cette Figure 6.2, il sera donc trouvé, le prescripteur, le pharmacien, le préparateur en pharmacie, l'infirmier, l'aide-soignant et enfin le patient.

Comme nous l'avons déjà dit, les processus que nous allons construire ne peuvent se contenter d'être une simple superposition ou juxtaposition de tâches ayant lieu dans le CDM. Il faut donc raisonner en termes de flux, qui ont un point d'entrée et de sortie dans ce circuit, afin de construire un ensemble cohérent s'articulant autour de points clés. Ces points clés (partie 6.2.3) permettront de découper le processus en diverses sous-parties, certaines d'entre elles prenant alors un niveau de granularité plus ou moins élevé.

Dans ce qui suit, nous allons donc commencer par présenter les acteurs et entités agissant dans le CDM en se basant sur les deux premières figures issues de [HAS 11]. Nous poursuivrons par la présentation des activités liées à ces entités avant d'exposer les points clés du processus. Enfin les simplifications réalisées durant la modélisation seront détaillées avant de soumettre certaines améliorations que nous proposons pour BPMN.

Prescripteur ¹	Pharmacien	Préparateur en pharmacie	Infirmier (ères)	Aide-soignant(s)	Patient ²
<ul style="list-style-type: none"> ■ Prend une décision thérapeutique ■ Prescrit les médicaments selon la conformité à la réglementation, aux référentiels scientifiques actuels et de manière adaptée à l'état du patient ■ Finalise sa prescription par la rédaction d'une ordonnance, enregistre sa prescription dans le dossier du patient ■ Informe le patient et s'assure de son consentement le cas échéant écrit ■ Évalue l'efficacité du traitement et sa bonne tolérance ■ Réévalue la balance bénéfiques/risques 	<ul style="list-style-type: none"> ■ Analyse et valide la prescription ■ Emet une opinion en tant que de besoin ■ Prépare et fabrique les médicaments en tant que de besoin ■ Délivre les médicaments en s'assurant de la maîtrise des stocks ■ Assure la mise à disposition des informations nécessaires au professionnel de santé et au patient 	<ul style="list-style-type: none"> ■ Prépare et fabrique les médicaments sous le contrôle effectif du pharmacien ■ Délivre les médicaments sous contrôle effectif du pharmacien ■ Participe à la gestion des stocks 	<ul style="list-style-type: none"> ■ Vérifie la prescription ■ Prépare les doses à administrer extemporanément ■ Réassort le stock ■ Vérifie la concordance entre la prescription, le médicament et le patient ■ Informe le patient et obtient son consentement ■ Administre les médicaments au patient ■ Enregistre l'acte d'administration ■ Suit les effets attendus et les réactions éventuelles 	<ul style="list-style-type: none"> ■ Aide à la prise sous la responsabilité de l'infirmier(ère) ■ Informe l'infirmier(ère) de toute modification d'état du patient ■ Participe au maintien de l'autonomie et à l'éducation du patient 	<ul style="list-style-type: none"> ■ Informe les précédents acteurs de ses traitements habituels pour conciliation ■ Communique les renseignements relatifs à ses facteurs de risque et allergies connues ■ S'informe sur son traitement et les effets indésirables éventuels ■ Observe les indications de bon usage du médicament ■ Participe en tant que partenaire de sa prise en charge médicamenteuse
Participent à l'éducation du patient / Participent à la surveillance du patient / Notifient les incidents/erreurs médicamenteuses					

1. Selon la réglementation en vigueur
 2. Selon son niveau d'autonomie

Figure 6.2 : Liste des activités réalisées par les différents acteurs du CDM [HAS 11]

6.2.1 Les acteurs et entités du CDM

Etant un modèle des processus supposé générique, c'est-à-dire s'adaptant à tous les hôpitaux, il est nécessaire d'y intégrer tous les acteurs et entités pouvant y être trouvés. Cela implique donc de considérer toutes les parties prenantes du service à la pharmacie.

6.2.1.1 Les pôles

Pour commencer, il est nécessaire de séparer les différents acteurs et entités du système suivant quatre pôles. En effet, les processus peuvent être triés suivant le pôle « Service », le pôle « Logiciel de gestion de la pharmacie », le pôle « Logistique » et bien entendu le pôle « Pharmacie ». Chacun de ces pôles sera donc représenté dans le modèle BPMN par un « Pool » dans lequel apparaîtront les différents acteurs et entités.

6.2.1.2 Les différentes catégories d'acteurs

Afin de simplifier la suite du discours, quand nous parlerons d' « acteur » nous ferons référence autant à une personne physique qu'à une personne morale.

Les nombreux acteurs du CDM peuvent être groupés selon trois catégories : Humain, Robot/Automate, et Autre. Le Tableau 6.1 permet de voir ces nombreux acteurs. De surcroît il précise dans quel « Pool » ils seront intégrés :

Tableau 6.1 : Répartition des acteurs par « Pool »

Acteur	Type			Pool
	Humain	Robot/Automate	Autre	
Médecin/Prescripteur	X			Service
Infirmière	X			Service
Armoire automatisée		X		Service
Pharmacien	X			Service Pharmacie
Préparateur en pharmacie	X			Pharmacie
Robot de dispensation		X		Pharmacie
Stocker mécanique		X		Pharmacie
Robot de reconditionnement		X		Pharmacie
Logiciel de gestion			X	Logiciel de gestion
Agent logistique	X			Logistique

De nombreux acteurs ici présents sont identiques à ceux pouvant être trouvés dans la Figure 6.2. Parmi ces derniers, l'aide-soignant n'apparaît pas étant donné que la plupart de ses tâches sont similaires à celles de l'infirmière. Les deux « acteurs » seront donc « fusionnés » pour plus de clarté. De même, le patient n'apparaît pas, ce dernier n'interagissant que passivement dans nos processus (il les subit).

6.2.1.3 Pourquoi ces acteurs ?

La question du pourquoi de ces différents acteurs peut se poser. Réfléchissons « pool » par « pool ». Au niveau du service, il est impératif de retrouver au moins les acteurs médecin et infirmière sans lesquels les soins ne pourraient avoir lieu. La notion de prescripteur est associée au médecin afin de bien mettre en valeur que c'est lui qui définit le traitement des patients. Le pharmacien est présent dans le service pour deux raisons. Tout d'abord, celui-ci peut participer aux « tournées » des médecins dans les services et les accompagner lors de la définition du traitement (conseil, correction, etc.). La seconde est que les pharmaciens peuvent être présents dans les services pour faire de la « conciliation des traitements médicamenteux²⁷ » (notée **CTM**) et de l'accompagnement (éducation) aux patients pour gérer leur traitement. Enfin, l'utilisation possible d'armoires automatisées dans les services, impose de considérer celles-ci dans le processus. En effet,

²⁷OMS : https://www.high5s.org/pub/Manual/TrainingMaterials/MR_Getting_Started_Kit.pdf

quand utilisées, elles feront le lien entre le service et la pharmacie, ces armoires passant automatiquement des commandes à cette dernière quand le besoin s'en fait sentir.

Au niveau de la pharmacie, il est évident que les pharmaciens et les préparateurs en pharmacie y ont leur place. La présence moins évidente des différents robots et stockeurs est cependant à expliquer. L'évolution des techniques robotiques a permis de renforcer la sécurité et l'efficacité du CDM ces dernières années. Prendre en considération ces différents types de robots dans nos processus est donc désormais une obligation. En effet, comme nous l'avons présenté dans le Chapitre 1, il s'agit de l'une des voies les plus efficaces pour sécuriser le CDM.

Il en va de même pour les stockeurs mécaniques qui présentent de nombreux avantages tels que leurs capacités de stockage, leurs faibles encombrements et la possibilité de les interfacer avec des logiciels de gestion de pharmacie tels que Pharma©.

En ce qui concerne les deux derniers « pool » à savoir la logistique et le logiciel de gestion, les deux acteurs y étant présents jouent des rôles très spécifiques ce qui explique pourquoi ils ont leur propre « pool ». Le logiciel de gestion sera à la croisée de toutes les informations transitant dans la pharmacie, pouvant être le support des prescriptions, des commandes, de la production, etc. L'agent logistique prendra place entre la pharmacie et le service en ce qui nous concerne. Il s'occupera de faire transiter les produits de l'un vers l'autre quand il sera sollicité.

6.2.2 Les activités

Connaissant désormais les acteurs principaux (à noter que des acteurs secondaires sont susceptibles d'apparaître lors de la modélisation du processus), il est possible de lister et d'étudier les nombreuses tâches et activités qu'ils réalisent. Cependant, connaître avec quel degré de granularité nous allons représenter ces différentes actions est nécessaire pour pouvoir avancer dans notre modélisation.

6.2.2.1 Quel degré de granularité/modélisation ?

Comme déjà énoncé à plusieurs reprises, ce degré peut varier suivant les besoins que l'on aura et suivant l'utilité et l'impact de l'action modélisée sur le système. En se référant à la partie 2.5.5 et aux Figures 2.19 et 2.20, il apparaît une ambiguïté dans notre travail. En effet, pour obtenir un modèle générique, un niveau de modélisation macroscopique serait requis, tandis que notre volonté d'obtenir un modèle pouvant servir aux pharmaciens pour étudier de nouvelles solutions pour leurs processus, nous pousserait plus vers un niveau mésoscopique/microscopique. Un choix se doit donc d'être fait.

Mais est-ce réellement nécessaire ?

En effet, rien n'impose à notre système d'être modélisé avec le même degré de granularité partout. Il est tout à fait envisageable de trouver certaines actions qui resteront macroscopiques dans leur modélisation. Par exemple, « Passer une commande » qui sous-entend de faire certaines démarches pour cela, mais encore « Transporter les demandes/produits » où il serait nécessaire préalablement de charger le moyen de transport, faire le trajet, etc. Le niveau microscopique n'est donc pas requis partout, et si la réflexion est poussée plus loin, il ne le serait même nulle part. En effet, en se référant aux Figures 2.19 et 2.20 une nouvelle fois, il est constaté qu'un niveau microscopique de modélisation a tendance à priver de sa généralité le modèle conçu en le dédiant à un système précis. Il serait donc bon de viser un degré de granularité oscillant autour du mésoscopique, tantôt penchant vers macroscopique, tantôt vers microscopique.

6.2.2.2 Les activités par acteur

Les activités qui vont être présentées ci-dessous ne sont encore pour beaucoup qu'une vue générale des tâches et actions à effectuer. Un listing plus précis de ces différentes activités sera présenté une fois toute la subtilité du processus connue (Partie 6.2.5.1).

Ainsi, en se référant à la Figure 6.2 nous pouvons déjà commencer à dresser une pré-liste des actions réalisées par les différents acteurs :

- ❖ Le médecin/prescripteur : parmi les tâches énoncées sur cette figure, nous ne nous intéresserons pas aux trois dernières, celles-ci étant pour nous transparentes ;
- ❖ Le pharmacien : pour nous, le pharmacien n'est pas la personne qui délivre physiquement les médicaments aux services et patients. Celui-ci valide bien les ordonnances et prescriptions mais il incombera aux préparateurs de délivrer réellement les produits. De plus, la phase de fabrication des préparations hospitalières ne sera pas modéliser dans le processus standard ;
- ❖ Le préparateur en pharmacie : les diverses tâches énoncées sur la figure seront toutes reprises mis à part celle concernant la préparation pour la cause énoncée ci-dessus ;
- ❖ L'infirmière : nous ne considérerons pas ici l'acte de suivi des effets qui sera pour nous de nouveau un acte transparent. Le reste sera pris en compte.

En plus de ces acteurs, il y a ceux que nous avons introduits avec le Tableau 6.1 dont les tâches majeures sont pour :

- ❖ L'armoire automatisée : en plus des contrôles qu'elle réalise lors du prélèvement du médicament, elle peut lancer des commandes automatiques ainsi qu'émettre des alertes en cas de ruptures de stocks ;
- ❖ Le robot de dispensation : comme son nom l'indique, celui-ci sera chargé de préparer les demandes des services ou les prescriptions des patients suivant son mode d'utilisation. Il communiquera donc avec le logiciel de gestion de la pharmacie mais pourra aussi interagir avec les préparateurs dans des cas bien précis ;
- ❖ Le robot de reconditionnement : pouvant être automatisé ou manuel, ce robot interagira donc avec des préparateurs si besoin, et pourra aussi être interfacé avec le logiciel de gestion de la pharmacie ;
- ❖ Le stockeur mécanique : lui aussi pouvant être en lien direct avec le logiciel de gestion, il interagira tout comme ces deux prédécesseurs avec les préparateurs. Il ne sera cependant pas en mesure de préparer par lui-même des demandes ou de fournir des produits ;
- ❖ Le logiciel de gestion de la pharmacie : deux choix sont possibles le concernant. Un raisonnement « boîte noire » où les différentes actions qu'ils réalisent ne nous concernent pas. Ou bien le mode « Boîte blanche » où nous détaillerons les différentes interactions qu'il a avec les différents robots et opérateurs humains ;
- ❖ L'agent logistique : tout comme le logiciel, son rôle sera vu d'une manière macroscopique où seule la tâche « Transporter » nous importera dans la plupart des cas. Dans le cas où des appareils de contrôles de livraison seraient utilisés, les actions en lien seraient à faire apparaître.

6.2.3 Les points clés du processus

Nous venons de commencer à dégrossir les activités que les différents acteurs réalisent dans le CDM. Les processus liés à chaque acteur sont pour le moment assez linéaire et ne tiennent pas compte des différents événements et choix qui se produisent sur le processus global. Chacun de ces items aura une conséquence sur le système, comme nous allons le voir, et rythmera dans certains cas son avancée. Dans un premier temps, il sera retrouvé les événements rythmant l'avancée du processus, souvent assimilés à des horaires ou des délais, et dans un second temps apparaîtront les décisions qui orientent le processus dans sa progression.

Il est à noter que les différents items présentés dans ce qui suit sont directement tirés du formalisme BPMN. Cependant, afin de faire le lien entre le système réel et ce langage, il nous a été nécessaire d'identifier dans un premier temps les différents points-clés sur le circuit réel. Ainsi, et afin de gagner en clarté et ne pas se répéter, ces différents points-clés sont finalement présentés directement dans leur forme BPMN.

6.2.3.1 Les événements rythmant notre processus

Durant la progression du processus, il y aura pour chaque acteur des événements qui rythmeront son évolution. Comme nous l'avons vu dans le Chapitre 4, ces différents événements peuvent être de types très variés. Il sera majoritairement trouvé dans notre modèle BPMN des événements de type « timer », « conditionnel », « normal », « message », « erreur » et enfin « signal ». Le Tableau 6.2 présente les différents événements triés par type et par acteur :

Tableau 6.2 : Evènements rythmant l'avancée du processus

Type évènement	Acteur	Description	Numéro
Début 	Médecin	<i>Patient en attente</i>	<i>E1</i>
Conditionnel 	Infirmière	<i>Demande reçue</i> <i>Médicament arrivé</i>	<i>E2 et E3</i> <i>E4</i>
	Armoire automatisée	<i>Armoire réapprovisionnée</i>	<i>E5</i>
	Préparateur en pharmacie	<i>Prescription validée</i> <i>Préparateur devant armoire</i>	<i>E6</i> <i>E7</i>
	Robot de dispensation	<i>Interruption pour une commande externe</i>	<i>E8</i>
Timer 	Infirmière	<i>Heure préparation traitements</i> <i>Heure administration traitement</i>	<i>E9</i> <i>E10</i>
	Préparateur en pharmacie	<i>Heure préparation des demandes/prescriptions</i>	<i>E11</i>
		<i>Heure préparation réapprovisionnement stock</i>	<i>E12</i>
		<i>Heure réapprovisionnement stock</i>	<i>E13</i>
		<i>Heure livraison demande(s) ou demande(s)/prescription(s) prête(s)</i>	<i>E14</i>
	<i>Préparation du robot terminée</i>	<i>E15</i>	
Signal 	Armoire automatisée	<i>Demande de réapprovisionnement automatique</i>	<i>E16</i>
	Préparateur en pharmacie	<i>Demande(s)/Prescription(s) expédiée(s)</i>	<i>E17</i>
	Stockeur mécanique Robot de dispensation	<i>Produit insuffisamment en stock</i>	<i>E18</i> <i>E19</i>

	Message	Médecin	<i>Demande d'information/de modification</i>	E20
		Pharmacien	<i>Nouvelle prescription</i>	E21
			<i>Réponse du médecin</i>	E22
		Préparateur	<i>Demande globale en main propre</i>	E23
	Erreur	Infirmière	<i>Médicament insuffisamment en stock en armoire automatisée</i>	E24
			<i>Médicament insuffisamment en stock en armoire standard</i>	E25
			<i>Manque un médicament</i>	E26
			<i>Un ou plusieurs médicaments/traitements ne sont plus utilisés ou sont en surplus</i>	E27

Comme il est possible de le constater, les événements présents dans ce tableau touchent à toutes les composantes du CDM (préparation, administration, contrôle, etc.), accentuant le fait que le processus du circuit est dépendant de ces événements et rythmé par eux.

6.2.3.2 Les décisions orientant notre processus

Tout comme les événements, des décisions organisationnelles sont à prendre durant la progression du processus afin d'orienter celui-ci. Représentées par le biais de portes décisionnelles, ces interrogations prennent place à divers endroits du processus et cela au niveau de tous les acteurs. Parmi elles, certaines auront une importance moindre, tandis que d'autres seront capitales et définiront à elles seules le type de CDM que représentera le processus. Nous allons ci-dessous lister les grandes questions qui jalonnent le processus et nous les lierons à des éléments du Chapitre 1, à savoir les nombreuses recommandations permettant d'améliorer l'efficacité du CDM.

Tableau 6.3 : Décisions orientant l'évolution du processus

	Décision	Choix possibles	Partie du Chapitre 1
D1	Un pharmacien est-il présent durant la consultation/visite	Oui/Non	1.4.3.6
D2	Type de dispensation	DJIN/Autres	1.4.3.4
D3	Type de gestion de la prescription	Papier/Informatique	1.4.3.1
D4	Type de dispensation	Nominative Partielle/Globale/Globalisée	1.4.3.4
D5	Type d'armoires de stockage utilisées	Standard/Plein-Vide/Automatisée	1.4.3.2
D6	Les prescriptions sont-elles validées par un pharmacien	Oui/Non	1.4.2.2
D7	Type de distribution	Globalisée/Nominative ou Partielle	1.4.3.4
D8	Type de préparation	Manuelle/Robotisée	1.4.3.5
D9	Type de livraison	Agent logistique/Préparateur	Aucune
D10	Type de gestion du plan de soins	Papier/Informatisée	1.4.3.1

Faisons un point rapide sur ce que chacune de ces décisions impliquera pour le CDM. La décision D1 rend possible la participation d'un pharmacien aux consultations et visites, permettant

une validation des prescriptions en direct et allégeant donc le travail de validation qui a lieu en aval, et son absence sur le terrain rendant l'interprétation des prescriptions plus complexes.

Les décisions D2 et D4 impacteront directement le mode de commande des demandes et des prescriptions à la PUI et seront par conséquent liées à la décision D3. En effet, D3 est interdépendante avec celles-ci étant donné qu'une gestion papier des demandes et prescriptions ne pourra pas être faite avec de la DJIN (ou difficilement) et ne sera applicable que pour de la « Globale », la « Globalisée » étant par définition informatisée pour nous.

La décision D5 aura un effet sur le mode de commande et les quantités commandées par les infirmières. L'acte de validation sera décidé dans la décision D6, tandis que D7 et D8 impacteront le mode de préparation des prescriptions et demandes à la pharmacie. En effet, avec une dispensation « Globale » ou « Globalisée », la préparation sera forcément manuelle, mais avec la possibilité d'utiliser des stockeurs mécaniques. Les autres pourront au contraire disposer d'une préparation robotisée (si faisable et prévue) nécessitant donc la présence de robots de dispensation pour réaliser ces actes.

La décision D9 donnera la possibilité de faire livrer les demandes/prescriptions par le préparateur lui-même, apportant un peu plus de sécurité avec une chaîne contrôlée de bout en bout par la PUI. De plus, une telle action peut permettre un meilleur suivi des stocks des services dans le cas où le préparateur irait jusqu'à ranger les produits en armoire et donc à contrôler les stocks.

Enfin, D10 permet deux types de suivi, l'un suivant les dernières recommandations (plan de soins informatisé avec une optique de dossier patient informatisé dans le futur), et l'autre avec un plan de soin à l'« ancienne », c'est-à-dire une version papier.

6.2.4 L'impact des points clés sur les actions des acteurs

Comme énoncé à plusieurs reprises, les différents points-clés présentés ci-dessus vont impacter le processus du CDM. Ces différents points vont devoir être intégrés aux ébauches des processus des acteurs afin de commencer à construire un modèle BPMN plus complet. Pour réussir à visualiser le début de notre modèle, nous avons construit une vue événement/décision permettant de pré-structurer notre modèle. Nous avons donc repris les nombreux points-clés identifiés et les avons ordonnés tel que cela peut se trouver dans la réalité. Les événements de type « erreur » (et les actions en découlant) n'apparaîtront pas sur ces vues, ces derniers se produisant dans des cas très spécifiques. Afin d'alléger les Figures 6.3 (a) et (b), nous avons utilisé les numérotations des Tableaux 6.2 et 6.3 précédents pour identifier les événements et les décisions. De plus nous avons rajouté sur chaque portion entre deux événements, deux décisions ou entre un événement et une décision, les acteurs qui seront impliqués dans les tâches qui ont lieu sur chaque section. Leurs identifiants sont :

- ❖ Ph : pharmacien ;
- ❖ M : médecin ;
- ❖ p : préparateur en pharmacie ;
- ❖ I : Infirmière ;
- ❖ A : armoire automatisée ;
- ❖ L : logiciel de gestion de la pharmacie ;
- ❖ S : stockeur ;
- ❖ R : robot de dispensation ;
- ❖ r : robot de déconditionnement.

La Figure 6.3 (a) présente l'ébauche du processus au niveau de l'unité de soins, tandis que la 6.3 (b) présente l'ébauche de celui de la pharmacie.

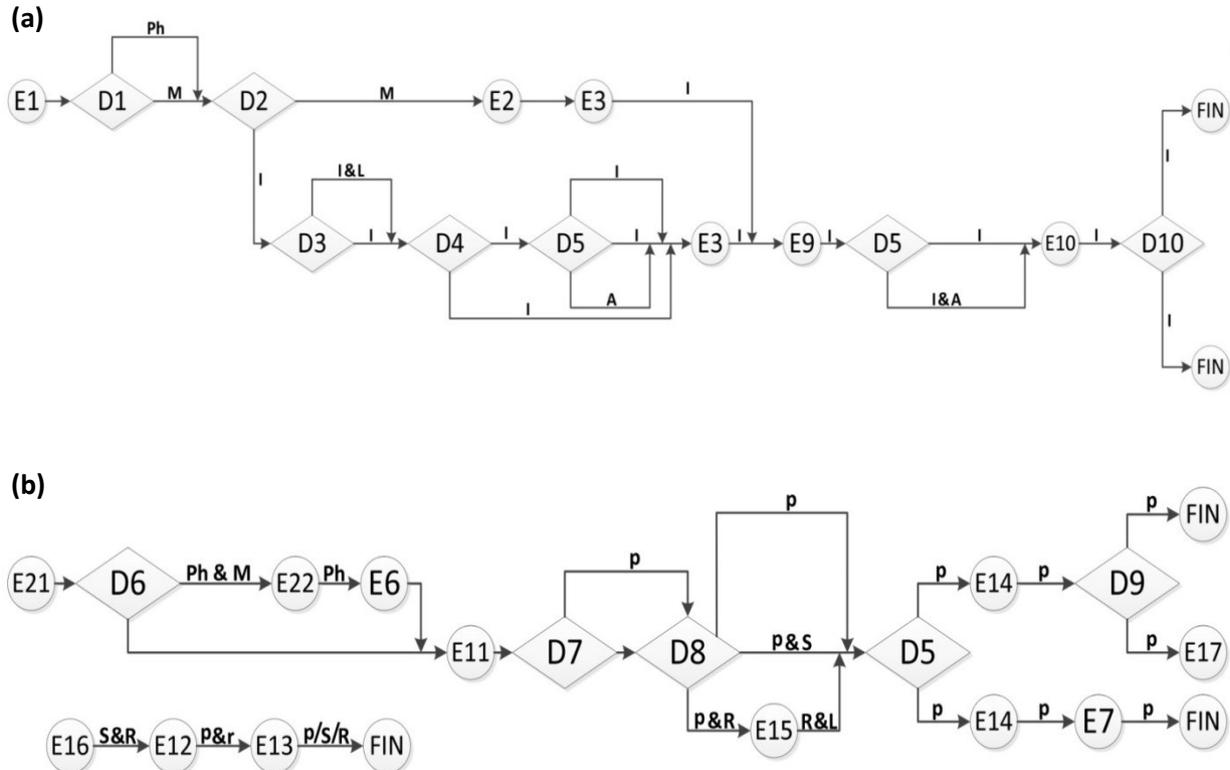


Figure 6.3 : Ebauche de la structure des processus composant le CDM : (a) unité de soins, (b) pharmacie

Ces deux premières ébauches nous permettent de voir que chaque processus possède une « zone de complexité », certes évidente mais qu'il était nécessaire de mettre en valeur. Ainsi, pour le processus de l'unité de soins, cette zone se situe au niveau de la création de la prescription et au niveau de la commande des produits à la pharmacie. Pour le processus de la pharmacie, cette zone sera la préparation des commandes. Ces constatations sont normales étant donné que c'est à ces deux endroits que se concentrent le plus de décisions à prendre et donc où se multiplient les possibilités de progression.

Il n'apparaît pas ici d'ébauche du processus du logiciel de gestion de la pharmacie. En effet, le cas de celui-ci étant spécial, il faut se pencher vers la simulation pour décider ce que nous allons faire. Etant donné que le fonctionnement du logiciel ne sera pas construit dans le modèle de simulation, il n'est pas nécessaire de construire le processus de celui-ci. Cependant, afin d'apporter un modèle complet, son processus sera bien conçu et implanté dans le modèle BPMN final, mais nous ne nous y attarderons pas dans ce chapitre.

Passons désormais à l'étape suivante, à savoir établir une liste plus précise des nombreuses actions réalisées par les acteurs en utilisant et observant les segments événement/décision des vues de la Figure 6.3.

6.2.5 Dernier pas vers un modèle final

Dans un premier temps, comme énoncé ci-dessus, nous allons lister de manière précise (mais pas exhaustive) les actions qui prennent place dans nos processus. Nous poursuivrons ensuite par une observation des nombreux dialogues (homme/homme ou homme/machine) qui symbolisent la transmission des informations dans le modèle BPMN.

6.2.5.1 Les activités : une vue plus précise

En partant des deux vues des ébauches de processus de la Figure 6.3, nous aurons donc deux parties, l’une sur l’unité de soins et la seconde sur la pharmacie.

Comme attendu, il est constaté que la majorité des activités sont réalisés par les infirmières dans l’unité de soins, les médecins et les pharmaciens n’étant alors utiles qu’au début de ce processus (Tableau 6.4).

Tableau 6.4 : Actions réalisées sur chaque segment du processus de l’unité de soins

Segment	Acteur(s)	Activité(s)
D1 – D2	Pharmacien	Réaliser conciliation des traitements médicamenteux et entretien thérapeutique
	Médecin	Etablir diagnostic Etablir prescription
D2 – E2	Médecin	Saisir prescription
D2 – D3	Infirmière	Ecrire prescription
D3 – D4	Infirmière – Logiciel	Renseigner la prescription dans le plan de soins papier ou informatique
D4 – D5	Infirmière	Compiler prescriptions de l’unité de soins
D4 – E3	Infirmière	Transmettre les prescriptions à la PUI
D5 – E3	Infirmière – Armoire automatisée	Contrôler les stocks Passer la commande
E3 – E9	Infirmière	Contrôler les produits reçus Ranger les produits
E9 – D5	Infirmière	Contrôler les prescriptions pour identifier d’éventuelles modifications
D5 – E10	Infirmière - Armoire automatisée	Préparer les traitements Mettre à jour les traitements
E10 – D10	Infirmière	Administer le traitement
D10 – Fin	Infirmière	Valider la prise du traitement dans le plan de soins

De la même manière, les activités dans le processus de la pharmacie sont majoritairement réalisées par le préparateur, avec l’aide de robots et de stockeurs si nécessaire. Le début du processus est quant à lui assuré par le pharmacien si besoin (Tableau 6.5).

Tableau 6.5 : Actions réalisées sur chaque segment du processus de la pharmacie

Segment	Acteur(s)	Activité(s)
D6 – E22	Pharmacien – Médecin	Analyser la prescription Dialoguer avec le Médecin
E22 – E6 D7 – D8	Pharmacien Préparateur	Valider la prescription Editer la/les demande(s) de l'unité de soins
D8 – D5	Préparateur – Stockeur mécanique – Robot de dispensation	Préparer la/les demande(s) manuellement, avec un stockeur mécanique ou à l'aide d'un robot de dispensation.
D5 – E14	Préparateur – Logiciel	Valider la fin de la préparation des demandes Regrouper les demandes de l'unité de soins
D9 – Fin	Préparateur	Transporter les produits jusqu'à l'unité de soins Ranger les produits
D9 – E17	Préparateur	Mettre en zone de transit les produits devant être livrés
E14 – E7	Préparateur	Valider la fin de préparation Transporter les produits jusqu'à l'armoire automatisée
E7 – Fin	Préparateur	Valider la livraison et le rangement Enregistrer heure de fin

Nous avons désormais des processus dont le déroulement se précise. Il est à rappeler que les événements de type « erreur » ne sont pas inclus dans les processus décrits ci-dessus, ceux-ci représentant le fonctionnement normal du CDM, et que par conséquent les nombreuses actions qui y sont liées n'apparaissent pas non plus (demande en urgence, mauvais produit, etc.).

6.2.5.2 Les interactions entre acteurs : dialogue et transmission de messages

Le fait que BPMN impose d'intégrer des acteurs de secteurs différents dans des « pools » différents afin d'obtenir un modèle fidèle à la réalité, oblige l'utilisation de « message » transitant entre ces « pools » et symbolisant un dialogue direct (téléphone, discussion de vive voix) ou indirect (email, message, etc.). Ces messages peuvent relier deux humains, un humain et une machine/logiciel ou bien entendu deux machines/logiciels. Lister ces flux va donc permettre de compléter notre modèle avec le dernier type d'informations nous manquant. Les différents dialogues existants sont :

- ❖ Médecin – Pharmacien : en cas de discussion sur la prescription ;
- ❖ Médecin/Infirmière – Logiciel : lors de la saisie de la prescription sur le logiciel ;
- ❖ Infirmière – Pharmacie : lors de l'envoi des prescriptions/ des demandes papiers à la pharmacie ;
- ❖ Armoire automatisée – Logiciel : gestion des stocks (alerte, réapprovisionnement) ;
- ❖ Logiciel – Robot de dispensation : envoi des demandes à préparer, mise à jour des stocks ;
- ❖ Pharmacien – Logiciel : correction et validation des prescriptions ;
- ❖ Préparateur – Logiciel : validation et édition des demandes ;
- ❖ Stockeur/Robot – Logiciel : gestion des stocks (alerte, réapprovisionnement).

6.2.6 Le modèle final

En reprenant tous les éléments précédemment présentés, nous avons pu construire le modèle générique des processus que nous voulions. Nous allons tenter dans ce qui suit de présenter au mieux le contenu de ce modèle mais du fait de sa taille conséquente, il est impossible de le lire de manière correcte sur un format A4. Les différents blocs mésoscopiques composant le processus seront détaillés ci-dessous en raisonnant « pool » par « pool ». Ce détail bloc par bloc sera précédé par une vue générale (Figure 6.4) du processus afin de pouvoir identifier l'emplacement des différents blocs sur le processus.

6.2.6.1 Vue globale du processus

Comme nous l'avons déjà énoncé, le processus du CDM se répartit sur quatre « pools » (Figure 6.4) chacun représentant une entité et des acteurs distincts. Afin de les identifier, chacun se trouve doté d'une couleur particulière qui le suivra tout au long de la présentation du modèle :

- ❖ Bleu : Unité de soins ;
- ❖ Rouge : Logiciel de gestion ;
- ❖ Orange : Pharmacie ;
- ❖ Vert : Logistique.

Ainsi, l'unité de soins est composée de sept blocs allant de la « Consultation et conciliation thérapeutique » à l'« administration des traitements ». Dans le logiciel de gestion de la pharmacie se trouvent les blocs concernant la gestion des prescriptions, de la production des robots de dispensation et celles des armoires automatisées. Ensuite, dans la partie pharmacie, il y a un bloc concernant la « validation des prescriptions », un pour les différents modes de préparation, un pour le « réapprovisionnement des stocks » et un dernier concernant l'« expédition des demandes ». Enfin dans la partie logistique, seul un bloc « transport au service » peut être trouvé (non présenté dans ce qui suit).

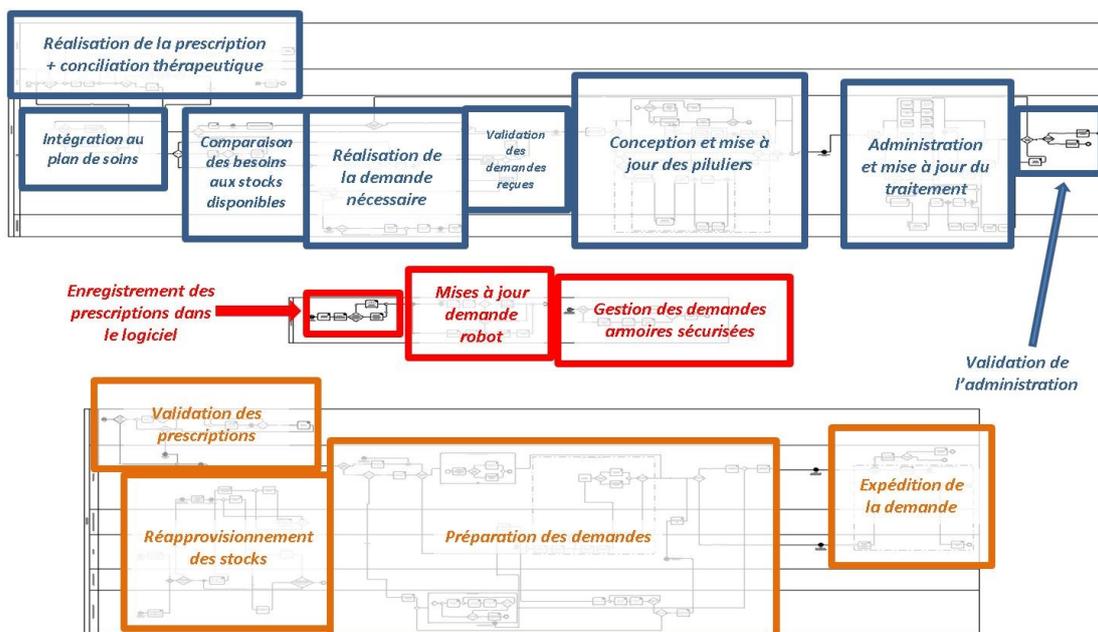


Figure 6.4 : Vue mésoscopique du processus du CDM

6.2.6.2 L'unité de soins

Dans ce « pool », le bloc qui apparaît en premier est celui concernant la consultation et la réalisation de la conciliation thérapeutique des médicaments (CTM). Ce bloc correspond au segment « E1 – D2 ». Dans celui-ci se trouvent les activités liées à la consultation (examen, choix d'un traitement) ainsi que celles concernant la CTM (entretien, mise en adéquation du traitement prescrit avec celui du patient, mise en accord avec le livret thérapeutique, etc.) (Figure 6.5).

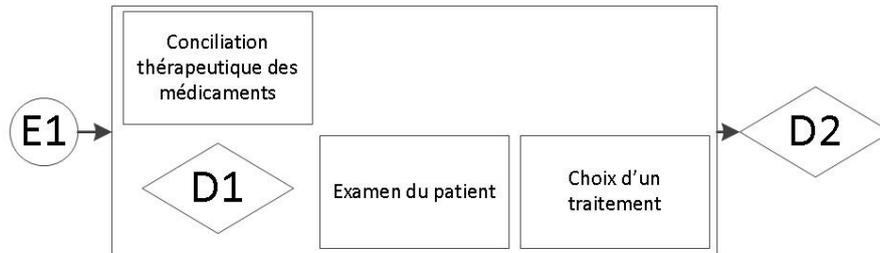


Figure 6.5 : Détails des activités sur le segment E1 – D2

Le bloc suivant (Figure 6.6) concentrera toutes les actions liées à la réalisation de la prescription quel que soit le mode de dispensation. Les activités suivantes y seront donc trouvées : renseigner et enregistrer la prescription sur le logiciel, écrire la prescription, retranscrire la prescription (sur le papier ou le logiciel). Ce bloc réunit les segments « D2 – E2 » et « D2 – D4 ».

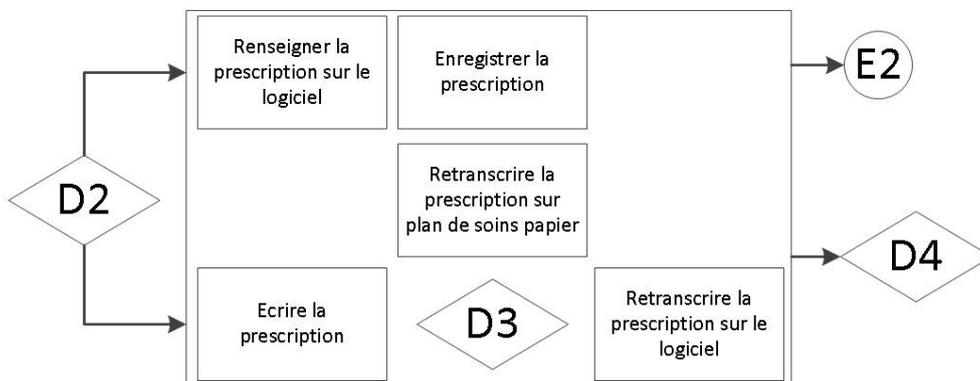


Figure 6.6 : Détails des activités sur les segments D2 – E2/D4

Le bloc numéro 3 (Figure 6.7) contiendra toutes les activités relatives à la réalisation de la commande destinée à la pharmacie et cela en fonction du type de dispensation et du type d'armoire dont dispose le service. Ainsi, on aura des activités telles que compiler les prescriptions du service, calculer les besoins en médicaments du service, transmettre les prescriptions à la pharmacie, comparer les besoins aux stocks disponibles. Les segments représentés sont ici en partie le « D4 – E3 » et « E2 – E3 ». En effet, nous avons scindé en deux ces segments afin d'avoir une partie relative au contrôle des stocks et une relative à la commande (bloc suivant).

La seconde partie des segments « D4 – E3 » et « E2 – E3 » va donc apparaître dans ce nouveau bloc. Les activités qui y ont lieu sont : transmettre la demande à la pharmacie, enregistrer l'heure de livraison, enregistrer la demande, réaliser la demande, attendre la commande, contrôler les produits reçus, ranger les produits reçus, valider la livraison (Figure 6.8).

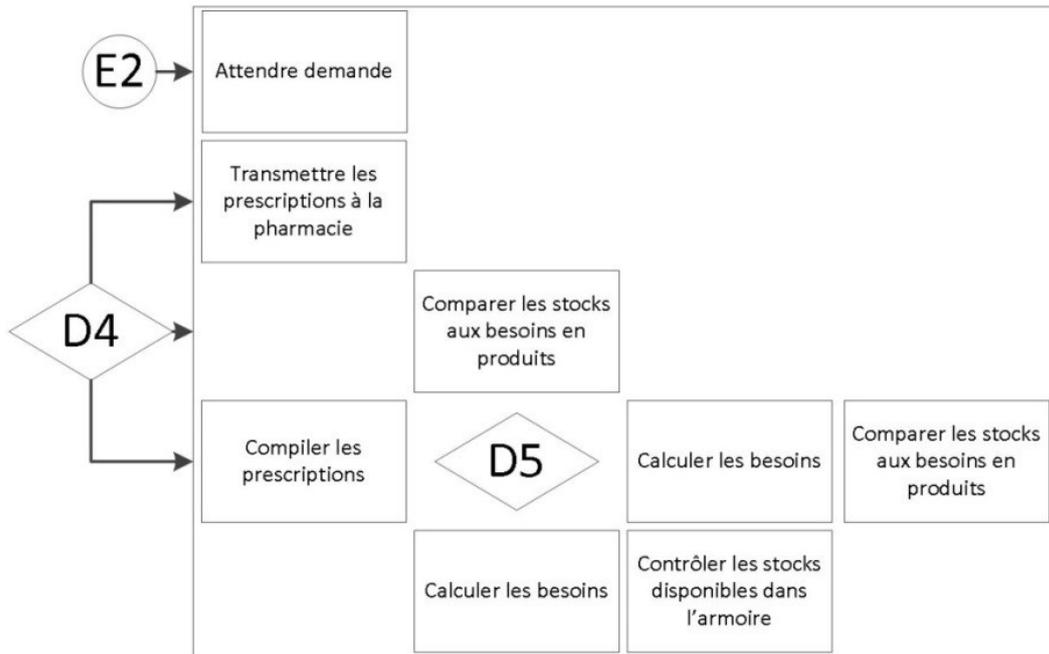


Figure 6.7 : Détails des activités sur la partie « Contrôle » des segment E2/D4 – E9

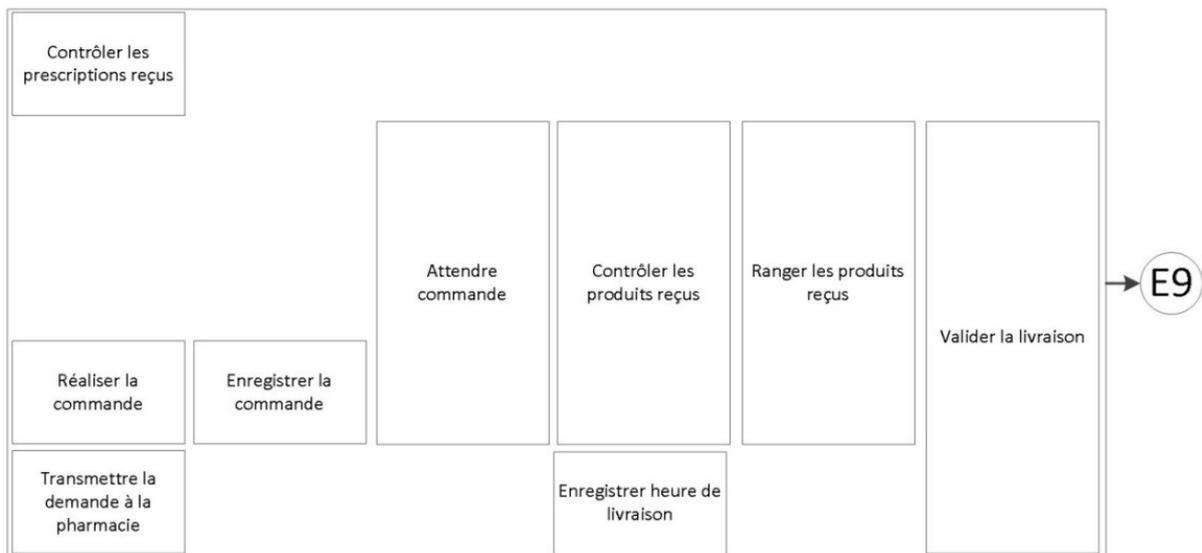


Figure 6.8 : Détails des activités sur la partie « Commande » des segment E2/D4 – E9

Une fois les produits reçus, il est possible de passer à la préparation des traitements en vue de son administration (présentée avec le bloc suivant). En plus des activités standards liées à ces actions, il y aura des activités concernant les « erreurs » pouvant avoir lieu ici. Le mode de préparation des traitements étant en partie dépendant de l'armoire utilisée, différents cas de figure peuvent donc être trouvés. Les activités sont ici : contrôler les prescriptions pour identifier d'éventuelles modifications, sélectionner un produit dans l'armoire, réintégrer un produit dans l'armoire, intégrer le médicament au traitement, prélever le médicament dans l'armoire, établir une demande urgente de produit, ranger le médicament dans l'armoire. Visuellement cela donne la Figure 6.9.

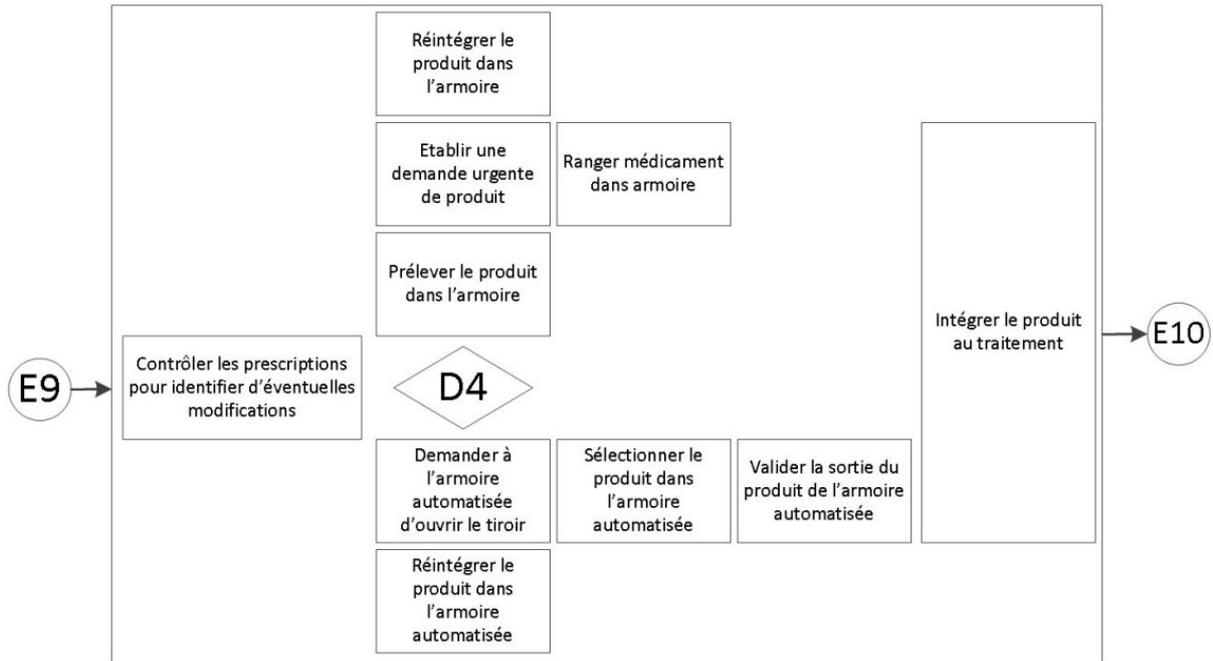


Figure 6.9 : Détails des activités sur le segment E9 – E10

Lors de cette nouvelle étape, des « erreurs » peuvent de nouveau apparaître et venir se greffer sur les actions liées à l'administration du traitement. Ainsi, les diverses activités ici présentes sont : comparer les produits à administrer/utiliser avec le plan de soins du patient, administrer le traitement, jeter le produit, réintégrer le produit dans l'armoire, établir une demande urgente de produit, prendre le produit dans l'armoire (Figure 6.10).

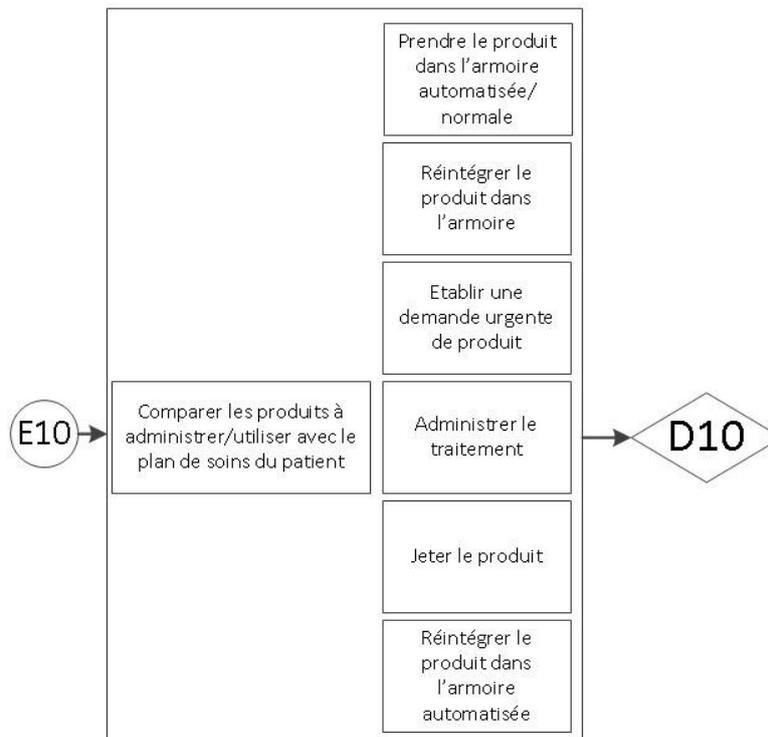


Figure 6.10 : Détails des activités sur le segment E10 – D10

Enfin le dernier bloc concernera la validation de l'administration du traitement dans le plan de soins du patient. Peu d'activités ont lieu ici, hormis écrire les actions réalisées dans le cadre de l'administration sur le plan de soins papier ou saisir les différentes actions réalisées sur le plan de soins informatique puis les enregistrer (Figure 6.11).

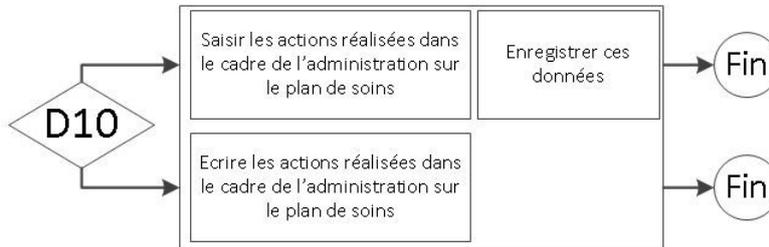


Figure 6.11 : Détails des activités sur le segment D10 – Fin

6.2.6.3 Le logiciel de gestion de la pharmacie

Ce « pool » n'est pas indispensable pour appréhender le processus du CDM, cependant nous avons souhaité quelque peu le détailler pour apporter des informations quant à la gestion des armoires automatisées et des robots. Les actions qui y sont gérées sont donc en lien direct avec ces tâches auxquelles se rajoute la gestion des prescriptions, comme nous le verrons. Ce « pool » est composé de trois blocs suivant la logique « enregistrement, validation et préparation » en ce qui concerne les prescriptions (en comprenant les robots de dispensation), et d'un bloc pour la gestion des armoires automatisées. Ainsi, dans le premier bloc se trouve les activités : enregistrer la prescription, intégrer la préparation au plan de soins et mettre à jour les besoins du service (Figure 6.12).

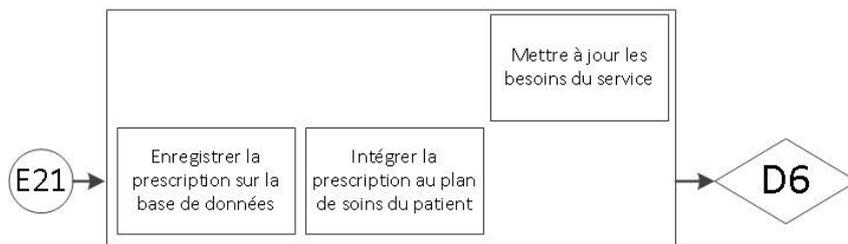


Figure 6.12 : Détails des activités sur le segment E21 – D6

S'en suit donc la possible validation de la prescription par le pharmacien passant par les activités : prévenir le pharmacien de la présence d'une nouvelle prescription, enregistrer les modifications apportées, mettre à jour le plan de soins et afficher les prescriptions dans le plan de soins (Figure 6.13).

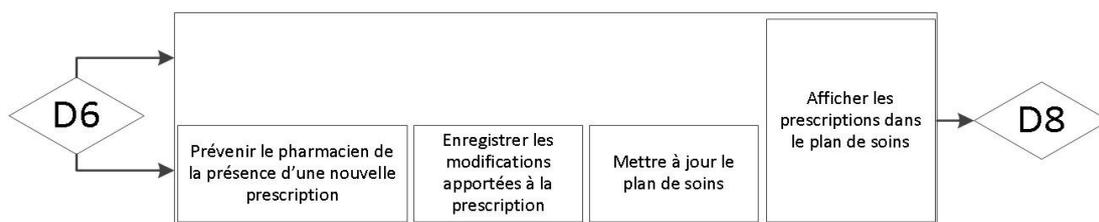


Figure 6.13 : Détails des activités sur le segment D6 – D8

Enfin, le dernier bloc de cette section concerne la préparation, ou plutôt le suivi, des prescriptions. Il sera donc retrouvé les deux pendants de cette préparation à savoir la méthode manuelle ou robotisée. Ainsi, les activités ici présentes sont : synthétiser la liste des produits à préparer à partir des prescriptions, mettre à jour le plan de cueillette du robot, envoyer le plan de cueillette au robot, enregistrer la fin de préparation de la prescription (ou de la demande liée si la préparation n'est pas individuelle). La Figure 6.14 présente cela.

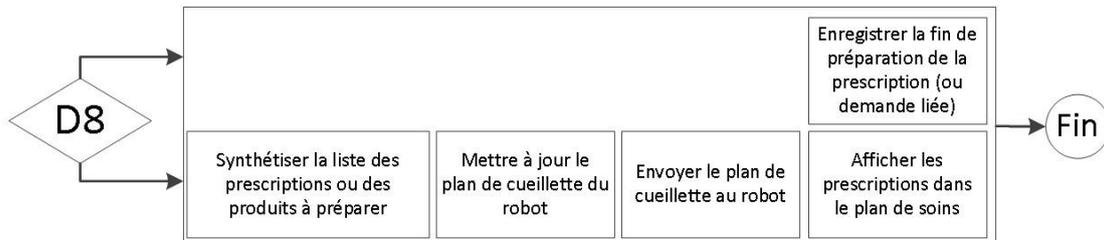


Figure 6.14 : Détails des activités sur le segment D8 – Fin

La seconde partie de ce « pool » est comme déjà dit liée à la gestion des stocks des armoires automatisées présentes dans les services. Il s'agira donc ici de gérer la commande de produits suivant deux modes : le mode supervision ou le mode gestion centralisée. Le premier sera utilisé pour une gestion réactive des stocks tandis que le second sera utilisé pour une gestion proactive. Les activités pouvant être trouvées ici sont : enregistrer la demande, émettre la demande à destination du robot de préparation, signaler la demande à la pharmacie, enregistrer l'heure d'expédition et enregistrer l'heure de livraison (Figure 6.15).

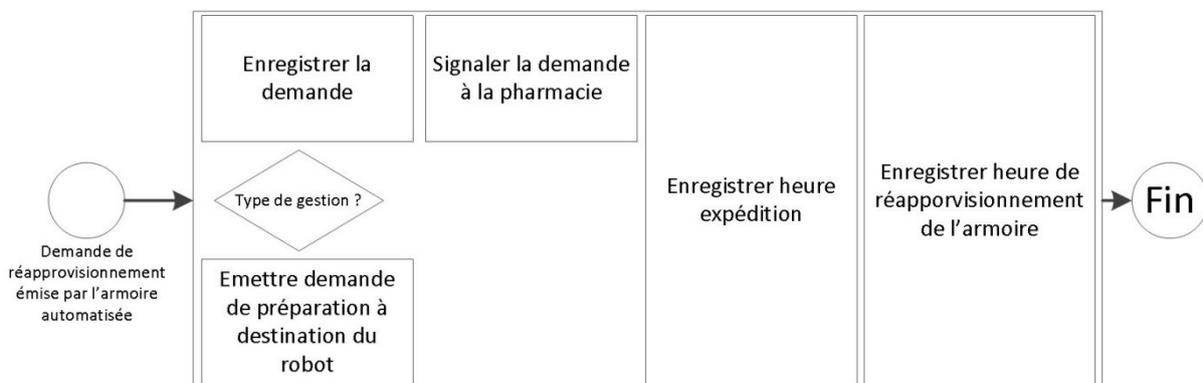


Figure 6.15 : Détails des activités sur le segment Demande réappro – Fin

6.2.6.4 La pharmacie

Dernier « pool » et pas des moindres pour nous, la pharmacie concentrera toutes les actions liées à la préparation des demandes et prescriptions, ainsi que la validation et le réapprovisionnement des stocks des divers robots de la pharmacie. Ce « pool » se verra être décomposé en quatre blocs : validation, préparation, expédition et réapprovisionnement.

Le premier d'entre eux concerne la validation par le pharmacien des prescriptions passant par les activités : analyser la prescription, demander des modifications au prescripteur et valider la prescription comme visible sur la Figure 6.16.

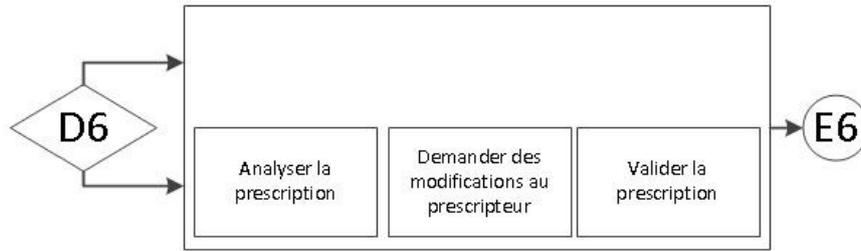


Figure 6.16 : Détails des activités sur le segment D6 – E6

Le second bloc est certainement le plus imposant de tous, celui-ci regroupant toutes les actions liées à la préparation des demandes et prescriptions. Cette préparation peut être, comme nous l’avons énoncé plusieurs fois, faite manuellement, assistée par un système automatisée ou entièrement réalisée par un robot. Cependant, bien que ce bloc soit unique sur la vue globale, nous allons finalement le scinder ici en deux afin d’être plus clair. Dans un premier temps, un bloc (a) fera référence aux divers tests de tri des demandes/prescriptions (globale, globalisée, manuelle, etc.) tandis que le bloc (b) se concentrera sur les actions de préparation.

Ainsi, dans ce premier sous bloc nous ne trouverons que deux actions mais il était nécessaire de les séparer de la suite, ces dernières se trouvant entre deux décisions D7 et D8 (Figure 6.17).

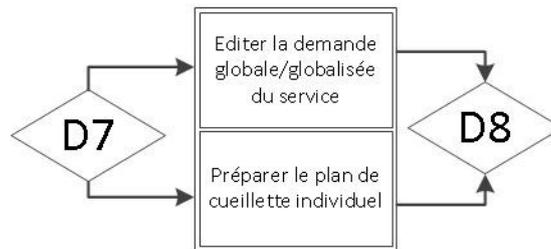


Figure 6.17 : Détails des activités sur le segment D7 – D8 (Sous bloc (a))

Nous ne citerons pas toutes les activités se trouvant dans le sous bloc (b) étant donné leur nombre mais parmi elles il y a : prendre le produit dans son emplacement de stock, afficher la demande à préparer sur le stockeur, amener la nacelle, prendre le produit, enregistrer le plan de cueillette, lire la ligne à préparer sur le plan de cueillette, mettre à jour les stocks du robot, intégrer le produit sur le chariot, etc. (Figure 6.18).

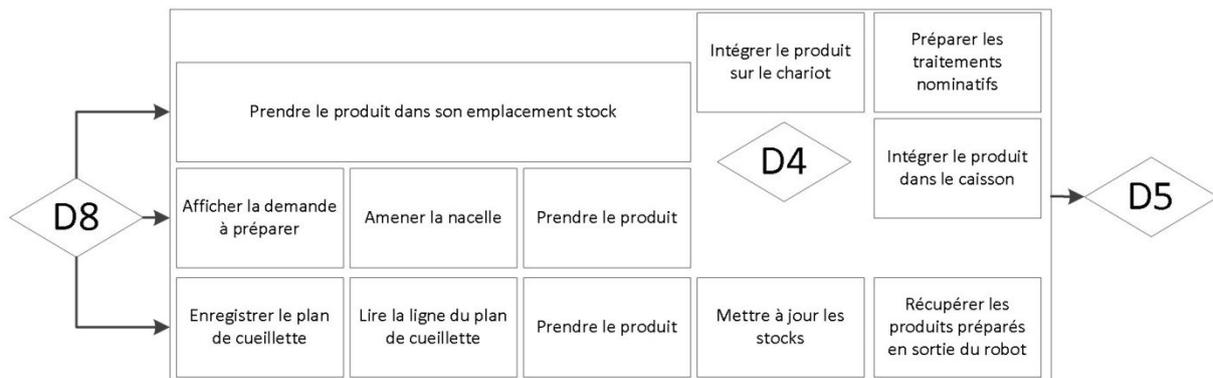


Figure 6.18 : Détails des activités sur le segment D8 – D5 (Sous bloc (b))

Le bloc suivant traite de l'expédition des demandes et prescriptions au service en considérant diverses situations. En effet, suivant le type d'armoires utilisées, le réapprovisionnement ne se gèrera pas forcément de la même manière. Pour les armoires automatisées par exemple, certaines informations transiteront entre la pharmacie et l'armoire (heure expédition, etc.). De plus, dans certains hôpitaux, les préparateurs peuvent aller livrer et ranger eux-mêmes les divers produits commandés dans les armoires des services (Figure 6.19).

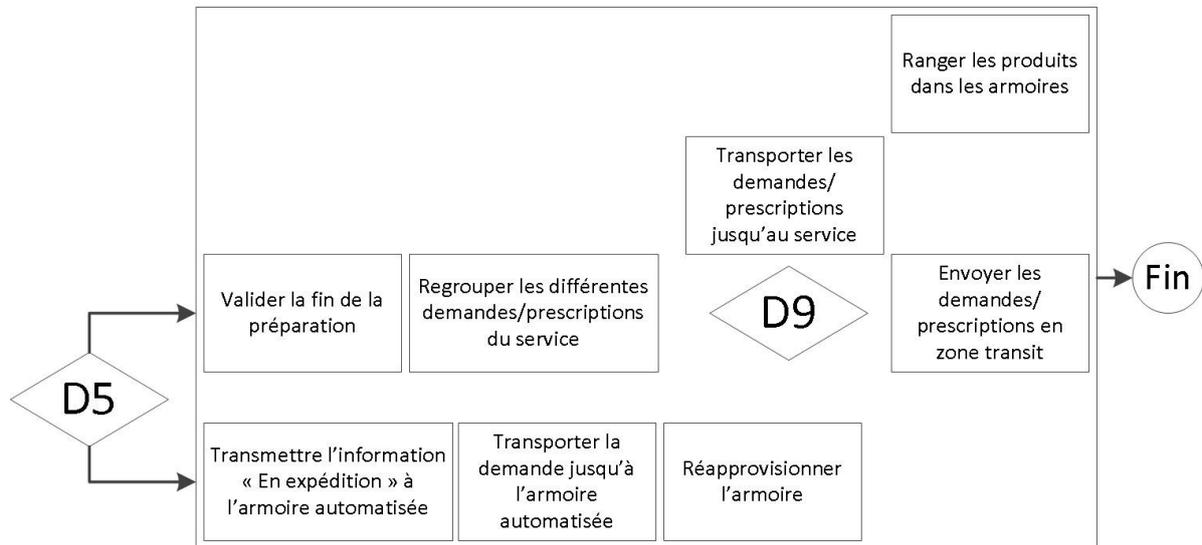


Figure 6.19 : Détails des activités sur le segment D5 – Fin

Enfin le dernier bloc composant le « pool » pharmacie est celui lié au réapprovisionnement des stocks pour les robots de dispensation et les stockeurs mécaniques. Ces actions sont réalisées par le préparateur qui peut réapprovisionner les stocks à partir d'un stock intermédiaire. Dans le cas où les stocks intermédiaires seraient insuffisants ou inexistant, le préparateur peut utiliser un robot de reconditionnement unitaire pour préparer, si besoin, les produits devant être remis en stock (Figure 6.20).

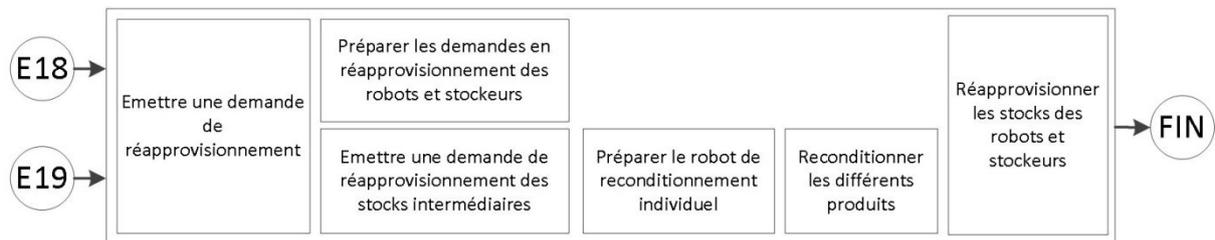


Figure 6.20 : Détails des activités sur le segment E18/E19 – Fin

Les nombreuses représentations utilisées ci-dessus pour expliciter le contenu du processus du CDM ne permettent pas d'en saisir pour autant toute la subtilité et toutes les possibilités qu'il contient. En effet, notre modélisation du processus permet de créer 135 scénarios différents de dispensation grâce aux nombreuses possibilités (les décisions du Tableau 6.2 notamment) existantes.

6.2.7 Les simplifications et hypothèses faites

Afin de concevoir le modèle présenté ci-dessus, nous avons fait certaines simplifications ou hypothèses dans le but de rendre possible sa construction. Certaines d'entre elles sont purement organisationnelles, tandis que d'autres permettent une modélisation plus simple du processus. Ainsi, parmi les simplifications/suppositions d'ordre organisationnel il y a :

- ❖ La DJIN, qui impose une prescription informatisée ;
- ❖ Une demande « Globale », qui impose un format papier pour cette demande ainsi qu'une préparation manuelle ;
- ❖ Une demande dite « Globalisée » sera elle aussi préparée manuellement mais imposera une demande soumise électroniquement à la pharmacie ;
- ❖ Seules les demandes nominatives (papier ou informatique) peuvent être préparées avec un robot de dispensation ;
- ❖ Les robots de dispensation seront toujours déchargés par un préparateur ;
- ❖ Les robots et stockeurs seront toujours réapprovisionnés par un préparateur ;
- ❖ Les armoires automatisées sont livrées et réapprovisionnées uniquement par des préparateurs ;
- ❖ Les armoires automatisées sont nécessairement interfacées avec le logiciel de gestion de la pharmacie ;
- ❖ Il y aura toujours des stocks intermédiaires permettant de réapprovisionner les robots et les stockeurs, ces stocks devant être eux aussi réapprovisionnés (notamment par l'utilisation d'un robot de reconditionnement) ;
- ❖ Les problèmes liés aux transports des demandes/prescriptions, tels que perte, destruction, choix de la zone d'entreposage dans le service, signature bon de livraison, etc., ne sont pas pris en compte ici.

En plus de ces choix organisationnels, certains choix « pratiques » ont été nécessaires :

- ❖ La DJIN sous-entend que les traitements des patients arrivent déjà tous prêts et ne nécessitent que des ajustements liés aux modifications du plan de soins patient ;
- ❖ Il n'y a pas de différence entre les patients arrivant en consultation (et qui seront hospitalisés) et les patients déjà hospitalisés subissant les diverses visites de contrôle ;
- ❖ Les modes de communication entre deux acteurs du système ne sont ici pas modélisés. Seule la transmission des informations nous importe ;
- ❖ Théoriquement aucune rupture de produit n'est possible en mode plein/vide ;
- ❖ La possibilité de choisir un mode de préparation autre que manuel a été supprimé pour les dispensations en mode globale et globalisée ;
- ❖ Si CTM il y a, celle-ci aura lieu en même temps que la consultation ou que la visite de contrôle (chose qui n'est pas toujours le cas dans la réalité) ;
- ❖ L'activité « Compiler les prescriptions » a été supprimée en mode globalisé, cela étant fait automatiquement par le logiciel.

6.2.8 Proposition d'évolution pour BPMN

Durant notre utilisation de BPMN, nous avons été confrontés à certaines difficultés lors de la conception du processus du CDM. Par conséquent, voici quelques propositions d'améliorations du langage afin de pouvoir incorporer de nouvelles variantes et possibilités dans les enchaînements des blocs BPMN, mais aussi pour rendre celui-ci plus mathématique.

6.2.8.1 Nouveau formalisme pour les messages

Afin de limiter l'encombrement des schémas lors de la présence de nombreux échanges de messages, nous proposons deux nouveaux symboles permettant de gagner en lisibilité. En effet, en gardant la symbolique de l'enveloppe propre aux messages, nous créons deux symboles, un en réception et un en émission accompagné d'un numéro permettant de s'affranchir des longues flèches en pointillés (Figure 6.21). Le défaut majeur de cette nouvelle symbolique est cependant une plus grande difficulté à trouver les couples de messages à travers le modèle si celui-ci est de taille conséquente.

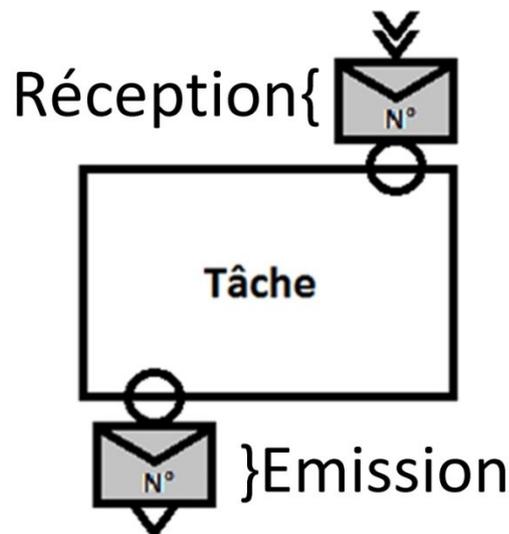


Figure 6.21 : Nouveau formalisme pour les messages

6.2.8.2 Nouvelles portes décisionnelles

Un des problèmes du langage BPMN est qu'implicitement, le symbolisme impose que la ressource utilisée, qu'elle soit humaine ou mécanique, soit disponible. Même si dans son déroulement, l'action ne peut avoir lieu sans disponibilité de l'acteur adéquat, nous proposons d'introduire deux portes permettant automatiquement de demander la disponibilité de la ressource associée (Figure 6.22). Un autre avantage de ces portes serait de permettre de questionner la disponibilité d'une ressource autre que celle dont l'activité a besoin.

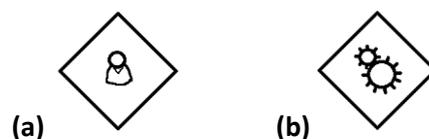


Figure 6.22 : Nouvelles portes proposées pour compléter BPMN : disponibilité d'une ressource humaine (a) et non humaine (b)

Il est à noter que le fonctionnement de ces portes peut se faire avec une porte exclusive à laquelle est associée une question portant sur la disponibilité de l'acteur lié ou bien par l'utilisation d'une porte exclusive basée sur des évènements. L'utilité première de ces deux nouvelles portes serait donc surtout de simplifier le contenu du processus en introduisant deux symboles facilement reconnaissables et d'éviter l'utilisation de porte compliquée. Ainsi, la première permet de demander si une ressource humaine est disponible. La seconde concerne quant à elle une ressource non-humaine.

De même, BPMN ne prend pas en compte l'aspect mathématique inhérent aux systèmes de production, ou en général aux systèmes comportant de nombreuses sources de variabilité. L'idée est donc d'introduire des portes basées non plus sur une réflexion logique mais mathématique. Parmi les portes pouvant être créées, les deux qui pourraient être mises en avant sont la porte basée sur une décision portant sur des pourcentages (Figure 6.23 (a)) ou celle basée sur une loi mathématique (Figure 6.23 (b)).

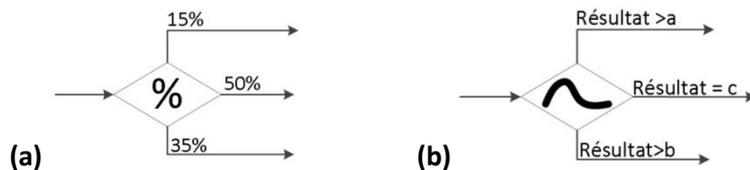


Figure 6.23 : Portes « mathématiques » proposées pour compléter BPMN

6.2.8.3 Prise en compte d'acteurs multiples pour une ou plusieurs tâches

L'une des grosses lacunes de BPMN est la gestion des activités incluant plusieurs acteurs. Il y a bien les diagrammes de chorégraphie permettant de traduire les échanges de messages entre différents acteurs et cela avec une symbolique précise (Figure 6.24), mais il n'existe malheureusement pas son pendant pour les diagrammes de collaboration.

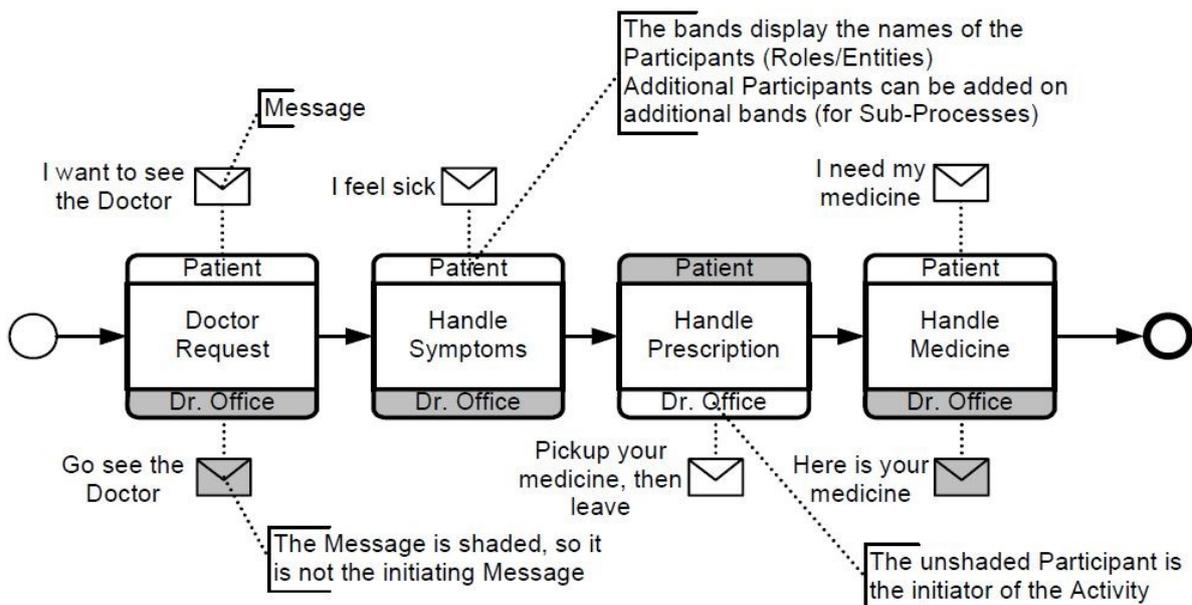


Figure 6.24 : Exemple de diagramme de chorégraphie avec le rôle des différents messages

Deux propositions peuvent être formulées pour apporter une solution à ce manque. La première permet de ne pas compliquer les « pools » en précisant sous les activités les autres acteurs intervenant dans celle-ci à la manière des diagrammes de chorégraphie, mais uniquement en dessous de l'activité (Figure 6.25).

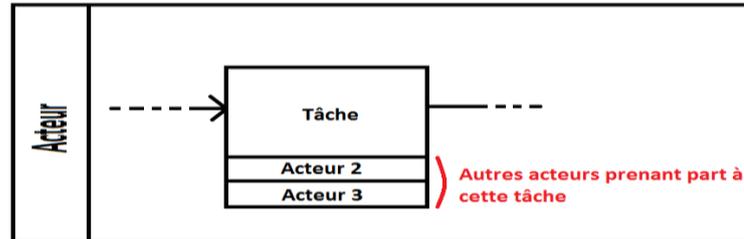


Figure 6.25 : Première proposition pour lier plusieurs acteurs à une même activité

La seconde proposition (Figure 6.26) permet quant à elle de définir une zone commune entre deux ou plusieurs acteurs, où les activités y étant situées seront forcément réalisées par tous les acteurs liés.

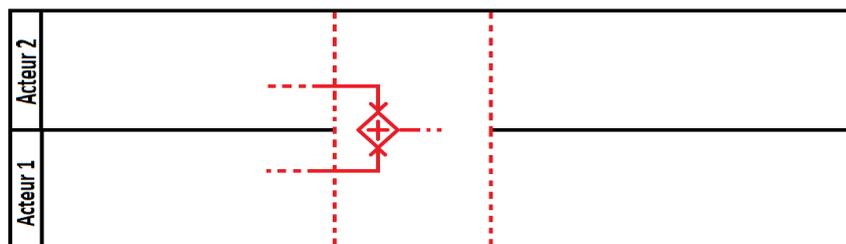


Figure 6.26 : Seconde proposition d'amélioration

Globalement, la première solution permettra de préciser ponctuellement des activités réalisées par divers acteurs ou bien des activités incluant des acteurs n'étant pas côte à côte dans le diagramme, ce qui rends impossible l'utilisation de zones communes à la manière de la seconde proposition, permettant de définir des pans entier de processus en commun.

6.2.8.4 Vue macroscopique plus synthétique

L'un des problèmes de BPMN est que malgré la possibilité de représenter un processus d'un niveau macroscopique à un niveau microscopique, le formalisme ne change pas entre les deux. Ainsi, pour représenter un processus, quel qu'il soit, aucune différenciation visuelle ne sera faite entre les deux types de vue. De plus, si le modélisateur souhaite déjà intégrer tous les acteurs au niveau macroscopique, il lui sera impossible de garder une représentation macroscopique de son processus, toutes les activités des différents acteurs devant alors apparaître, ce qui mènera automatiquement à la création d'un processus mésoscopique dans le meilleur des cas. Afin de contourner cette problématique, nous proposons un formalisme plus général, volontairement plus abstrait mais permettant de comprendre en un bref instant le déroulement d'un processus (Figure 6.27). Celui-ci s'appuie en partie sur l'idée de la Figure 6.25 en intégrant plusieurs acteurs dans le pictogramme, ainsi qu'une évolution de sa forme afin de différencier une vue macroscopique (Figure 6.28) et microscopique.

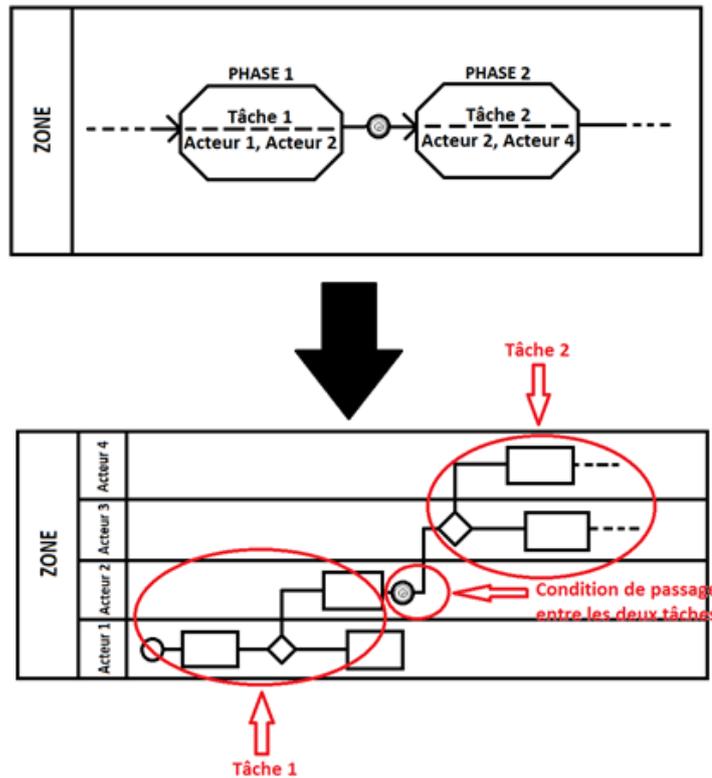


Figure 6.27 : Formalisme macroscopique proposé et sa traduction en microscopique

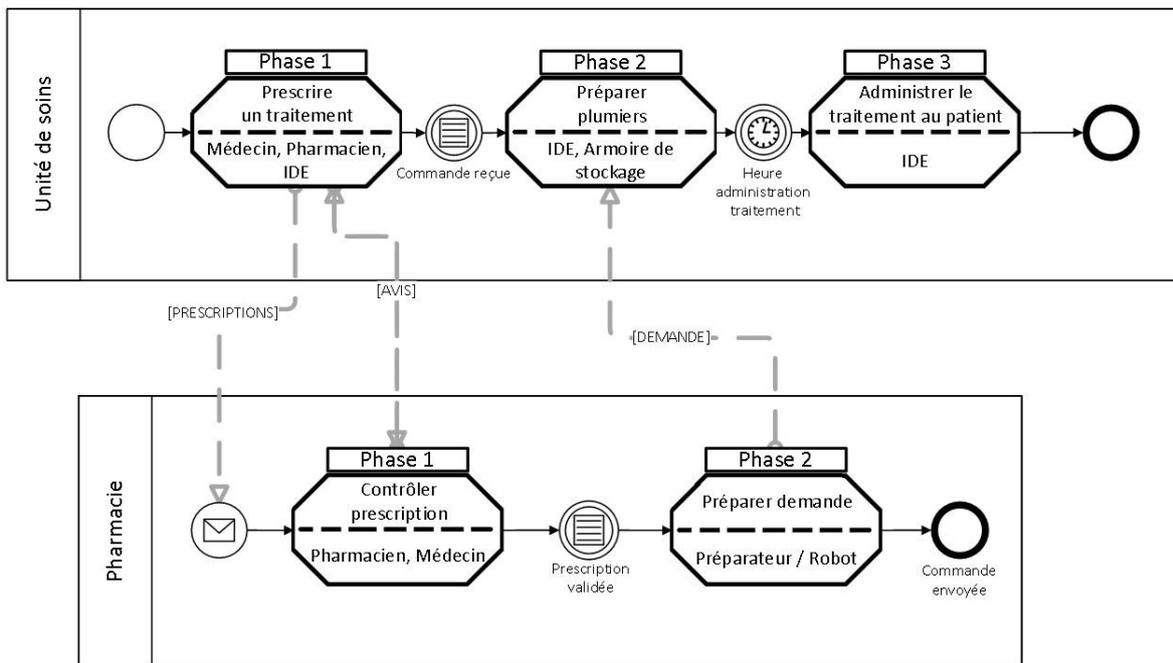


Figure 6.28 : Exemple d'application du formalisme macroscopique

6.2.8.5 Echelle de temps

La dernière des évolutions que nous pouvons proposer, et qui tient ici plus un rôle informatif, est l'incorporation d'une échelle de temps en haut du modèle des processus pour permettre une meilleure visualisation du déroulement des processus et donc une meilleure compréhension des

impératifs et des dates/heures articulant le processus. Il serait alors possible de structurer les processus suivant cet axe. Malheureusement, cela aurait le défaut de faire exploser la taille du modèle et compliquerait la modélisation. L'utilisation d'une telle échelle pourrait donc se faire au niveau macroscopique/mésoscopique permettant ainsi de limiter l'expansion du modèle.

6.3 DIAGRAMME GENERIQUE DE CLASSES UML

Comme énoncé dans la partie concernant la méthodologie ASCI, celle-ci requiert la création de trois sous-systèmes permettant de comprendre quels sont les entités/acteurs qui agissent/parcourent notre système et quelles sont leurs interactions basiques. Afin de créer ce diagramme, nous utiliserons le diagramme de classes UML permettant de visualiser pour chaque classe (correspondant ici aux diverses entités composant notre système) leurs caractéristiques ainsi que les activités principales qu'elles réaliseront.

Cependant, contrairement au modèle générique des processus que nous avons entièrement conçu, ce diagramme générique de classes s'appuiera sur les travaux de [RODIER 10] et [HUET 11]. En effet, ces deux auteurs ont réalisé à travers leurs recherches des diagrammes très proches de ce que nous souhaitons construire (notamment [HUET 11]). Ainsi, afin de ne pas recréer à partir de zéro ce diagramme, nous allons réutiliser ce qui a été fait et le compléter/modifier quand cela sera nécessaire.

6.3.1 Le diagramme SSPDL de [RODIER 10]

Ce premier diagramme de classes (Figure 6.29) est plus générique à l'hôpital que ce que nous recherchons. En effet, il est décrit à travers celui-ci l'évolution d'un patient à travers l'hôpital et donc incorpore de nombreuses classes que nous ne considérerons pas dans notre cas. Cependant, ce diagramme est très complet et permet d'appréhender parfaitement la complexité de gestion du parcours d'un patient dans un hôpital en visualisant les nombreuses interactions que celui-ci a avec son environnement.

Le SSD composant ce diagramme s'articule autour de trois notions de bases régissant le parcours du patient dans l'hôpital à savoir l'« Organe de décision », la « Règle » et la « Contrainte ». Ces trois notions étant régi par le « Centre de décision ». Le SSL s'articule en toute logique autour des flux composant le système (Physique, Financier et Informationnel) et plus particulièrement autour des objets de flux (Patient, Activité, Produit, Matière première, etc.). Enfin, le SSP est composé des entités telles que « Ressource », « Zone », « Service », « Stock » permettant de modéliser la structure physique du système.

6.3.2 Le diagramme SSPDL de [HUET 11]

Le diagramme réalisé par cet auteur est plus proche de ce que nous voulons réaliser. Rien d'étonnant à cela, ce dernier ayant lui aussi travaillé sur une modélisation du CDM grâce à la méthodologie ASCI. Par conséquent, notre diagramme générique de classes sera largement inspiré de ses travaux, les différences majeures étant une conséquence de l'orientation de l'étude. En effet, [HUET 11] étant centré sur l'interprétation holonique du système en prenant comme point de départ les systèmes industriels, certaines classes ne seront plus identifiées sous le même nom, et

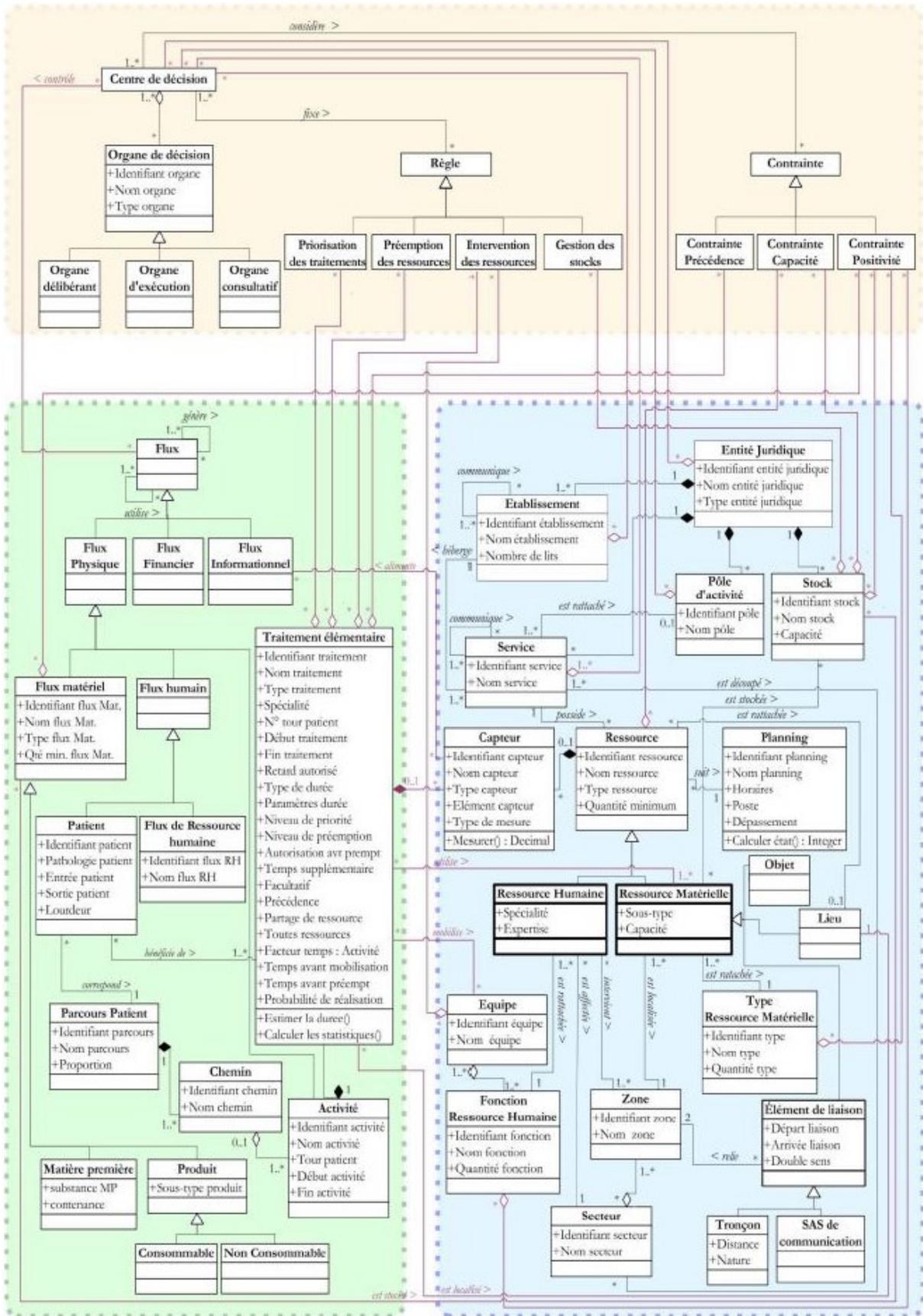


Figure 6.29 : Diagramme des classes conçu par [RODIER 10]

leurs caractéristiques varieront quelque peu. Cela s'explique parfaitement par le fait que notre diagramme n'a pas vocation à évoquer l'interprétation industrielle du CDM, mais à présenter les éléments pouvant constituer le CDM d'un point de vue pharmacien et donc moins technique.

Contrairement à [RODIER 10] qui se concentre, dans son SSD, sur les composantes d'une décision à prendre, [HUET 11] se focalise sur le type de décision (stratégique, tactique ou opérationnelle) et cela suivant deux niveaux (Figure 6.30). Ainsi, les décisions centralisées sont prises par les pharmaciens, tandis que les décisions décentralisées sont prises par les entités sur le terrain.

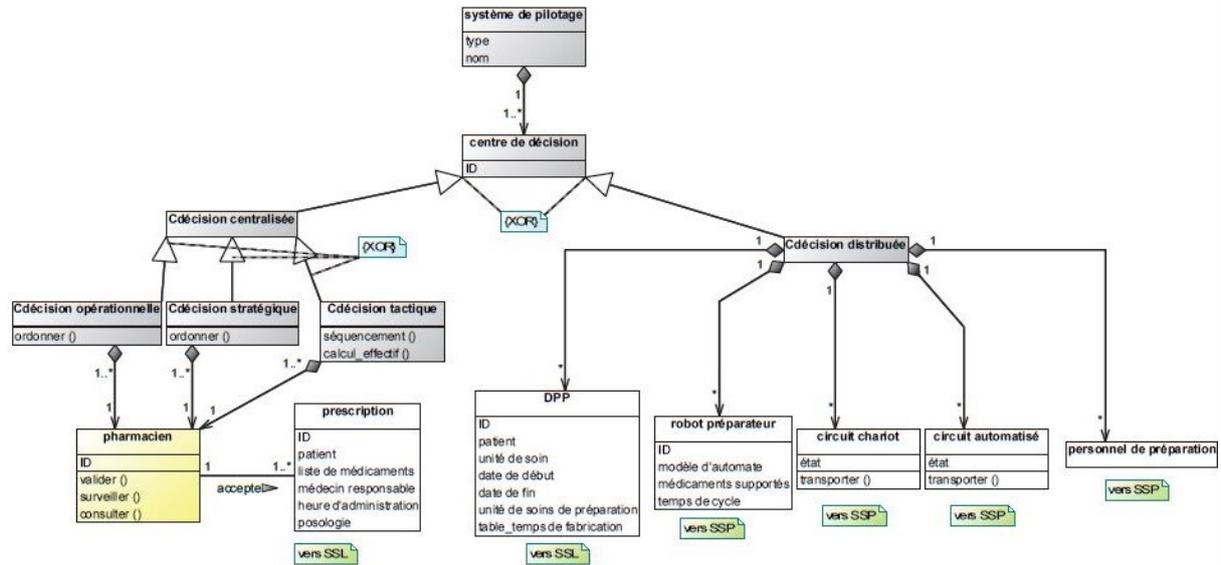


Figure 6.30 : Sous-système Décisionnel conçu par [HUET 11]

Le SSL est ici centré sur les entités traitées par le système et qui dans notre cas sont majoritairement des demandes de services (DPS), prescriptions (DPP) de traitement et des produits pharmaceutiques (Figure 6.31). Comme avec les travaux de [RODIER 10], le SSL est ici concentré sur le flux physique, mais celui-ci ne concerne non plus le patient mais son traitement.

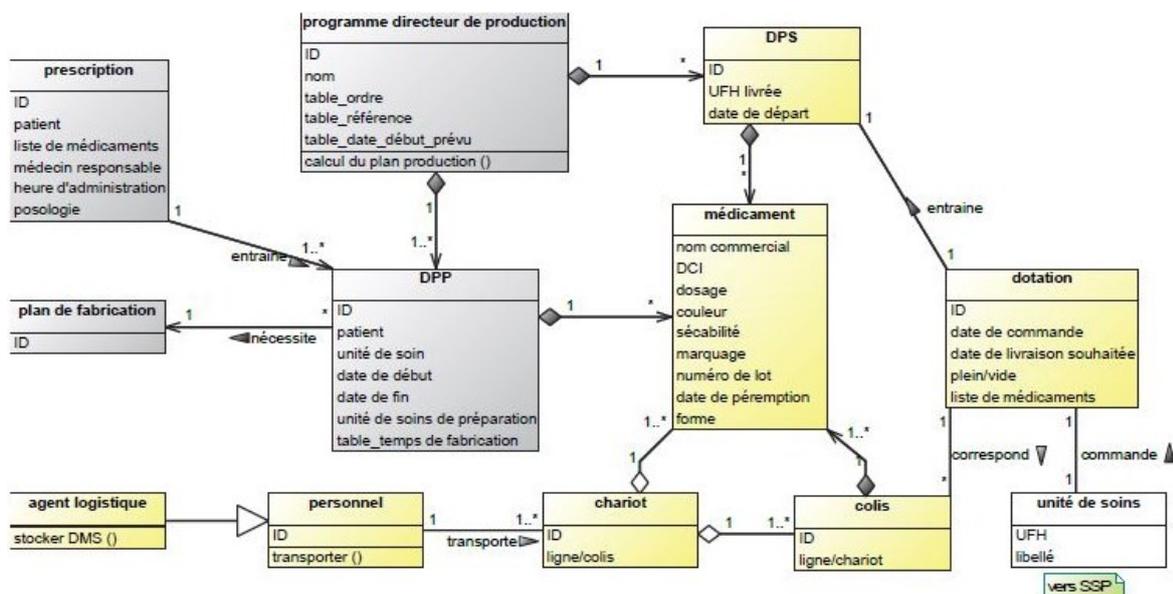


Figure 6.31 : Sous-système Logique conçu par [HUET 11]

Enfin dans le SSP, on retrouve comme nous l’avons énoncé des termes qui sont proches du monde industriel tels que « Ligne d’assemblage », « Feeder », mais encore « Poste de travail ». Tout comme le SSL, le SSP de [HUET 11] est très différent dans le contenu de celui de [RODIER 10]. En effet, comme la logique le veut, celui de [RODIER 10] est centré sur les équipes soignantes qui s’occupent du patient tandis que celui de [HUET 11] comprendra les nombreuses entités aidant à la préparation du traitement des patients. Parmi celles-ci il y a le « Médecin », le « Pharmacien », le « Magasinier », le « Robot de préparation », etc.

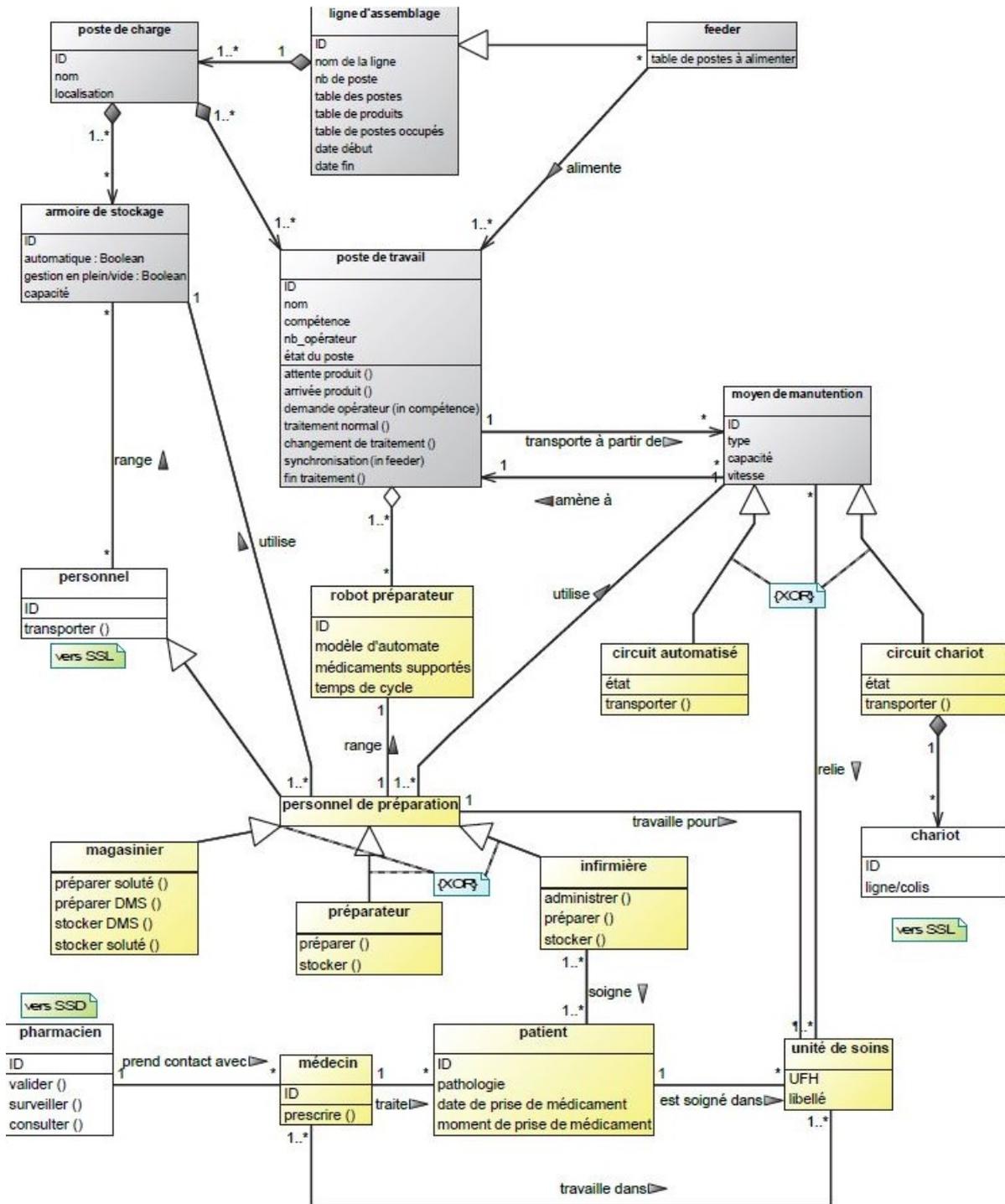


Figure 6.32 : Sous-système Physique conçu par [HUET 11]

6.3.3 Notre diagramme des classes

En reprenant le diagramme des classes issu des Figures 6.30, 6.31 et 6.32, il ne reste plus qu'à apporter certaines modifications pour obtenir celui que nous voulons. Parmi les modifications majeures il y a :

- ❖ Plus d'informations sur les caractéristiques et les activités des différentes entités ;
- ❖ Un changement de nom pour certaines classes (« Armoire de stockage » transformée en « Système de stockage ») ;
- ❖ Une restructuration du SSL qui s'articulera autour de « Demande préparée » ;
- ❖ Le « Pharmacien » qui est transféré du SSD au SSP, celui-ci pouvant être désormais en contact direct avec le patient ;
- ❖ Plus de détails dans les moyens de transports.

Pourquoi faire de telles modifications ?

En ce qui concerne le changement de nom des classes, cela se justifie par le besoin de clarté que ces classes doivent avoir afin que les pharmaciens identifient bien ce qui est attendu. Dans d'autres cas, il s'agit de rendre plus générique la définition de la classe afin de pouvoir y intégrer de nouveaux éléments (notamment pour le « Système de stockage »).

Le SSL a été restructuré afin d'être en adéquation avec notre CDM, et les deux moyens de transport rajoutés permettent de prendre en considération de nouvelles situations.

Enfin, il a été nécessaire de compléter les différentes classes afin d'avoir à disposition toutes les informations qui sont utiles à la compréhension du fonctionnement de toutes les entités, à leur caractérisation. Les nombreuses informations complémentaires apportées serviront notamment dans le cadre de la simulation plus tard. En effet, le but final de ce diagramme est de faire connaître à l'utilisateur, au minimum toutes les caractéristiques permettant de simuler cette entité dans notre modèle de simulation.

Au final, nous obtenons la Figure 6.33 où le SSL est en bleu, le SSD en rose et enfin le SSP en vert. Voyons rapidement quelles sont les principales entités de chaque sous-système :

- ❖ SSP : zone de production, moyen de manutention, poste de travail, robot, personnel de préparation (magasinier, préparateur, infirmière), personnel hospitalier, moyen de transport (roll, chariot, chariot motorisé), unité de soins, patient, médecin, valideur (pharmacien, interne).
- ❖ SSL : logiciel de gestion, produit pharmaceutique, colis, dotation, demande préparée (demande nominative, demande service), prescription, plan de cueillette.
- ❖ SSD : centre de décision, centre de décision décentralisée, décision tactique, décision organisationnelle, décision stratégique.

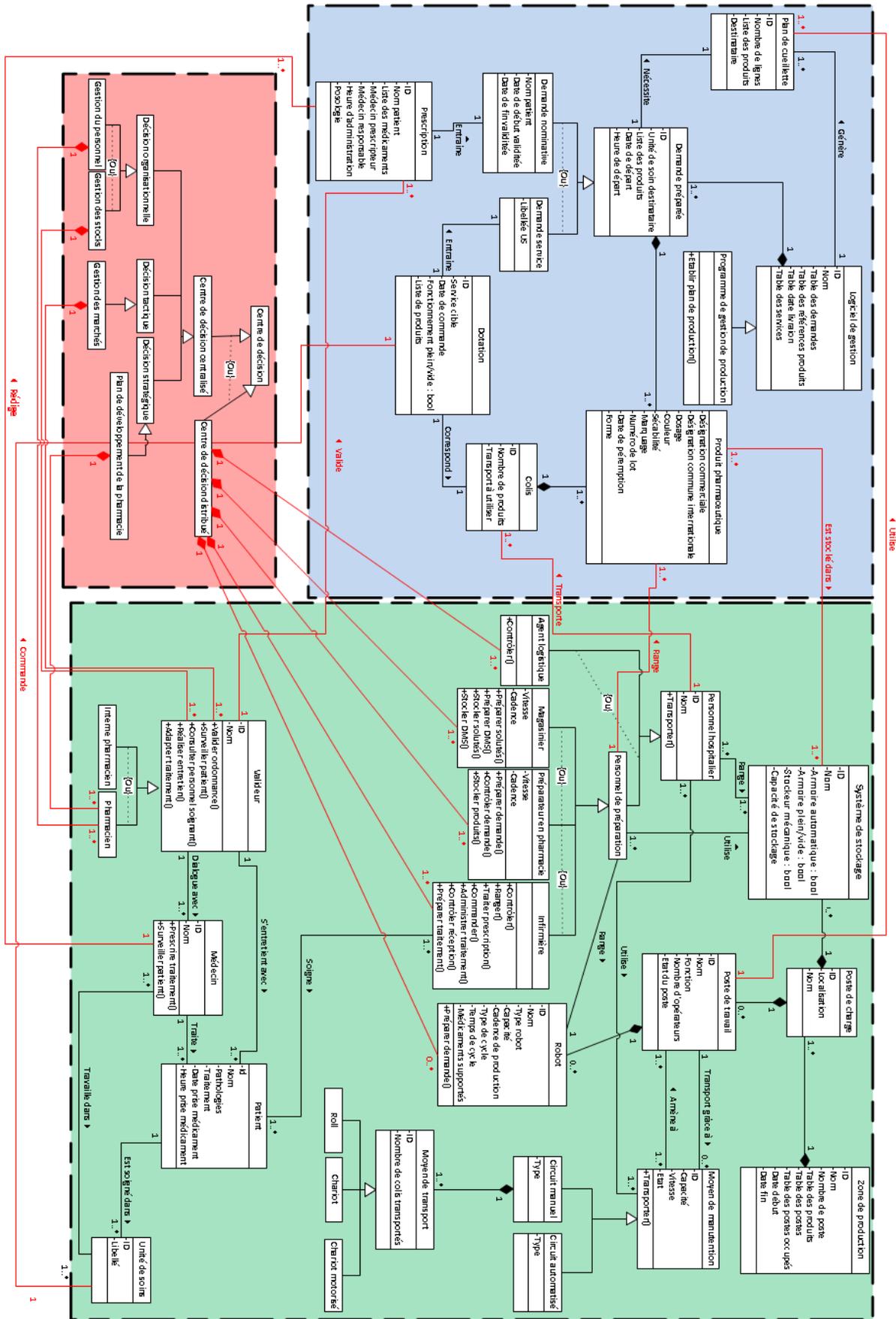


Figure 6.33 : Diagramme des classes générique conçu pour notre démarche

6.4 CONCLUSION

L'utilisation de la méthodologie ASCI pour orienter et diriger nos travaux a, au final, nécessité la construction d'un MGC. Etant tous deux une composante de ce MGC, le modèle générique des processus sous BPMN et le diagramme générique de classes UML ont fait l'objet d'une présentation détaillée dans ce chapitre.

Le langage BPMN et son utilisation étant la plus grande inconnue dans ces travaux (comparativement au diagramme de classes UML, bien plus usité), nous avons détaillé dans ce chapitre comment obtenir un modèle générique des processus sous BPMN en partant de systèmes réels. Pour commencer, les grandes lignes des processus du CDM ont été obtenues avec le Macro-processus fourni par [HAS 11]. Ce Macro-processus donnant les activités majeures réalisées sur le CDM, est complété par une liste d'acteurs et de leurs fonctions, que nous avons par la suite classés par « pool » (US, pharmacie, etc.) dans le but d'identifier leur zone d'action. En observant avec attention ces activités, la question du niveau de granularité à adopter s'est alors fait sentir. En effet, les niveaux macroscopique et microscopique n'étant pas adaptés à une représentation générique, nous avons alors penché pour un niveau mésoscopique capable selon les besoins d'osciller tantôt vers le macro, tantôt vers le micro.

La seconde étape du développement du modèle BPMN a été d'identifier les points-clés de nos processus et cela sur deux niveaux : un évènementiel et un décisionnel. Le tri et la classification de ces points clés, nous a permis d'étudier et de comprendre l'impact de ces derniers sur les activités des acteurs et sur le « rythme » d'évolution des processus, rendant ainsi possible la construction d'une ébauche des processus du CDM.

Grâce à cette ébauche, une étude plus précise des activités ayant lieu sur les segments reliant deux points clés (évènement/évènement, évènement/décision, décision/décision), a alors été menée dans le but d'obtenir une liste d'activités plus précises et mieux découpées. Dès lors, la conception du modèle final était réalisable, ce dernier étant présenté sous forme de blocs mésoscopiques par « pool », chaque bloc représentant les activités présentes sur un segment entre deux points-clés. Afin que cette conception ait lieu, nous avons dû faire certaines simplifications, hypothèses pour ne pas en entraver la bonne marche. Celles-ci sont diverses et variées, et parmi elles se trouvent l'obligation d'interfaçage des armoires informatisées avec le logiciel de la pharmacie, la nécessité de faire charger et décharger le robot de dispensation par un préparateur, etc. Cette conception nous a aussi donné l'opportunité de voir que BPMN souffrait de certaines lacunes, nous permettant dès lors de proposer certaines améliorations apportant plus de confort dans la lecture et l'utilisation de BPMN (par exemple : nouveau formalisme des messages, nouvelles portes, etc.). Cependant, nous n'utiliserons pas ces améliorations afin de garder le formalisme BPMN 2.0 international et ainsi pouvoir utiliser les logiciels existants sur le marché.

Le modèle BPMN étant obtenu, la construction du diagramme générique de classes UML a alors pu démarrer. Le développement de ce dernier a été plus succinct, étant donné que des travaux conséquents, [HUET 11] et [RODIER 10] avaient déjà érigé des modèles du CDM ou de la prise en charge d'un patient à l'hôpital. Ainsi, après avoir présenté les modèles SSPDL de ces deux auteurs, nous avons conçu notre propre modèle en nous basant sur leurs travaux, surtout sur ceux de [HUET 11], auxquels des compléments et certaines modifications ont été apportées.

Le MGC étant par conséquent terminé, il est désormais temps de se pencher sur la conception du modèle d'action, qui dans notre cas sera un modèle de simulation développé sous le logiciel SIMIO. Ainsi après un premier chapitre, le 7, présentant les bases et le fonctionnement de ce logiciel, un second, le 8, abordera en précision la construction de notre modèle de simulation.

VIII

BASE DE LA CONCEPTION ET DU DÉVELOPPEMENT DU MODÈLE DE SIMULATION

RESUME DU CHAPITRE 7

Bien que la modélisation des processus métiers soit désormais largement utilisée et bien considérée par les organisations quelles qu'elles soient, il existe toujours un fort taux d'échec dans leur implémentation, notamment à cause du manque d'outils permettant d'évaluer les effets des nouvelles solutions proposées sur le système avant leur implantation.

Ainsi, et en vue de créer l'outil permettant de répondre à cette problématique, l'objet de ce chapitre sera de présenter le logiciel de simulation permettant de créer cet outil mais aussi de détailler comment ce dernier a été conçu en se basant sur le modèle générique des processus développé dans le chapitre précédent.

SOMMAIRE DU CHAPITRE

7.1	INTRODUCTION	187
7.2	CRITERES DE PERFORMANCE ET CURSEURS	187
7.2.1	Evaluation de la performance	187
7.2.2	Leviers de simulation.....	189
7.3	SIMIO.....	190
7.3.1	Qu'est-ce que SIMIO ?.....	190
7.3.2	Les objets.....	197
7.3.3	Les processus dans SIMIO	209
7.3.4	Un exemple simple d'application de SIMIO	212
7.4	NOTRE APPROCHE POUR PASSER DE BPMN A SIMIO	213
7.4.1	Comment traduire une activité	214
7.4.2	Comment traduire les évènements.....	217
7.4.3	Comment traduire les portes décisionnelles.....	220
7.4.4	Comment traduire les acteurs.....	221
7.5	CONCLUSION	222

7.1 INTRODUCTION

Des trois outils nécessaires à l'exécution de notre démarche, deux d'entre eux sont déjà à notre disposition, ne laissant plus que le troisième et dernier d'entre eux à concevoir, mais qui n'en est pas le moins complexe. Par conséquent, pour construire ce dernier outil, à savoir le modèle générique de simulation, nous utiliserons le logiciel de simulation SIMIO, dont il va être nécessaire de détailler le contenu et le fonctionnement. Ce dernier outil ne sortira pas pour autant de nulle part, le contenu et la structure de ce dernier étant basé sur les nombreuses informations qui ont été récoltées et utiles à la construction des deux premiers outils.

Bien que la conception d'un modèle de simulation soit très souvent une tâche problématique, les utilisateurs finaux ne portent que peu d'intérêt au comment de la conception, faisant de cette tâche une « boîte noire » dont ils ne se préoccupent pas. Donner envie aux futurs utilisateurs de se pencher sur le comment est donc l'un des objectifs de ce chapitre et du suivant.

A un niveau conceptuel, la majorité des outils de simulation à événements discrets fonctionne d'une manière similaire, à savoir des entités/transactions/parties transitant dans une série de files d'attentes/listes/sets/buffers requérant ou libérant des ressources/machines/travailleurs, eux-mêmes pouvant voyager dans le modèle. Celui-ci est entièrement piloté par une séquence d'évènements discrets qui se déclenche quand un(e) activité/processus/délai/cycle est complété, et le mouvement de l'entité résultant de ce changement d'état à lui-même une conséquence sur les futurs évènements à venir. Nous allons donc dans ce qui va suivre présenter tous ces éléments qui composent les outils de simulation.

Précédée par un bref descriptif des divers leviers de simulation désirés par le CHU, cette seconde partie explique de quelles fonctions le logiciel SIMIO est pourvu, avant de présenter en détails les objets – fixe, nœud, agent – et les processus pouvant constituer un modèle de simulation. Afin d'illustrer ce qui a été présenté, un exemple d'application conclura cette partie.

Pour terminer, nous aborderons dans une dernière section, comment et sous quelles règles il est possible de passer d'un modèle BPMN à un modèle sous SIMIO. Le but est donc de fournir des règles de passage du premier outil vers le dernier, ces lois n'étant pas absolues mais permettant au moins de donner des pistes de solutions pour cette traduction.

7.2 CRITERES DE PERFORMANCE ET CURSEURS

7.2.1 Evaluation de la performance

Afin de pouvoir évaluer le comportement d'un système face à des modifications qui lui sont apportées (théoriquement ou physiquement), l'utilisation de valeurs qui seront ici nommées critères de performances, se révèle un impératif. Ces critères de performances se révéleront nécessaires pour la prise de décisions, et ils peuvent porter sur de nombreux aspects (coût, temps, efficacité, rendement, etc.). Parfois antagonistes, ils symbolisent parfaitement la réalité des systèmes en mettant en lumière des contradictions propres à la prise de décisions.

La détermination des critères de performances peut se révéler un problème en soi et ces critères peuvent être très variables d'une structure à une autre. En effet, évaluer ces critères sur des

systèmes complexes impose généralement la construction de modèle (mathématique, de simulation) dédiés à ce ou ces systèmes. Ainsi, à travers notre modèle de simulation et dans le but de ne pas omettre de critères importants, nous tenterons de retranscrire autant de données que possible aux utilisateurs de notre outil, en privilégiant les données souhaitées par les décideurs. Bien entendu, un certain nombre de critères basiques peuvent déjà être mis en évidence:

- ❖ La charge de travail du personnel ;
- ❖ Le rendement en termes de demandes préparées ;
- ❖ Les temps de production, d'analyse ;
- ❖ Les retards ;
- ❖ La répartition de la production ;
- ❖ Etc.

Ainsi, il est possible de visualiser dans l'interface de sortie les informations suivantes (Tableau 7.1) :

Tableau 7.1 : Liste des informations présentes dans l'interface de sortie

En rapport avec	Information
Préparateur	Charge de travail individuelle Nombre de demandes préparées individuellement Durée d'inactivité Coût sur la période de travail
Valideur	Nombre de demandes validées individuellement Coût sur la période de travail
Sources	Nombre de demandes de chaque type générées Nombre moyen de produits par demande Nombre moyen d'éléments par demande Temps de retard moyen par demande
Pôle de production (Frigo, Zone picking, etc.)	Temps de présence d'une demande dans chaque pôle : - Moyen - Minimum - Maximum Nombre de demandes y passant
Transport	Temps de transport jusqu'au service : - Moyen - Minimum - Maximum
Demande	Nombre de demandes ayant des sous-demandes Temps de préparation à la pharmacie : - Moyen - Minimum - Maximum Pour chaque demande : - Nombre de produits - Nombre d'éléments - Préparateur ou robot affecté - Temps de préparation - Service de destination

7.2.2 Leviers de simulation

A la demande de l'hôpital un certain nombre de leviers, de curseurs devront être intégrés au modèle de simulation. Bien que ces demandes soient issues du CHU de Clermont-Ferrand, il s'est révélé que celles-ci peuvent convenir à toutes les structures hospitalières.

Le premier levier souhaité est de donner la possibilité à l'hôpital de réfléchir sur la préparation de demandes, de prescriptions pour des structures extérieures. La seule réelle contrainte associée à cette demande est que les structures prises en compte ne soient pas situées à plus d'une heure de trajet, et cela pour des raisons de conditions de transports. L'impact se fera donc sentir sur le pôle logistique.

Le second levier est la prise en compte des nombreux types de dispensations existants afin de pouvoir tester des situations différentes. L'idée est donc de pouvoir composer sa distribution comme l'hôpital l'entend. Il sera donc nécessaire de donner la possibilité à l'utilisateur de répartir comme bon lui semble sa distribution (par exemple : 50 % en Globale et 50% en Nominative informatisée).

Le levier humain est quant à lui le troisième curseur que l'hôpital souhaite intégrer à ce modèle. Bien qu'évident, il est toujours bon de le rappeler afin de ne pas oublier, il s'agit du levier qui aura certainement un des plus gros impacts sur l'efficacité. Ainsi, l'utilisateur devra pouvoir choisir la composition de ces équipes, la répartition des professions, leurs rôles, etc.

Le dernier levier est quelque peu particulier. En effet, celui-ci fait majoritairement référence aux robots de dispensations. En effet, l'hôpital désire pouvoir choisir quand les robots de dispensations (si utilisés) prépareront les prescriptions. Il est usuel de considérer l'existence de trois « cycles » potentiels. Il y est fait référence sous les termes de préparation « Matin », « Midi » et « Soir ». Ainsi une préparation le « Matin » permettra de dispenser les traitements des patients pour des prises midi/soir/matin, une préparation le « Midi » permettra une prise soir/matin/midi, et enfin le « Soir/Nuit » permet de donner les doses pour le matin/midi/soir.

Chacun de ces cycles présentent des avantages et des inconvénients comme nous allons le voir. Pour estimer l'impact de ces cycles, il est nécessaire de considérer certains paramètres extérieurs à savoir la prise en compte des modifications matinales sur le traitement des patients, le travail de nuit et enfin la prise en compte des prescriptions des nouveaux patients, ceux-ci prenant place l'après-midi. Le Tableau 7.2 dresse le bilan des différents cycles.

Tableau 7.2 : Cycles de préparation et impact

Cycle/Critères	Modification matinale des prescriptions	Travail de nuit évité	Prescription nouveau patient hors cycle de préparation
C1 : Matin		X	X
C2 : Midi	X	X	
C3 : Soir/Nuit	X		X

Le cycle C1, correspondant à une préparation le matin, permet d'éviter le travail la nuit mais à l'inconvénient majeur de ne pas intégrer les modifications de la contre-visite du matin réalisée par les médecins. Cependant il prend en compte les prescriptions réalisées pour les admissions de nouveaux patients l'après-midi. Le second cycle intègre quant à lui les modifications des contre visites matinales, évite le travail de nuit, mais ne prends pas en compte le traitement des nouveaux

patients. Enfin le dernier cycle prend en compte les modifications matinales et les nouveaux traitements mais a malheureusement lieu la nuit, ce qui augmente le coût de préparation de ces traitements. Au final, le choix du cycle dépendra de la volonté de la structure et de ses objectifs.

Le Tableau 7.3 présente donc un bilan des leviers de simulation requis par les demandeurs :

Tableau 7.3 : Récapitulatif des leviers de simulation

Leviers	Critères à prendre en compte
Préparer des demandes pour une structure extérieure	Type de demandes traitées
Choisir les types de distributions/dispensations	Au choix entre : <ul style="list-style-type: none"> - DJIN manuelle ou automatisée - Dispensation globale - Dispensation globalisée - Dispensation nominative partielle
Choisir la composition des équipes de travail	Horaire pour chaque agent Type de chaque agent
Choisir les cycles de préparations des robots de dispensation	Nombre de robots Capacité de productions des robots Horaires et durée des cycles

7.3 SIMIO

Nous allons dans un premier temps présenter très rapidement SIMIO, avant de détailler son contenu en passant en revue une grande partie des options que ce logiciel mets à la disposition des utilisateurs.

7.3.1 Qu'est-ce que SIMIO ?

SIMIO est un outil permettant de concevoir et d'exécuter des modèles dynamiques de systèmes afin de voir comment ceux-ci évoluent et se comportent. SIMIO fournit une animation 3D du comportement du système à travers le temps permettant de visualiser l'impact des changements apportés directement.

SIMIO aborde la conception du modèle, de la manière la plus répandue qu'il soit aujourd'hui, à savoir le « drag and drop ». Cela facilite grandement la réalisation des modèles et permet ainsi de concevoir facilement un large panel de systèmes incluant aussi bien les industries, les supply-chains, le transport, la défense, le domaine minier ou encore hospitalier.

Pour concevoir ces différents modèles, SIMIO donne donc accès à un certain nombre d'interfaces permettant d'aborder différents aspects de la construction du modèle.

7.3.1.1 Interface « Facility » et ses options

Cette première fenêtre (Figure 7.1) permet de définir la logique du modèle en termes d'objets animés et il est possible d'y observer différents bandeaux. Celui du haut présente les différentes options liées à ce mode « Facility » où il est construit visuellement le modèle de simulation par la technique du « drag and drop ». Ainsi en déposant les nombreux éléments de la

librairie standard (le bandeau vertical de gauche), que nous détaillerons plus tard, il est possible de concevoir un modèle sur lequel les options « Run », « Drawing », « Animation », « View » et « Visibility » pourront chacune agir. Tandis que la première permet de faire tourner le modèle de simulation en vue d'obtenir des résultats, la seconde « Drawing » donne la possibilité à l'utilisateur de jouer sur les textures des éléments, de rajouter des couleurs, des formes, etc. La troisième, « Animation », permet de rajouter de nombreux indicateurs visuels aussi bien sous forme de camembert, de graphique, de texte, etc. L'option « View » rends possible la réalisation de film sur la simulation, ainsi que l'ajout d'arrière-plan et le passage du mode 2D à 3D. Enfin le « Visibility » permet de cacher ou d'afficher les nombreux éléments qui apparaissent sur la fenêtre graphique de SIMIO.

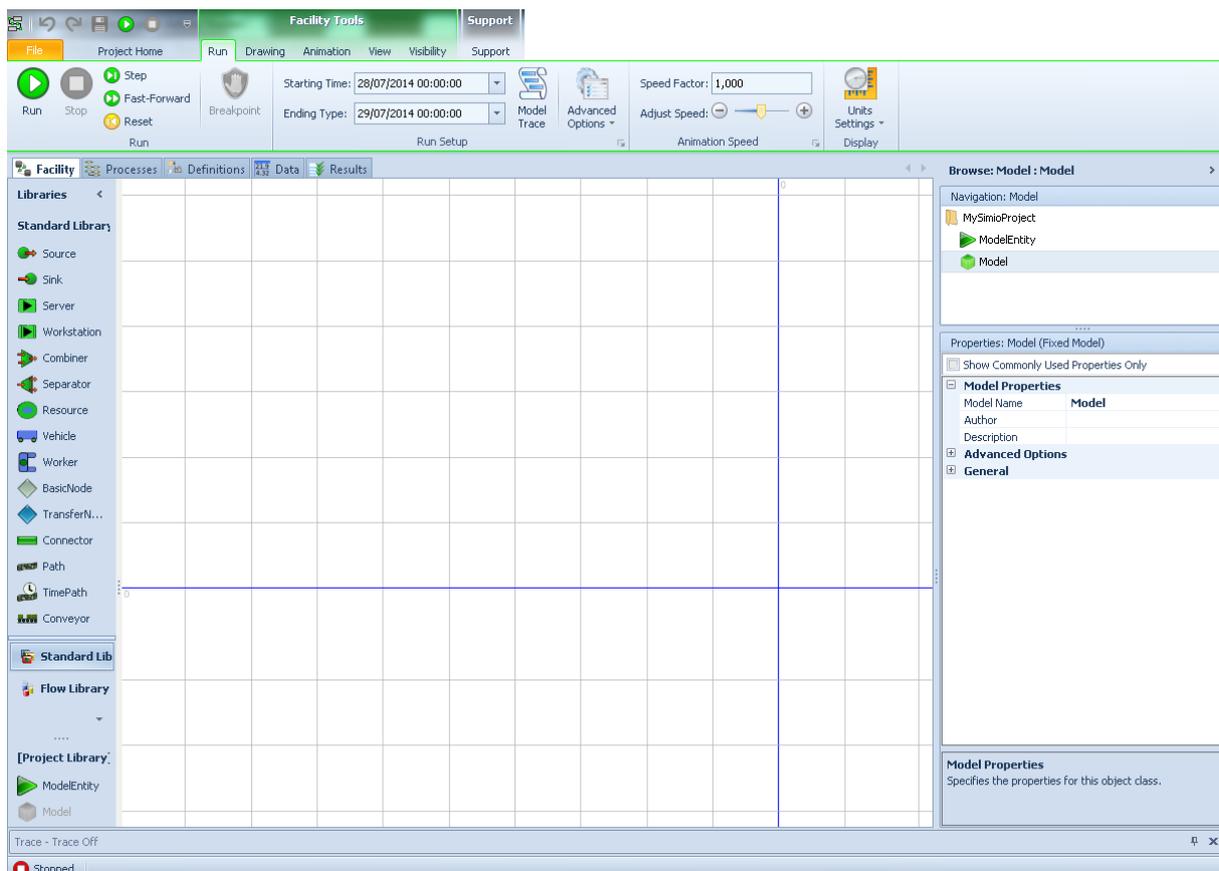


Figure 7.1 : Interface « Facility » positionnée sur l'option « Run »

Le dernier bandeau, à droite verticalement, affiche les différents éléments composant notre projet. En effet, un modèle est défini à l'intérieur d'un projet, pouvant contenir lui-même de nombreux modèles différents. Typiquement un modèle contient un « modèle principal » et un « modèle d'entité », ce dernier permettant de définir le comportement des entités qui parcourront le modèle. Tout comme il peut y avoir de nombreux modèles différents dans un projet, il peut y avoir de nombreuses entités différentes (et par conséquent de « modèles d'entités » différents).

7.3.1.2 Interface « Processes » et ses options

La relative simplicité de SIMIO à construire des modèles à un inconvénient majeur qui est celui de la flexibilité. En effet, en se limitant au comportement des objets tel que conçus dans la

librairie standard, l'utilisateur se retrouve vite sans possibilité de faire évoluer son modèle. Ainsi, dans le but d'obtenir des objets qui fonctionnent dans de nombreuses situations et dans de nombreux domaines d'application, sans pour autant devenir extrêmement complexe, SIMIO intègre la notion de processus (« Processes » dans la version originale). Ici, cette notion de processus, fait référence à une fonction logique pouvant être insérée dans les objets de la librairie standard à certains endroits afin d'en améliorer le fonctionnement. Ces processus peuvent aussi bien servir lors des phases de saisie/libération d'entités, lors de l'assignation de variables, du choix d'un trajet, etc. Ainsi SIMIO combine les avantages de la modélisation orientée objet avec la force et la flexibilité de la modélisation graphique des processus (Figure 7.2).

SIMIO voit les processus comme une séquence d'étapes exécutées par un « jeton » et qui peuvent changer l'état d'un ou plusieurs éléments. Ce « jeton » pouvant lui aussi avoir ses propres propriétés et états. Les étapes composant ces processus possèdent des propriétés mais n'ont pas d'états. Ainsi les étapes ont des valeurs d'entrées qui contrôlent leur comportement, mais ne possèdent pas de variables qui changent durant la simulation.

Il existe dans SIMIO trois grands types de processus, qui n'ont de différence que la manière dont ils sont déclenchés. Le premier type est le processus « standard » qui s'exécute quand celui-ci doit l'être. En effet ce type de processus ne nécessite pas d'évènement déclencheur. Le second type est le processus « add-on ». Celui-ci sera intégré à un objet du modèle et sera déclenché dans des circonstances bien précises visibles dans le Tableau 7.4.

Tableau 7.4 : Situation d'exécution des processus

Situation d'exécution du processus	Description
« Run initialized »	Se déclenchera durant l'initialisation du modèle
« Run Ending »	Se déclenchera lorsque la simulation prend fin
« Entered »	Se déclenchera immédiatement après qu'une entité soit entrée dans l'objet
« Before processing »	Se déclenchera quand l'entité a réquisitionné une capacité d'un serveur mais avant d'entrée en production
« Processing »	Se déclenchera juste avant qu'une entité commence sa production
« After processing »	Se déclenchera quand une entité a complété sa production et est sur le point de libérer la capacité serveur réquisitionnée
« Exited »	Se déclenchera immédiatement après qu'une entité ait quittée l'objet
« Failed »	Se déclenchera immédiatement après qu'un objet soit dans un état d'erreur (de panne)
« Repaired »	Se déclenchera immédiatement après qu'un objet soit dans un état de réparation
« On shift »	Se déclenchera après que l'objet se soit mis en fonctionnement
« Off shift »	Se déclenchera après que l'objet se soit arrêté de fonctionner (mode veille par exemple)

La liste ci-dessus n'est bien sûr pas exhaustive, et il existe d'autres circonstances permettant de déclencher des processus add-on suivant l'objet auquel il sera associé.

Le dernier type de processus est celui déclenché par un évènement précis du modèle. Cet évènement peut être de diverses natures telles qu'un changement de capacité de production d'un serveur, un changement de travailleur pour un serveur, etc.

Nous reviendrons sur les processus en partie 7.3.3, afin d'expliquer comment les construire.

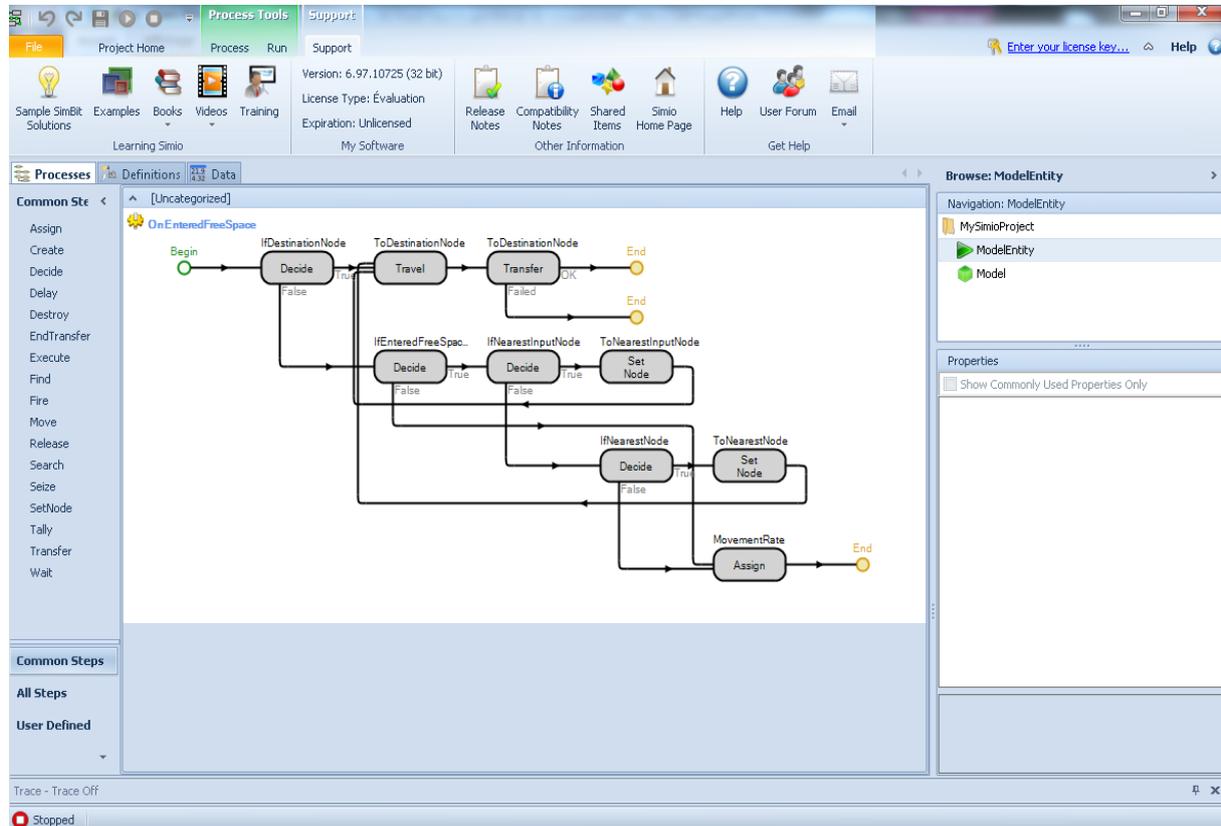


Figure 7.2 : Aperçu de l'interface « Processes »

7.3.1.3 Interface « Definition » et ses options

Cette troisième interface (Figure 7.3) donne à l'utilisateur accès aux neuf options permettant de compléter le comportement des objets, à savoir : « Elements », « Properties », « States », « Events », « Functions », « Lists », « Tokens », « External view », et enfin « Console ».

Les « Elements » sont des composants du modèle avec des propriétés, des états et un comportement spécifiques. Ils peuvent être de différentes natures comme il est possible de le voir dans le Tableau 7.5 (tableau non exhaustif).

Les « Properties » sont des paramètres d'entrée statiques dans le sens où elles ne changent pas durant une simulation. Pouvant être de différentes natures (entier, expression, chaîne de caractères, etc.), elles peuvent aussi faire référence à des objets ou à des « Elements ».

Les « States » sont contrairement aux propriétés des variables qui peuvent évoluer au fil de la simulation. Les « States » peuvent aussi bien être continu ou discret et sont de nature identique aux « Properties ».

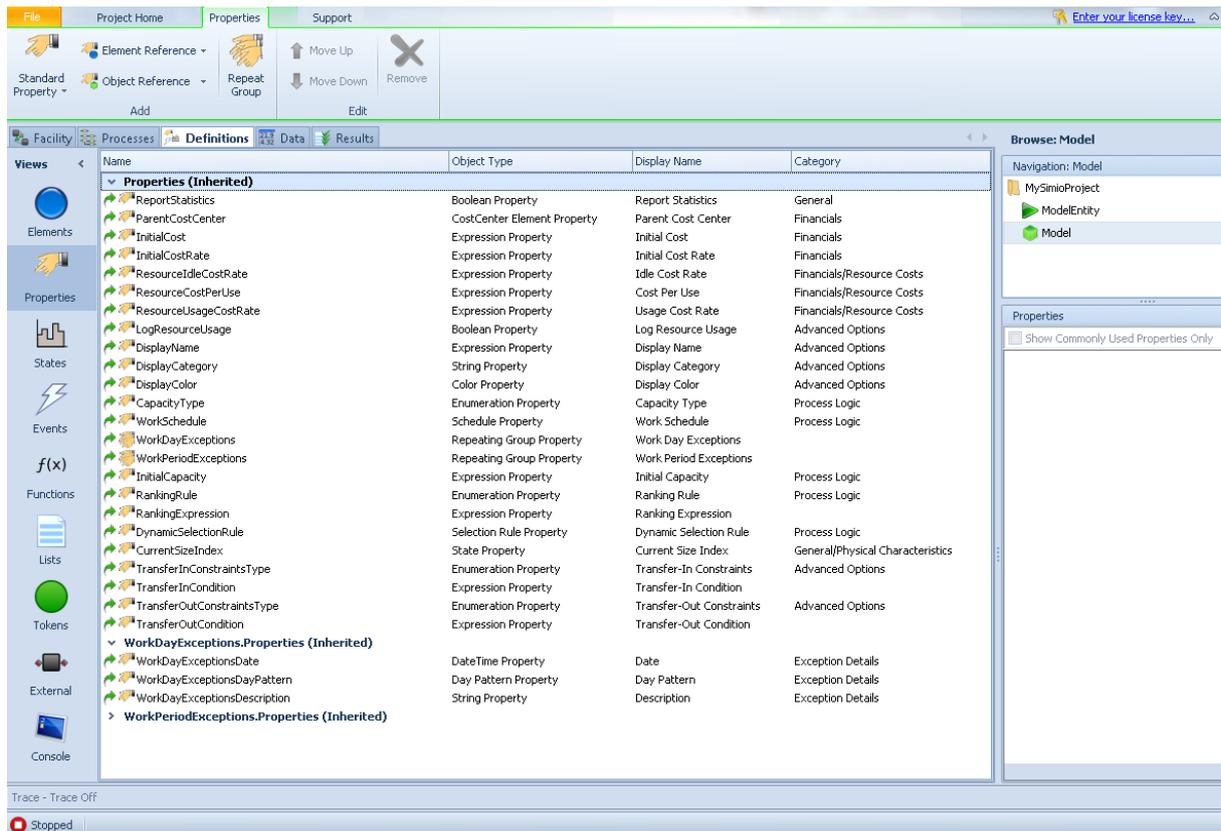


Figure 7.3 : Aperçu de l'interface « Definitions »

Tableau 7.5 : Principaux types d'« Elements »

Nom de l'élément	Description
« Station »	Définit un emplacement à capacité limité dans un objet
« Timer »	Déclenche un flux d'évènements
« Failure »	Définit un état de panne pour l'objet parent
« BatchLogic »	Forme un ensemble d'entités et les lie à l'entité parent
« Storage »	Définit une file d'attente stockant temporairement une ou plusieurs entités dans un ordre précis
« Monitor »	Déclenche un évènement quand l'état change ou atteint une valeur spécifique
« Network »	Définit un réseau de liens
« Routing Group »	Définit une logique de déplacement en sélectionnant un nœud destination pour l'entité y arrivant
« Activity »	Définit une activité qui est réalisée dans une opération
« Operation »	Définit une séquence d'activités réalisée au cours le temps
« Material »	Définit un matériel qui peut être produit ou consommé et qui possède une nomenclature

Les « Events » sont des notifications qui sont envoyées par un objet, et auxquelles plusieurs objets peuvent répondre.

Les « Fonctions » sont des expressions nommées qui sont évaluées chaque fois qu'il est fait référence à la fonction. Elles sont généralement utilisées pour simplifier l'utilisation d'expressions complexes en pouvant y faire référence uniquement grâce à un nom.

Les « Lists » sont utilisées pour définir des collections de chaîne de caractères, d'objets, de nœuds, de transporteurs. Elles peuvent aussi être sollicitées dans le cadre de matrice de changement (« Changeover matrix »). Les éléments de la liste possèdent tous un numéro (débutant à zéro) afin de pouvoir y faire référence lors de leur utilisation.

Les « Tokens » exécutent les étapes dans les processus, peuvent avoir un ou plusieurs états et transportent des informations d'étapes en étapes. SIMIO fournit par défaut un jeton « Token » utilisable par tous les processus, mais s'il y a besoin d'un jeton avec des états spécifiques, il est préférable de les créer.

L'option « External » permet de définir la forme graphique que prendra l'objet lorsqu'il sera visualisé. Il est ainsi possible de changer la forme, texture, etc. des objets composants le modèle.

Enfin la « Console » permet de visualiser en temps réelle l'évolution de certains critères à l'aide de divers graphiques ou boîtes de dialogue.

7.3.1.4 Interface « Data » et ses options

Les modèles de simulation utilisent de nombreuses informations pour pouvoir être exécutés, ce qui peut vite se révéler problématique avec la multiplication des utilisateurs et des modes de simulation. SIMIO donne donc accès à une interface « Data » où il est possible de définir de nombreuses tables de données différentes.

Parmi les nombreuses options de cette interface, il est possible de définir des tables standards, des plannings, des matrices, des tables d'arrivées et enfin des tables de recherches. Chacune de ses options donne bien sûr accès à diverses possibilités. Les tables standards font référence aux classeurs qu'il est possible de retrouver sous EXCEL par exemple. Il est aussi possible de définir des tables de séquences utilisées par de nombreux objets de SIMIO. Ces dernières permettent comme énoncé de construire des séquences (par exemple une gamme de production) qui seront suivies par les objets du modèle, ces séquences étant complétées par de nombreuses données disponibles dans les colonnes suivantes (Figure 7.4).

La seconde option permet de définir des plannings et cela avec une grande flexibilité (semaine, jour, heure, jour férié, heure spécifique, etc.). Les matrices quant à elles permettront de faire le lien entre des variables différentes, tandis que les tables d'arrivées rendent possible de préciser un rythme d'arrivée des entités (pour une source avec par exemple une entité par heure). Enfin les tables de recherche permettent de retourner une valeur obtenue à l'aide d'une courbe, cette valeur étant dépendante d'un paramètre extérieur (par exemple le temps d'usinage d'une pièce serait dépendant de l'expérience de l'ouvrier).

Majoritairement, nous utiliserons des tables standards dans notre modèle, chose que nous justifierons plus tard. Il est possible de définir de nombreux types de variable différents dans ces tables tels que des entiers, des réels, des expressions mathématiques, des événements, des états mais aussi des variables faisant référence à des listes, d'autres tables standards ou de séquence, des plannings, etc.

A noter que, SIMIO permet aussi d'importer des tables de l'extérieur, notamment à partir du logiciel EXCEL, ce qui se révélera très utile dans la suite de nos travaux.

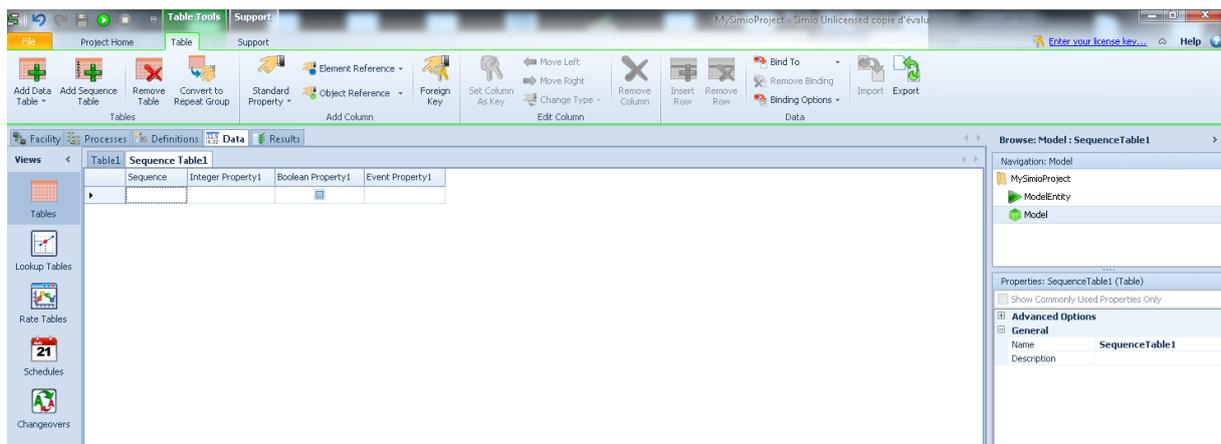


Figure 7.4 : Aperçu de l'interface « Data » avec une table de séquence

7.3.1.5 Les modes d'exécutions

Comme nous l'avons dit plus tôt, il existe le mode « Run » permettant de lancer la simulation de notre modèle. Ce mode permet de visualiser en direct, l'évolution de notre système grâce à la représentation 3D. De plus, il est possible d'observer par la même occasion des graphiques dynamiques s'ils ont été placés dans le modèle par avance. Ce mode se prête particulièrement bien à la construction et à la validation du modèle.

Une fois le modèle de simulation construit, il est coutume de concevoir des scénarios à faire tester par le modèle. Dans ce cas de figure, l'aspect graphique devient moins primordiale, tandis que l'aspect réplication devient une nécessité.

Il s'agit donc du second mode de simulation appelé « Experiment ». Ce mode donne la possibilité de choisir une ou plusieurs propriétés du modèle afin d'en observer les impacts sur les performances du système. Ces propriétés peuvent faire référence à la vitesse de déplacements de certains objets, aux nombres d'opérateurs mais encore à des règles de décisions.

Quand toutes les propriétés sont connues, il est alors possible de créer des scénarios dans ce mode, où chacun d'entre eux sera modélisé par une ligne possédant divers paramètres dont un faisant référence au nombre de réplifications souhaitées pour ce scénario (Figure 7.5).

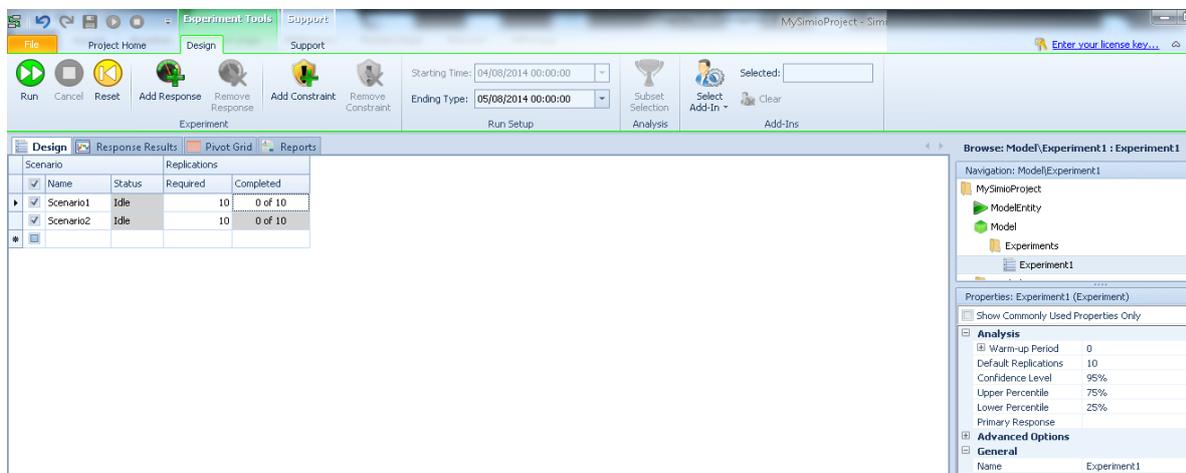


Figure 7.5 : Aperçu de l'interface « Experiment »

7.3.2 Les objets

Les objets sont les éléments qui vont interagir avec les entités qui parcourent le système. Comme nous allons le voir ces éléments possèdent des classes et sont de nature diverses et variées.

7.3.2.1 Les classes d'objets

En premier lieu, il est possible de trier les objets par classe. Actuellement SIMIO les définit suivant cinq classes majeures :

- ❖ Fixe : Un objet de ce type sera fixe dans le modèle (une machine-outil par exemple) et possède des nœuds d'entrée et de sortie par lesquels transitent les entités. De plus, ils peuvent utiliser des ressources.
- ❖ Lien : Ce type d'objet définit un chemin entre deux nœuds pour les entités et les transporteurs. Ils peuvent être aussi bien unidirectionnels que bidirectionnels, appartenir à un ou plusieurs réseaux et possèdent un « poids », utilisable pour orienter le parcours des entités (la logique de parcours peut être du « plus court chemin » ou « par poids »).
- ❖ Nœud : Ils définissent le début et la fin de un ou plusieurs liens, et symbolisent l'intersection de ces nombreux liens. Comme déjà dit, les nœuds peuvent être libres ou bien associés à un objet « Fixe » pour lequel il servira soit de point d'entrée soit de point de sortie. Enfin, les « Nœuds » ont une zone de parking où les entités peuvent patienter sans pour autant bloquer l'intersection.
- ❖ Agent : Un « Agent » possède une ou plusieurs représentations 3D dans le modèle, peut se déplacer librement dans celui-ci et possède une population initiale. Les « Agents » se découpent en deux catégories :
 - Entité : Les entités sont des objets dynamiques pouvant être créées et détruites, voyager entre les nœuds en empruntant des liens et entrer et sortir des objets fixes. Elles peuvent de plus être assignées à un réseau précis et suivre une séquence de déplacement spécifique.
 - Transporteur : Ils sont des éléments qui peuvent prendre, déposer et transporter des entités. Ils peuvent gérer de diverses manières des tournées et suivre différentes politiques quant au prélèvement et au dépôt des entités. Ils possèdent de plus une station dédiée.

La classe de l'objet ne définira que son comportement général, mais pas son comportement spécifique. En effet, ce dernier est défini par le modèle interne de l'objet. Il est donc possible d'avoir dans un modèle douze transporteurs différents chacun ayant son propre comportement, mais partageant tous la faculté de se déplacer sur un réseau de nœuds et de liens, et pouvant saisir, déposer des entités.

7.3.2.2 Les objets de la librairie standard

SIMIO propose une base de seize objets (Tableau 7.6), appartenant chacun à une des classes ci-dessus, et permettant de modéliser et concevoir les modèles souhaités.

Tableau 7.6 : Objets appartenant à la librairie standard de SIMIO

Objet (Nom dans le logiciel)	Description
Source	Génère des objets « entité » d'un type spécifique et suivant un schéma d'arrivée défini
Puits (Sink)	Détruit les entités qui ont complété leur processus dans le modèle
Serveur (Server)	Représente un processus d'une capacité finie telle qu'une machine-outil ou une opération manuelle
Station de travail (Workstation)	Modélise une station de travail complète comprenant les phases de chargement, production et déchargement, ainsi que les besoins en ressources et matériels
Combineur (Combiner)	Combine de multiples entités « membre » avec une entité « parent » (par exemple des produits sur une palette)
Séparateur (Separator)	Sépare un ensemble d'entités ou crée des copies d'une entité
Ressource (Resource)	Un objet générique qui peut être saisi et libéré par d'autres objets
Véhicule (Vehicle)	Un transporteur qui peut suivre une route fixe ou répondre à des demandes de ramassage et de dépose
Travailleur (Worker)	Une ressource pouvant se déplacer et qui peut être saisie et libérée aussi bien pour des tâches que pour le transport entre deux nœuds
Nœud de base (BasicNode)	Modélise une simple interaction entre des liens multiples
Nœud de transfert (TransfertNode)	Modélise une intersection complexe pour changer de destination et de mode de transport
Connecteur (Connector)	Un lien entre deux nœuds dont le temps de parcours est nul
Chemin (Path)	Un lien sur lequel les entités peuvent voyager à la vitesse qu'elles veulent
Chemin à temps défini (Timepath)	Un lien qui définit un temps de trajet précis
Convoyeur (Conveyor)	Un lien modélisant un convoyeur pouvant accumuler ou non, suivant ce qui est désiré, des entités
ModelEntity	Une entité représentant une pièce, une personne, ou tout autre objet qui est dynamiquement créé par le système, transite dans celui-ci et le quitte.

Les différents objets présentés ci-dessus vont maintenant être détaillés avec plus de précisions, chacun d'entre eux possédant des propriétés qui régissent son comportement. Mais avant cela, faisons un point rapide sur les entités (« ModelEntity »).

7.3.2.3 Les attributs des entités « ModelEntity »

Les entités du modèle sont des objets dynamiques qui peuvent voyager à travers le modèle, être créées ou détruites, entrer et sortir d'autres objets (fixes). Ces entités possèdent des propriétés et des états, étant plus communément appelés « Attribut ». Ces attributs sont définis communément pour toutes les entités d'une même nature, mais chacune d'entre elles peut posséder une valeur différente pour chaque attribut.

↳ L'« Attribut » de type « Propriété » est statique et ne peut changer durant une simulation. Ces propriétés peuvent être un coût horaire, un temps de production, une vitesse initiale, etc.

La « Propriété » peut elle-même être de différent type :

- ❖ Standard : un entier, un réel, une expression, une variable booléenne, etc. ;
- ❖ Elément : Une activité, une erreur, une opération, etc. ;
- ❖ Objet : Une entité, un transporteur, un nœud, etc. ;
- ❖ Un groupe répété : Un ensemble de propriétés répétées sélectionnées indifféremment parmi les précédentes.

A noter que même si une propriété de type expression est statique, vu que celle-ci peut comporter des variables aléatoires et des états dynamiques, sa valeur peut changer au cours du temps.

↳ L'« Attribut » de type « Etat » sera quant à lui dynamique et pourra changer durant la simulation. Il pourra par exemple être le nombre d'entités créées ou détruites, la vitesse actuelle de l'objet, l'état d'une machine, etc. Tout comme la « Propriété », l'« Etat » possède différents types :

- ❖ Discret : l'« Etat » ne change que quand le modèle le lui signifie. Il peut donc être un réel, un entier, une variable booléenne, une date, une liste, une chaîne de caractères. Il a aussi la possibilité de faire référence à un « Elément » ou à un « Objet ».
- ❖ Continu : il s'agit d'un état qui peut changer continuellement à travers le temps et cela en fonction d'un taux, ce dernier pouvant évoluer discrètement. Il peut aussi être fonction d'un taux et d'une accélération à la fois, ceux-ci évoluant donc discrètement à travers le temps.

De plus, un « Etat » a la possibilité d'être un scalaire ou un vecteur (10 dimensions maximum). Enfin, l'« Etat » peut être hérité ou non, ce qui est le cas des états qui sont automatiquement générés à la création de l'objet. Les autres sont ajoutés par l'utilisateur.

7.3.2.4 La Source et le Puits

La source (Figure 7.6) est utilisée pour générer des objets arrivant suivant des séquences précises ou aléatoires. Quittant la source par son « Output node » (losange bleu), ces arrivées représentent les entités dynamiques qui entrent et quittent le système et peuvent aussi bien représenter des clients, patients, messages, ordinateurs, etc.



Figure 7.6 : Modèle SIMIO de la Source

Les propriétés de base de la Source sont présentées dans le Tableau 7.7.

De base, la source génère automatiquement des « Default Entity » si aucune entité n'est spécifiée dans le « Entity Type » (Type d'entité) et son « Arrival Mode » (Mode d'arrivée) est d'office paramétré pour suivre une loi de Poisson. Le nombre d'« Entities per arrival » (Entités par arrivée) est par défaut d'une unité, mais celle-ci peut être représentée par n'importe quelle expression mathématique qui sera évaluée à chaque arrivée.

Chaque propriété ci-dessus a donc une valeur par défaut afin de permettre une mise en route rapide des modèles, et il en va de même pour les nombreuses autres propriétés de la source non présentées ici, ainsi que pour tous les objets qui seront présentés par la suite.

Tableau 7.7 : Aperçu des propriétés basiques d'une Source

Type de propriété	Nom de la propriété	Description
Logique d'arrivée des entités		
	Type d'entité	Le type de l'entité créée par la source
	Mode d'arrivée	Le mode utilisé pour générer le flux d'entités
	Durée de report	Le temps avant la création de la première entité
	Intervalle de temps	Le temps entre deux arrivées successives d'entités
	Entité par arrivée	Le nombre d'entités créées à chaque arrivée
Condition d'arrêt		
	Nombre d'arrivée maximum	Le nombre d'arrivée devant être généré
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

Le Puits (Figure 7.7) quant à lui, est utilisé pour représenter la sortie des entités du système modélisé. Ainsi quand une entité entre dans le Puits, celle-ci est détruite et retirée du système. Les propriétés du puits n'étant pas très utiles nous ne les détaillerons pas.



Figure 7.7 : Modèle SIMIO du Puits

7.3.2.5 Le Serveur

Celui-ci est utilisé pour modéliser une ressource contrainte telle qu'une machine, un service qui a une position fixe dans le système. Parmi les exemples possibles, on peut citer, un comptoir pour le check-in dans les aéroports, un distributeur de billet, etc. Le Serveur (Figure 7.8) a un nœud d'entrée (losange gris) et un nœud de sortie (losange bleu) par où transitent obligatoirement les entités. Les propriétés communes du Serveur sont détaillées dans le Tableau 7.8.

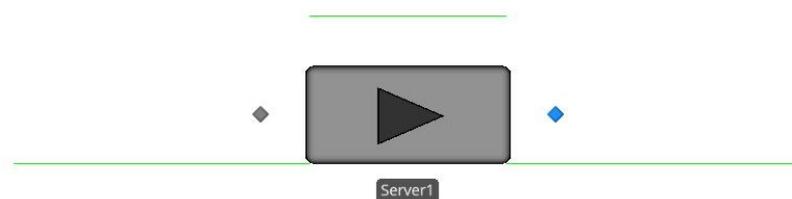


Figure 7.8 : Modèle SIMIO du Serveur

Tableau 7.8 : Aperçu des propriétés basiques d'un Serveur

Type de propriété	Nom de la propriété	Description
Logique du processus		
	Type de capacité	Le type de capacité du serveur, à savoir « Fixe » ou variable suivant un planning
	Capacité Initiale	Le nombre d'entités qui peuvent être traitées simultanément par le Serveur
	Règle de sélection	La règle définissant l'ordre de sélection des entités devant être traitées
	Temps de production	Le temps requis par le Serveur pour traiter chaque entité
Capacité des stocks tampon		
	Stock tampon en entrée	Le nombre d'entité pouvant être stockées dans le stock en entrée
	Stock tampon en sortie	Le nombre d'entité pouvant être stockées dans le stock en sortie
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

La logique de fonctionnement d'un Serveur est la suivante. Les entités rentrent par le « Input Node » et attendent dans le « Input Buffer » si le Serveur est indisponible. Une fois traitée, elles passeront par le « Output Buffer » avant de sortir par le « Output Node ». Il est important de noter qu'à tout instant une entité est pour SIMIO à la fois dans le « Input Buffer », la « Processing Station » et le « Output Buffer ».

La capacité de production du Serveur est régie à travers le temps par les propriétés « Capacity Type » (Type de capacité) et « Initial Capacity » (Capacité Initiale), la première pouvant être fixe (fonctionnement par défaut) ou « Planifiée » c'est-à-dire évolutive suivant un planning défini dans l'interface « Data » ([partie 7.3.1.4](#)). La propriété « Processing Time » (Temps de production) est par défaut définie par une loi triangulaire, mais peut être représentée par n'importe quelle formule ou loi mathématique.

7.3.2.6 La Station de travail

Similaire au Serveur dans la forme, la Station de travail a pour but d'apporter un fonctionnement plus flexible à ce premier tout en limitant sa capacité de production à une unité à la fois. La Station de travail (Figure 7.9) permet de compléter le procédé du Serveur en rajoutant avant l'activité de production, une activité de « montage » (« Setup Activity ») et après la production, une activité de « démontage » (« Teardown Activity »). Le Tableau 7.9 présente les propriétés les plus communes de cet élément.

La propriété « Setup time type » (Type temps de montage) définit la méthode utilisée pour déterminer le temps de montage et peut être spécifique (par défaut), dépendante d'un changement ou dépendante d'une séquence.

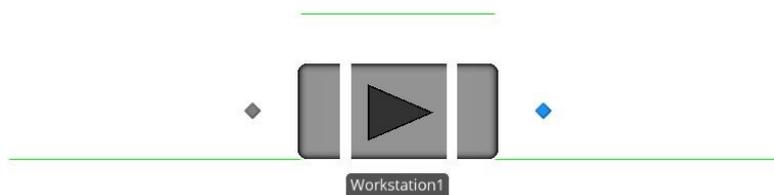


Figure 7.9 : Modèle SIMIO de la Station de travail

Tableau 7.9 : Aperçu des propriétés basiques de la Station de travail

Type de propriété	Nom de la propriété	Description
Logique du processus		
	Type de capacité	Le type de capacité, à savoir « Fixe » ou variable suivant un planning
	Règle de sélection	La règle définissant l'ordre de sélection des entités devant être traitées
	Type de temps de montage	La méthode utilisée pour spécifier le temps de montage
	Temps de montage	Le temps requis pour réaliser le montage afin de commencer la production
	Temps de production	Le temps requis par la Station pour traiter chaque entité
	Temps de démontage	Le temps requis pour réaliser le démontage suivant la production
Capacité des stocks tampon		
	Stock tampon en entrée	Le nombre d'entité pouvant être stockées dans le stock en entrée
	Stock tampon en sortie	Le nombre d'entité pouvant être stockées dans le stock en sortie
Ressources secondaires		
	Ressources secondaires	Donne accès à un éditeur permettant de spécifier des ressources requises pendant les phases de montage, production et de démontage
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

« Setup time » (Temps de montage) peut être une constante, un attribut de l'entité traitée, ou une valeur définie pour l'entité. Dans le cas où il est dépendant d'un changement, cela signifie que le temps changera suivant une caractéristique de l'entité (couleur, forme, etc.), alors que dans le cas d'une dépendance à une séquence, cela voudra dire que la valeur changera suivant une matrice de changement présente dans l'interface « Data ».

Ici il est fait référence à la possibilité de rajouter une ou plusieurs ressources secondaires. Il faut savoir que c'est aussi le cas pour le Serveur.

7.3.2.7 La Ressource

La Ressource est un objet générique pouvant être saisi et libéré par les autres objets composant le modèle. Par exemple une station de travail pourrait requérir un équipement spécifique comme ressource secondaire pour la production d'une pièce. Cet équipement peut être modélisé par la Ressource (Figure 7.10) et serait sélectionné dans l'option « Secondary Resource » de la station. Etant un objet pouvant être saisi ou libéré, la Ressource ne possède pas de nœuds d'entrée et de sortie. Ses propriétés principales sont présentées dans le Tableau 7.10.

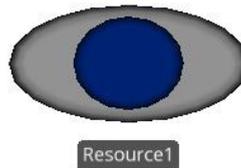


Figure 7.10 : Modèle SIMIO de la Ressource

Tableau 7.10 : Aperçu des propriétés basiques de la Ressource

Type de propriété	Nom de la propriété	Description
Logique de la ressource		
	Type de capacité	Le type de capacité de la ressource, à savoir « Fixe » ou « Planifiée »
	Capacité Initiale	La capacité initiale de la ressource
	Règle de sélection	La règle définissant l'ordre de sélection des objets requérant cette ressource
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

Définie par défaut sur « Fixe », la capacité de la ressource peut aussi être utilisée suivant un planning. La propriété « Initial capacity » définit le nombre d'unité de cette ressource pouvant être utilisé simultanément dans le modèle à son démarrage.

Faisant partie des objets fixes de la librairie standard, la Ressource ne peut se déplacer. Bien qu'elle puisse être utilisée par de nombreux objets dispatchés à travers le modèle, celle-ci ne bougera pas. Cet objet sera donc sollicité pour modéliser des ressources dont la localisation n'a pas d'importance. Dans le cas contraire, l'objet Travailleur sera utilisé, et nous allons maintenant le présenter.

7.3.2.8 Le Travailleur

Faisant parti de la catégorie « Transporteur », cet objet (Figure 7.11) peut être saisi et libéré par les autres objets, et peut transporter des entités et se mouvoir librement dans le modèle. Principalement utilisé pour modéliser des opérateurs dans une usine, des docteurs, médecins, infirmières dans les systèmes de santé mais encore du personnel administratif dans une banque, le Travailleur peut être utilisé aussi bien en tant que ressource principale que secondaire. Cependant

contrairement à la Ressource, le Travailleur se déplace et lorsque celui-ci est sollicité par un objet, il doit se mouvoir jusqu'à celui-ci. Les principales propriétés du Travailleur sont énoncées dans le Tableau 7.11 suivant.



Figure 7.11 : Modèle SIMIO du Travailleur

Tableau 7.11 : Aperçu des propriétés du Travailleur

Type de propriété	Nom de la propriété	Description
Logique de la ressource		
	Type de capacité	Le type de capacité, à savoir « Fixe » ou « Planifiée »
	Règle de sélection	La règle définissant l'ordre de sélection des objets requérant cette ressource
Logique de déplacement		
	Vitesse de déplacement initiale	La vitesse de déplacement de base du Travailleur
Logique de trajet		
	Nœud d'origine	Le nœud duquel partira le travailleur au début de la simulation
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

7.3.2.9 Le Combineur et le Séparateur

Le premier (Figure 7.12 (a)) objet est utilisé pour (re)grouper un ensemble d'entités, référencées sous le terme d'« entité membre », à une entité dite « parent », cette dernière représentant une palette, une boîte, etc. Les entités membres prennent donc place sur/dans cette entité parent. Les propriétés principales du Combineur sont résumées dans le Tableau 7.12.

Par défaut, le Combineur réunit n'importe quelle entité membre avec n'importe quelle entité parent (« Batch quantity » est donc égale à 1 et « Matching Rule » à « Any Entity »). Il est donc possible de regrouper autant de membres que voulu avec un parent et cela suivant différentes règles de sélection, qu'il est possible de définir.

Cet objet possède contrairement aux autres deux stocks tampons en entrée, un pour les entités parents et un autre pour les entités membres.

Tableau 7.12 : Aperçu des propriétés basiques du Combineur

Type de propriété	Nom de la propriété	Description
Logique de regroupement		
	Quantité réunie	Le nombre d'entités membres devant être attachées à une entité parent
	Règle de sélection	Détermine comment les entités membres et parent sont réunies
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

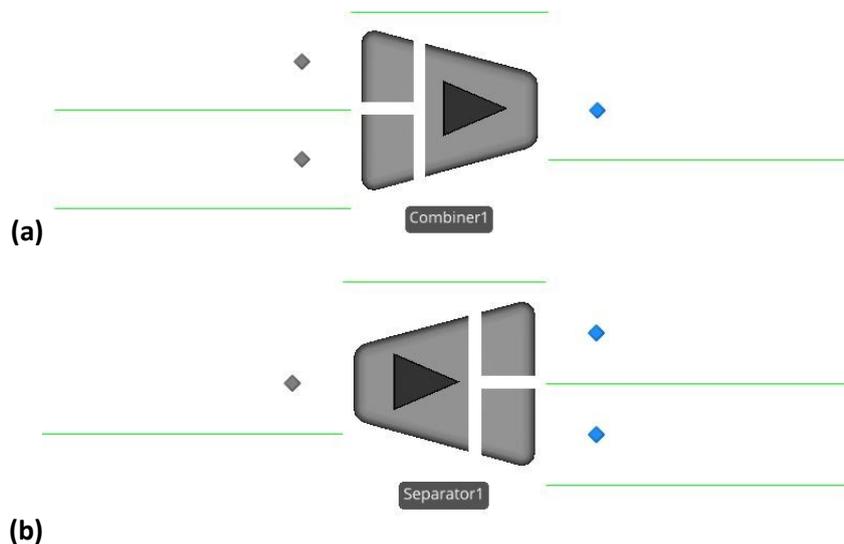


Figure 7.12 : Modèle SIMIO du Combineur (a) et du Séparateur (b)

Le second objet, le Séparateur (Figure 7.12 (b)), suit la logique contraire au Combineur. Il est donc utilisé pour séparer des entités membres des entités parents auxquelles elles ont été attachées, mais il peut aussi servir à dupliquer, à faire des copies d'entités. La première configuration est utilisée pour retirer des éléments d'un container, d'une palette, tandis que la seconde est utilisée pour cloner une entité, tel que faire une copie d'un document. Ses propriétés principales sont (Tableau 7.13) :

Tableau 7.13 : Aperçu des propriétés basiques du Séparateur

Type de propriété	Nom de la propriété	Description
Logique de séparation		
	Mode de séparation	Soit « Séparer entité » soit « Faire des copies »
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

Il est à noter que durant la séparation (ou la création d'une copie) il est possible d'assigner un certain nombre d'attributs aux entités (membre ou parent) tel que un identifiant par exemple.

7.3.2.10 Le Véhicule

Tout comme le Travailleur qui peut transporter les entités à travers le modèle, le Véhicule (Figure 7.13) fait de même, mais celui-ci peut choisir comment répondre aux sollicitations externes (« Type de trajet »). De plus, contrairement au Travailleur, il peut être soumis à des pannes. Il est donc utilisé pour modéliser un grand nombre de moyens de transport différents comme des chariots élévateurs, des camions, des bus, etc. Parmi ses propriétés principales se trouvent (Tableau 7.14) :

Tableau 7.14 : Aperçu des propriétés basiques du Véhicule

Type de propriété	Nom de la propriété	Description
Logique de transport		
	Capacité de transport initiale	La capacité de transport initiale du véhicule
Logique de déplacement		
	Vitesse de déplacement initiale	La vitesse de déplacement du véhicule au début de la simulation
Logique de trajet		
	Priorité initiale	La priorité initiale du véhicule
	Nœud d'origine	Le nœud duquel partira le véhicule au début de la simulation
	Type de trajet	Déplacement du véhicule suivant une logique « Sur demande » ou « Sur itinéraire »
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

La propriété « Routing Type » (Type de trajet) définit le comportement du véhicule à travers le modèle. Dans le cas d'un fonctionnement « On demand » (Sur demande), le véhicule se déplacera jusqu'à l'objet ayant requis sa présence. Au contraire dans le cas d'un fonctionnement « Fixed route » (Sur itinéraire), le véhicule suivra une route précise et visitera les divers objets dans un ordre bien précis. Pour cela il est utilisé des tables de séquences, présentes dans l'interface « Data ».

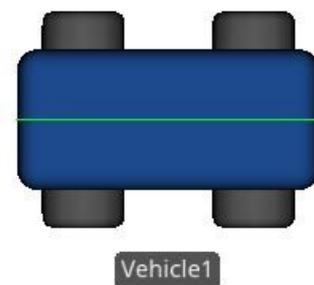


Figure 7.13 : Modèle SIMIO d'un Véhicule

7.3.2.11 Les nœuds basique et de transfert

Les nœuds sont les points d'intersections des nombreux liens composant le modèle de simulation (Figure 7.14). Quand une entité veut quitter un nœud, si plusieurs chemins sont possibles, ce sera la logique de routage du nœud « Outbound Link Rule » (Règle de sélection du lien sortant) qui déterminera le chemin à prendre. Plusieurs politiques de gestion des nœuds sont donc envisageables et rendent possible la création de réseaux complexes générant divers chemins pour relier deux points.

Les deux politiques ici possibles pour le choix du lien sortant sont celle du « Shortest Path » (Plus court chemin) et celle du « By link Weights » (Par poids). Tandis que la première utilisera toujours le chemin le plus court en sortie du nœud (afin d'atteindre une destination précise devant être définie), la seconde sélectionne suivant des probabilités le lien de sortie. Par exemple, un nœud possédant trois liens de sortie étant respectivement défini par des probabilités de 20%, 30% et 50% verra le dernier lien avoir une chance sur deux d'être sélectionné à chaque fois.

Les nœuds possèdent une file d'attente où certaines entités peuvent être stockées afin de ne pas entraver le passage des autres à travers ce nœud. Cette file d'attente nommée « ParkingStation » peut de plus être animée afin d'avoir une visualisation graphique de son contenu.

Toutes les informations précédentes s'appliquent aussi bien au nœud basique que de transfert. Cependant comme nous allons le voir, ces derniers se révèlent plus complexes. En effet, tandis que les nœuds basiques font office de nœud d'entrée, les nœuds de transfert font eux offices de nœud de sortie, ce qui nécessite de nouvelles propriétés pour pouvoir orienter correctement nos entités dans le modèle (Tableau 7.15).

Tableau 7.15 : Aperçu des propriétés basique d'un Nœud de transfert

Type de propriété	Nom de la propriété	Description
Logique de trajet		
	Règle de sélection du lien sortant	La règle utilisée par un voyageur pour sélectionner un lien sortant du nœud
	Type de destination pour l'entité	Une règle utilisée pour définir le nœud destination pour les entités
Logique de transport		
	Utilisation d'un transporteur	Spécifie si l'entité quittant le nœud doit être transportée par un Transporteur
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

Quelques précisions sur la propriété « Entity Destination Type » (Type de destination pour l'entité). Quatre possibilités sont offertes à l'utilisateur pour paramétrer cette propriété. La première est « Continue », ce qui signifie qu'aucun changement n'est fait quant au choix du nœud de destination. La seconde est « Specific », signifiant que le nœud destination sera changé et ne sera plus celui qui était défini en mode « Continue ». Le troisième mode « Select from List » donne la possibilité comme son nom l'indique de sélectionner le nouveau nœud destination dans une liste de

nœuds, la sélection pouvant avoir lieu suivant différents critères. Enfin le quatrième choix est « By sequence », l'entité sélectionnant le nœud destination suivant la séquence qui lui est assignée.



Figure 7.14 : Représentation d'un nœud basique et d'un nœud de transfert

7.3.2.12 Les Connecteurs, Chemins et Chemins à temps défini

Les trois objets ici décrits, représentent avec le dernier (le Convoyeur cf [partie 7.3.2.13](#)), les divers moyens pour faire transiter sous certaines conditions les entités entre deux nœuds.

Le premier, le Connecteur, est un type de lien, utilisé pour définir un trajet direct entre deux nœuds, dont le temps de parcours est nul. Sa propriété principale est « Selection Weight » qui permet de définir un poids pour ce lien, utilisable dans le cadre d'une règle « By link Weight » dans un nœud.

Le Chemin représente quant à lui un trajet sur lequel les entités voyageront à leur propre vitesse. Le temps de trajet est dépendant de la longueur du Chemin et de la vitesse de l'entité. Il est utilisé pour représenter des routes, des trottoirs, etc. Ses propriétés principales sont définies dans le Tableau 7.16.

Tableau 7.16 : Aperçu des propriétés basiques d'un Chemin

Type de propriété	Nom de la propriété	Description
Logique de déplacement		
	Type	Le type de déplacement, unidirectionnel ou bidirectionnel
	Autoriser le passage	Indique si l'entité est autorisée à parcourir ou non le chemin
Logique de trajet		
	Poids du chemin	Le poids du Chemin utilisé dans la règle « By Link Weight »
Affectation d'état		
	Nom de la variable d'état	Le nom de la variable d'état à changer
	Nouvelle valeur	La nouvelle valeur de la variable d'état à changer
Générale		
	Nom	Le nom de l'objet
	Description	Une description de l'utilisation de cet objet dans le modèle

Le troisième, le Chemin à temps défini, sert à modéliser des trajets dont le temps de parcours est prédéfini. La longueur du Chemin et la vitesse de l'entité n'ont donc plus d'impact. Les propriétés de ce chemin sont similaires à celles du précédent, excepté que cette fois-ci la propriété « Allow passing » (Autoriser le passage) est remplacée par « Travel Time » (temps de parcours).

7.3.2.13 Le Convoyeur

Cas particulier de chemin, d'où son traitement à part, le Convoyeur (Figure 7.15) permet de modéliser aussi bien des systèmes de transferts que des routes, voies ferrées, etc. La vitesse du Convoyeur contrôle la vitesse des entités qui le parcourent, et celui-ci présente l'avantage de pouvoir associer aux entités un volume, un espace qu'elles occupent sur ce Convoyeur.

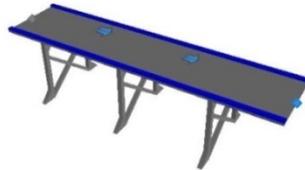


Figure 7.15 : Vue de l'un des modèles SIMIO pour le Convoyeur

De plus l'option « Accumulating » (Accumulation) permet de définir le comportement du Convoyeur, à savoir que si cette propriété est « Vrai », le Convoyeur ne s'arrêtera jamais, tandis que si elle est « Fausse », le Convoyeur s'arrêtera si l'entité est bloquée en sortie du convoyeur.

Le Convoyeur donne aussi accès à l'option « Entity Alignment » (Alignement des entités) donnant la possibilité aux entités de se déplacer comme elles le veulent sur le Convoyeur, mode « Any Location » (pas de distance réglementaire entre deux entités sur le Convoyeur), ou bien de devoir respecter une distance prédéfinie entre elles sur ce chemin, mode « Cell Location ».

7.3.3 Les processus dans SIMIO

En utilisant l'interface « Processes » présentée plus tôt, il est possible de construire et d'éditer les processus. Sur le bandeau gauche de cette interface (Figure 7.2) est présenté l'ensemble des étapes pouvant être intégrées aux processus. Ces étapes s'intègrent de la même façon que les objets dans le modèle, à savoir un « drag and drop » de celles-ci sur les processus.

7.3.3.1 Principes et étapes de bases

Lorsqu'un processus est créé, celui-ci ne comporte qu'un point de départ et un point de fin (Figure 7.16 (a)). En sélectionnant les étapes voulues et en les déposant sur le processus, il est alors obtenu un nouveau processus comme visible sur la Figure 7.16 (b). L'ordre des étapes peut alors être ré agencé comme voulu en les sélectionnant simplement et en les faisant glisser là où celles-ci sont désirées. Dans le volet de gauche, il est possible de préciser l'évènement qui sera déclencheur du processus (« Triggering event ») et la classe du « jeton » qui exécutera le processus (« Token Class Name »).

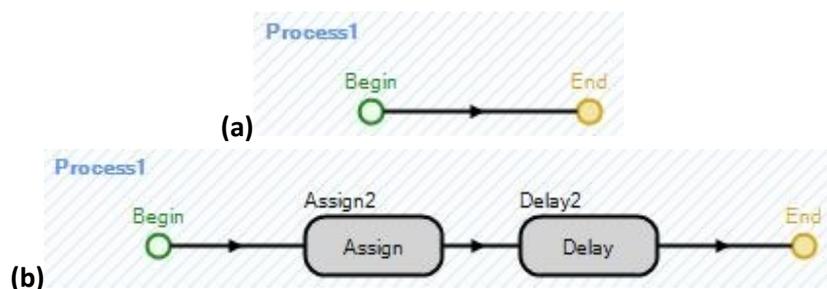


Figure 7.16 : Modèle SIMIO du processus de départ (a) et d'un processus avec étapes (b)

La plupart des étapes des processus sont génériques et peuvent être utilisées par tous les objets de la librairie standard. Cependant, certaines d'entre elles seront spécifiques à un des cinq objets de base tel que l'entité, le transporteur, le nœud ou le lien. De plus, parmi ces nombreuses étapes certaines se révèlent être beaucoup plus utilisées que d'autres. Nous allons maintenant décrire rapidement ces diverses étapes :

- ❖ « Assign » : Utilisée pour assigner une valeur à une ou plusieurs variables d'états ;
- ❖ « Decide » : Utilisée pour sélectionner selon la valeur de l'expression testée, une des deux sorties possibles. L'expression peut être interprétée de manière logique ou à travers des probabilités ;
- ❖ « Delay » : Utilisée pour retarder d'une durée spécifique l'avancée du jeton dans le processus ;
- ❖ « Execute » : Utilisée pour exécuter d'autres processus. Cette étape pourra retenir (si désiré) le jeton de son processus, jusqu'à ce que le processus lancé soit complété ;
- ❖ « Find » : Utilisée pour trouver la valeur d'une expression parmi un ensemble de variables ;
- ❖ « Tally » : Utilisée pour enregistrer une valeur pour un élément « TallyStatistic » précis ;
- ❖ « Create » et « Destroy » : Utilisées pour créer et détruire des objets dynamiques, ce qui inclus les entités et les transporteurs. L'étape « Create » peut créer un nouvel objet ou bien en copier un existant. L'étape « Destroy » peut quant à elle, tous les détruire ;
- ❖ « Fire » et « Wait » : Ces deux étapes apportent une certaine coordination dans la logique d'un objet ou entre plusieurs objets. La première est utilisée pour déclencher un événement associé à un objet, tandis que la seconde retiendra une entité jusqu'à ce qu'un événement spécifique ait été déclenché ;
- ❖ « Seize » et « Release » : Utilisées pour saisir et libérer la capacité d'un objet qui a été défini comme une ressource potentielle ;
- ❖ « Transfer » et « EndTransfer » : Utilisées pour initier ou compléter le transfert d'une entité entre deux objets, deux stations ou encore un objet et un emplacement libre.

7.3.3.2 Exemple de construction d'un processus

Afin de mieux cerner la construction d'un processus, nous allons construire un exemple simple, où il doit être décidé suivant une valeur d'assigner à une variable deux différentes valeurs. Ainsi nous prendrons comme exemple le poids d'une pièce qui déterminera si celle-ci passera sur la machine-outil numéro 1 ou 2. La variable sera « Poids_pièce » et le numéro de la machine-outil sera « Num_MO ». Ainsi si la pièce pèse moins de cinq kilos elle passera sur la machine numéro une, tandis que dans le cas contraire elle passera sur la numéro deux.

Dans un premier temps, il faut créer un nouveau processus (bouton « Create process ») et le nommer « Test_poids » (Figure 7.17 (a)). Cela fait, afin de pouvoir décider de quelle machine-outil il sera nécessaire, il faut utiliser une étape « Decide », qui sera nommée « Poids ? » (Figure 7.17 (b)).



Figure 7.17 : Processus de départ (a) et avec la fonction test « Poids ? » (b)

Afin de paramétrer le test du poids, il sera choisi dans l'onglet « Basic Logic » de l'étape, que le test sera réalisé suivant une condition et non pas des probabilités (« Decide Type » est défini sur « Condition Based »). L'expression de ce test se traduira donc sous la forme « Poids_piece<5 » dans la case « Expression » (Figure 7.18).



Figure 7.18 : Aperçu des propriétés du « Decide » « Poids ? »

Une fois ceci fait, l'étape « Decide » possédant deux sorties prenant les valeurs « Vrai » ou « Faux » (suivant le test effectué), « Num_MO » sera égale à un si le test est vrai, ou à deux dans le cas contraire. Afin d'assigner cette valeur à « Num_MO », il est nécessaire d'utiliser des étapes « Assign » sur chacune des deux sorties comme visible sur la Figure 7.19, que nous nommerons en conséquence « Num_MO=1 » et « Num_MO=2 ».

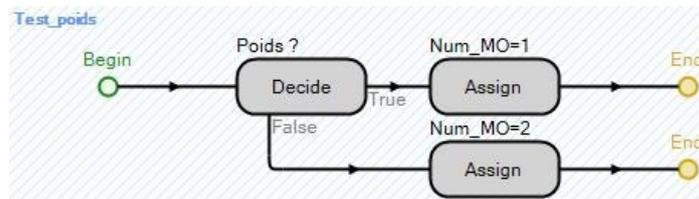


Figure 7.19 : Vue du processus complet avec assignation des valeurs pour Num_MO

Ne reste alors plus qu'à associer les valeurs à la variable adéquate suivant le cas. Ainsi dans la case « State Variable Name » sera placé « Num_MO », tandis que dans « New value » sera mise la valeur un ou deux (Figure 7.20), la construction du processus étant ainsi terminée.



Figure 7.20 : Aperçu des propriétés des « Assign » utilisés

7.3.4 Un exemple simple d'application de SIMIO

Pour terminer cette présentation de SIMIO, nous allons présenter un exemple simple d'application, qui consistera en l'accueil des patients dans les hôpitaux, communément appelé

« Bureau des entrées ». Afin de modéliser cela, commençons par réfléchir aux entités présentes dans notre système. Il y aura bien entendu les patients, mais surtout le personnel du bureau des entrées. Les patients étant les éléments de flux de notre système, ils devront être générés par une « Source » et seront « envoyés » dans les services par le biais d'un « Puits ». Le Tableau 7.17 suivant récapitule les éléments du système.

Tableau 7.17 : Eléments composant le système réel et le modèle SIMIO

Elément réel	Elément SIMIO	Paramètres
Porte d'entrée	Source	L'arrivée des patients suit une loi normale de moyenne 3 minutes et de déviation standard 0.5.
Bureau des entrées	Server	Le temps d'enregistrement des patients suit une loi triangulaire comprise entre 2 et 6 minutes et de moyenne 4 minutes
Patient	ModelEntity	
Personnel bureau des entrées	Worker/Resource	Au nombre de 10
Service	Sink	

Connaissant maintenant la traduction des éléments du système réel sous SIMIO, il est désormais possible de commencer la construction du modèle. Dans un premier temps, nous allons placer les cinq éléments SIMIO utilisés en les renommant dans la « Facility window », afin d'obtenir la Figure 7.21.



Figure 7.21 : Vue de l'ébauche du modèle SIMIO

Les différents éléments seront reliés par des liens de type « Chemin », car nous supposons que le temps de déplacement des patients sera dépendant de leur vitesse de déplacement. Cependant nous ne nous occuperons pas de cet aspect dans cet exemple.

Abordons désormais le paramétrage des différents éléments. Comme énoncé les patients arrivent suivant une loi normale de moyenne 3 minutes. SIMIO possède son propre formalisme en ce qui concerne les lois mathématiques les plus communes. Ce paramètre sera donc traduit sous la forme « Random.Normal(3, 0.5) » et il sera placé dans la case « Interarrival Time » de la « Porte_entree » (Figure 7.22 (a)). De plus, chaque arrivée ne concernera qu'un seul patient d'où le « Entities per Arrival » égal à un.

Ensuite, une loi triangulaire traduit le temps d'enregistrement des patients dans l'hôpital. Ainsi le paramètre « Processing time » du bureau sera « Random.Triangular(2,4,6) ». De plus, étant donné que c'est le personnel qui réalise la tâche d'admission, il est nécessaire de préciser dans l'onglet « Secondary Resources » qu'une ressource de type « Personnel » est nécessaire à la réalisation de la tâche (Figure 7.22 (b)).

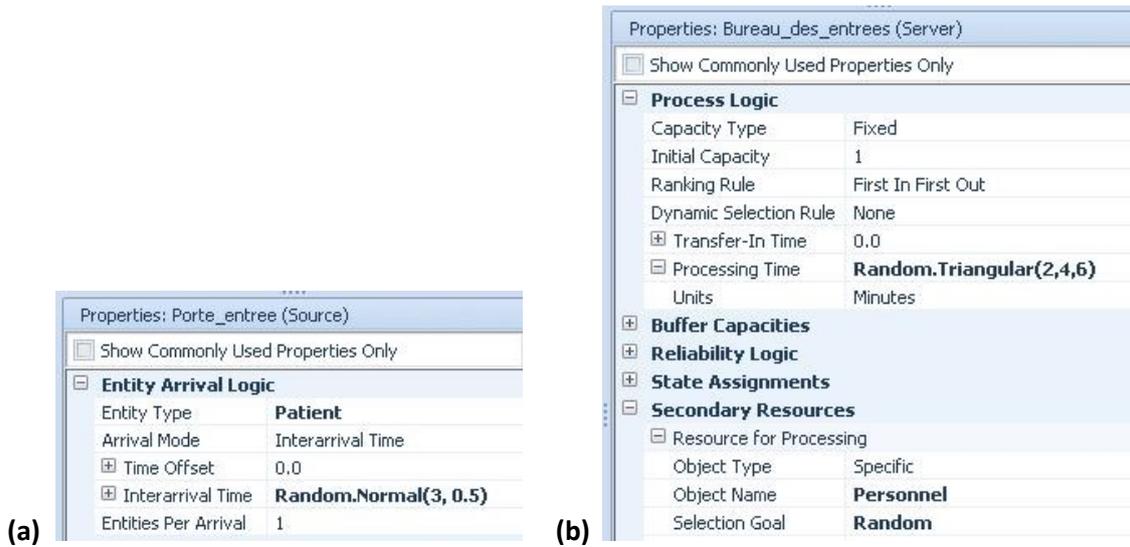


Figure 7.22 : Aperçu des propriétés de « Porte_entree » (a) et « Bureau_des_entrees » (b)

Le fonctionnement de cet exemple est illustré sur la Figure 7.23 suivante, où l’on voit les patients en train d’être enregistrés par le personnel, de couleur verte quand il est sollicité.

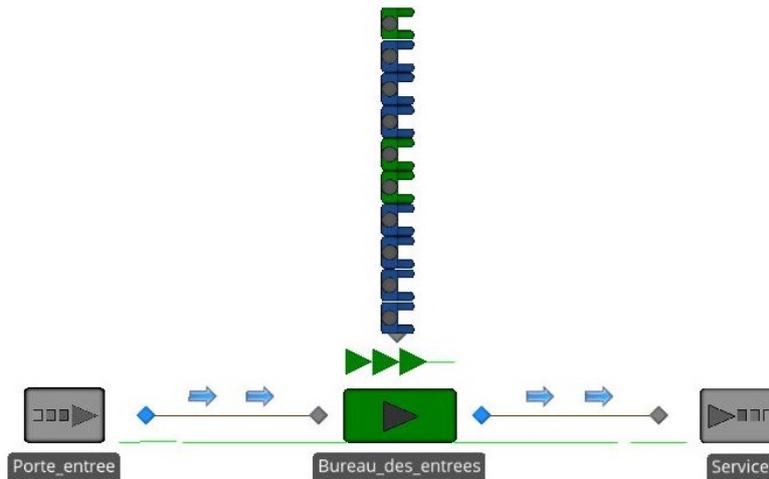


Figure 7.23 : Vue du modèle SIMIO en fonctionnement

7.4 NOTRE APPROCHE POUR PASSER DE BPMN A SIMIO

La partie que nous allons aborder maintenant pourrait faire office à elle seule de travaux conséquents. En effet, l’approche de BPMN consistant à modéliser graphiquement des processus et celle de SIMIO, raisonnement par agent, font que convertir le langage du premier vers le second n’est pas trivial.

Nous allons présenter ici notre méthodologie pour traduire les différents éléments BPMN en SIMIO, que ce soit sous la forme d’un objet, d’un processus et de toute autre chose. Par conséquent, nous aborderons tour à tour les divers éléments du langage BPMN, afin de donner la méthode permettant de réaliser cette conversion. Comme un parallèle au chapitre 4 : BPMN & ASCII, nous

suivrons la même trame que la partie BPMN de ce chapitre, à savoir traiter les activités, puis les évènements, les portes décisionnelles et enfin les acteurs.

Contrairement à notre modèle BPMN qui traite du CDM dans son entièreté, nous ne nous concentrerons ici que sur la partie pharmacie et logistique, l'important pour nous étant l'obtention de la demande ou de la prescription, et non pas comment celle-ci a été envoyée (bien que nous en ferons la différenciation).

7.4.1 Comment traduire une activité

Les activités de BPMN sont de natures diverses et variées ce qui posent un problème de traduction sous SIMIO. En effet, bien que chaque tâche puisse se traduire par un objet « Serveur », nombre d'entre elles se verraient être inutiles dans une simulation. Il va donc être impératif de filtrer, trier les nombreuses activités et tâches composant la partie pharmacie et logistique du modèle BPMN.

De base la pharmacie comporte 36 activités, sous-processus ou groupe d'activités. Afin de réduire ce nombre pour ne pas obtenir 36 serveurs plusieurs règles vont devoir être suivies :

- ❖ **Supprimer les activités « fantômes » :** Certaines activités se révèlent problématiques lors de leur exécution. En effet, elles ne peuvent pas être quantifiables en termes de temps ou d'impact sur le système. L'exemple le plus flagrant est l'activité « Attente de réponse du médecin ». Il n'y a ni donnée officielle, ni relevé pour estimer le temps de retour de la réponse et l'impact de celle-ci. Cette activité ne sera donc pas modélisée.
- ❖ **Supprimer les activités transparentes :** Bien que pouvant être assimilées à celles ci-dessus, nous les différencieront car elles seront généralement liées à la transmission d'informations. Parmi elles se trouvent majoritairement l'envoi de messages, mais il sera possible d'y associer toutes activités ponctuelles, ne nécessitant pas de temps pour être exécutées (par exemple une mise à jour de données).
- ❖ **Regrouper les activités :** Etant donné que la construction d'un modèle générique est notre objectif, il sera nécessaire de regrouper certaines activités entre elles afin d'obtenir des activités mésoscopiques. De même des activités de même nature où seul l'élément traité diffère pourront être fusionnées.

En appliquant les deux premières règles ci-dessus, le nombre d'activités est déjà réduit de 12 (Figure 7.24 et 7.25).

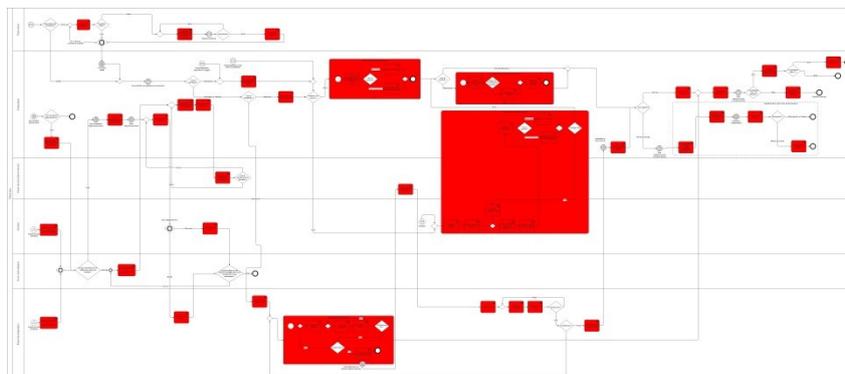


Figure 7.24 : Activités utiles avant réduction

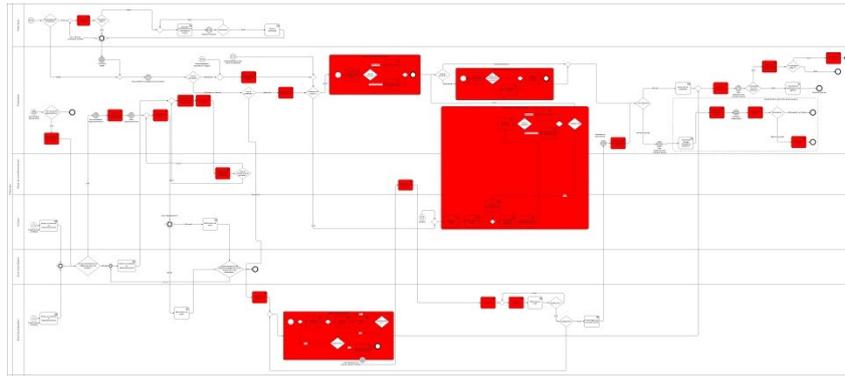


Figure 7.25 : Activités utiles avant après réduction

Maintenant essayons d’appliquer la troisième règle. En se référant à la vue par bloc présentée dans le chapitre précédent, la pharmacie était représentée ainsi (Figure 7.26).



Figure 7.26 : Vue par bloc de la pharmacie

En découpant plus finement ces quatre blocs mésoscopiques nous obtenons la Figure 7.27.

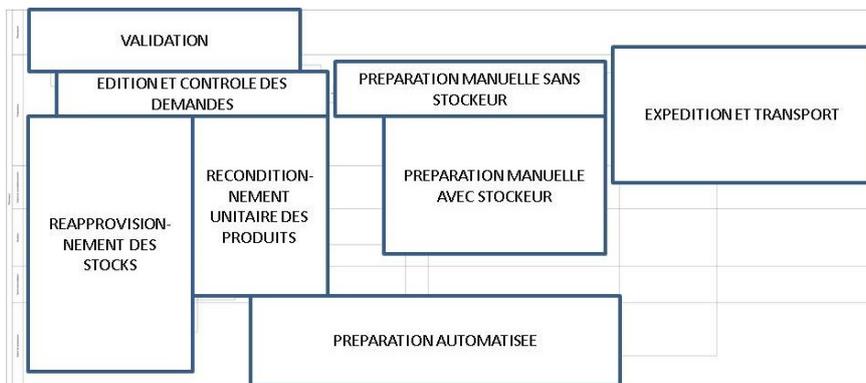


Figure 7.27 : Vue affinée par bloc de la pharmacie

En confrontant ce résultat avec celui de la Figure 7.25, il est possible de commencer à en déduire quelles seront les activités qui vont se convertir en « Serveur ». Raisonnons bloc par bloc :

- ❖ Validation : Pour celui-ci le bilan sera rapide, étant donné que seule l’activité « Analyse des prescriptions » restera ;
- ❖ Edition et contrôle des demandes : Même constat cette fois-ci. Seules les activités liées à l’édition et le contrôle (pour les divers types de demandes) sont encore présentes. Cela donne donc une unique activité « Editer et contrôler les demandes » ;

- ❖ Préparation sans stockeur : déjà composée de sous-processus étendus, cette partie ne comporte donc que deux activités. La première est « Préparer manuellement une demande de service » et la seconde « Préparer manuellement une demande nominative » ;
- ❖ Préparation avec stockeur : les activités sont ici identiques à celles de la préparation sans stockeur. Nous verrons donc dans ce qui suit comment contourner cette difficulté ;
- ❖ Préparation automatisée : les activités ici utilisées seront celles liées à la préparation automatisée des traitements, et celles liées à l'utilisation du robot pour la préparation de demande ponctuelle urgente ou hors cycle. Au final, il y aura quatre activités : « Préparer les traitements nominatifs », « Rentrer une demande », « Préparer une demande spéciale » et « Décharger le robot » ;
- ❖ Réapprovisionnement des stocks : de nouveau, nous n'avons que deux activités présentes. Il s'agit de l'activité « Editer les besoins des stocks » et « Réapprovisionner les stocks ». Nous ne regrouperons pas ces deux activités en un seul serveur car entre ces deux peut avoir lieu les activités liées au conditionnement unitaire ;
- ❖ Reconditionnement unitaire des produits : les activités ayant lieu ici sont liées aux différentes actions pour reconditionner unitairement des produits. Il est donc question de « Déconditionner les produits », « Charger le robot de reconditionnement » et enfin « Reconditionner les produits ». La première activité pouvant avoir lieu indépendamment des deux suivantes nous la garderons donc comme un serveur unique. Au final nous aurons un serveur « Déconditionner les produits » et un « Reconditionner les produits » ;
- ❖ Expédition et transport : pour terminer, ce bloc sera particulier plusieurs cas de figures devant être envisagés. En effet, selon le type de l'armoire et selon si le préparateur livre ou pas la commande, les actions changeront. Globalement, il sera retrouvé comme action dans ce bloc « Regrouper les demandes de l'unité de soins », « Transmettre la commande au service logistique », « Transporter la commande jusqu'à l'unité de soins », « Réapprovisionner la/les armoire(s) ».

A la fin de ces premières étapes, nous nous retrouvons avec 13 activités pouvant être traduites sous la forme d'un serveur dans SIMIO. Cependant, une dernière question doit être posée avant de poursuivre celle-ci pouvant permettre de simplifier encore le futur modèle :

Est-il utile et pertinent de créer deux serveurs pour des activités similaires ?

Comment traduire et appliquer exactement cette interrogation sur nos activités ? Deux activités que nous comprendrons comme similaire, seront des activités dont seuls les paramètres varieront, ces paramètres pouvant être des temps d'application, les produits traités, qui les réalisent, etc. Utilisons deux exemples pour que cela soit plus parlant.

Le premier cas est celui des activités de préparations manuelles avec et sans stockeur. Les activités sont identiques étant donné que les demandes/prescriptions traitées contiennent les mêmes éléments. En effet, seul le nombre d'éléments va varier entre une demande de service et une prescription. Il n'est donc pas utile de faire un serveur pour les préparations de service et un autre pour les préparations nominatives. Par contre, la différenciation ne se fera pas sur l'activité elle-même, mais sur son lieu d'exécution. L'utilisation d'un stockeur, comparativement à l'utilisation d'un

stock en rayonnage, impactera le temps de préparation des demandes. Par conséquent, il faudra différencier préparation en stockeur et préparation avec stock, d'où la création de deux serveurs correspondant. Pour ce premier cas, la réponse à la question précédemment posée « *Est-il utile et pertinent de créer deux serveurs pour des activités similaires ?* » est donc ici : oui.

Le second cas concerne le(s) robot(s) de dispensation, c'est-à-dire la préparation automatisée. Les activités entreprises dans le cadre de la préparation d'une demande urgente ou hors cycle ne se différencient que par l'action du préparateur sur le robot ou l'ordinateur de contrôle. Celui-ci rentre manuellement ou valide la nouvelle demande à réaliser. Ne peut-on pas assimiler ce temps de saisie au temps de préparation de la demande ? Nous supposons que cela sera possible, ne laissant dès lors que deux activités identiques : « Préparer les traitements nominatifs » et « Préparer une demande spéciale », l'unique différence portant sur leur temps de réalisation. Un unique serveur « Robot de dispensation » se révélant alors suffisant pour simuler ces activités.

Il y aura donc une réflexion au cas par cas à mener sur ces activités, afin de comprendre si oui ou non la création de deux serveurs est nécessaire pour en traduire le fonctionnement. Cependant en nous appuyant sur la logique interne de SIMIO, il est d'ores et déjà possible d'y apporter des éléments de réponse. **Les serveurs dans SIMIO étant des objets fixes, si deux activités ont des lieux d'exécution différents, alors ces deux activités seront traduites par des serveurs différents.**

Au final les serveurs devant être, pour l'heure, créés sont :

- ❖ Analyser les prescriptions ;
- ❖ Editer et contrôler les demandes ;
- ❖ Préparer une demande avec un stock standard ;
- ❖ Préparer une demande à l'aide d'un stockeur mécanique ;
- ❖ Préparer les demandes nominatives avec un robot de dispensation ;
- ❖ Décharger le robot de dispensation ;
- ❖ Déconditionner les produits ;
- ❖ Reconditionner les produits ;
- ❖ Transmettre les demandes à la logistique ;
- ❖ Transporter les demandes jusqu'à l'unité de soins ;
- ❖ Ranger les demandes dans les armoires.

7.4.2 Comment traduire les évènements

Maintenant que la traduction des activités de BPMN sous SIMIO est quasiment terminée, seules quelques modifications pouvant encore être réalisées, penchons-nous sur les divers évènements composant notre modèle BPMN.

7.4.2.1 Les évènements de débuts

En toute logique, les évènements de début seront les déclencheurs des activités sous SIMIO. Mais comment les traduire correctement sous ce dernier. Les évènements de début sous BPMN, initient les processus de celui-ci et créent l'équivalent de jetons qui vont parcourir les processus. Ils

peuvent être donc vus comme des sources de jetons. Cependant, tous ces évènements n'ont pas pour autant une signification « physique » dans le sens où le jeton qu'ils créaient ne représente pas nécessairement un élément tangible (tel qu'un message, une entité, etc.). C'est notamment le cas pour les évènements de début de type « timer », « signal », « erreur », etc.

La première règle pouvant être établie concernant les évènements est donc que : **Si un jeton produit par un évènement de début représente un élément manipulable (dans le sens qui est utilisé) par les acteurs du modèle, alors cet évènement sera traduit par une source sous SIMIO. Dans le cas contraire, ces derniers se verront intégrés dans les processus de SIMIO et cela sous diverses formes selon qu'il s'agisse de messages, de timers, d'erreurs, ou de signaux.**

En appliquant cela à la pharmacie modélisée sous BPMN, nous constatons que tous les évènements, à l'exception d'un seul (émission d'un signal quand les stocks sont insuffisants), produisent des éléments manipulables. Ainsi les évènements de début suivants se traduiront sous formes de sources SIMIO :

- ❖ Nouvelle prescription ;
- ❖ Nouvelle demande globale disponible ;
- ❖ Nouvelle demande globalisée disponible sur le logiciel ;
- ❖ Produits en stock insuffisant.

La traduction des trois premières semble évidente, des sources créant des demandes. La quatrième créera elle aussi des demandes étant donné qu'une alerte de stocks insuffisants peut être interprétée en tant qu'une demande de réapprovisionnement urgente. Ainsi ces quatre évènements se traduiront par quatre sources de demandes.

Mais est-ce réellement utile ?

Nous verrons dans le chapitre suivant qu'il ne sera pas nécessaire de créer quatre sources pour traduire ces quatre évènements. En effet, bien qu'une source puisse créer qu'un seul type d'élément, les paramètres de cet élément pourront varier.

Finalement, revenons-en au seul évènement de début ne générant pas directement une demande. Il s'agit d'un évènement de type timer, associé à une action de contrôle des stocks de produits en vue d'un réapprovisionnement. Chaque jour, à une heure précise aura lieu cette action, qui résultera en la création d'un certain nombre de demandes de réapprovisionnement des stocks. Ces demandes ne seront donc créées qu'au moment de l'exécution de ce timer.

Comme nous l'avons énoncé précédemment, dans ce cas, le timer sera intégré à un processus SIMIO (ce qui sera majoritairement le cas pour ce type d'évènement), et celui-ci se traduira par l'utilisation d'un « Delay » permettant de l'exécuter à l'heure souhaitée, et qui sera accompagné par un « Create » obligeant la création de demandes de réapprovisionnement.

Il est difficile d'énoncer des règles permettant de traduire les évènements sous forme de processus SIMIO, cette traduction dépendant des compétences de l'utilisateur et de ce qu'il en attend. En effet, un même évènement pourra se traduire par un processus pour un utilisateur, ou par l'utilisation d'une combinaison de fonctions, d'objets pour le second. Cet aspect reste donc une des difficultés majeures de la traduction des éléments BPMN sous SIMIO, une généralisation des règles de traductions étant compliquée.

7.4.2.2 Les évènements intermédiaires ou de fin

A l'instar des évènements de début, ceux-ci pourront se traduire eux aussi sous diverses formes mais uniquement dans les processus SIMIO. Nous allons donc tenter de donner des pistes pour la traduction de ces évènements sous SIMIO :

- ❖ Les timers : comme dit ci-dessus, ils ont un impact en termes de durée ou d'horaire, ce qui peut se traduire de différentes manières sous SIMIO. Ainsi, il est soit possible de passer par les plannings intégrés pour gérer ces évènements (par exemple chaque jour à une certaine heure le travail cesse pendant une heure), soit d'utiliser comme nous l'avons présenté ci-dessus des blocs « Delay ». Cela dépendra donc du choix de l'utilisateur ;
- ❖ Les évènements conditionnels : ces évènements ont la particularité d'être une réaction à un changement de condition, ou une règle. Ainsi ils opèrent plus dans une optique de tout ou rien, avec l'autorisation pour le jeton parcourant le processus de continuer son parcours dans le cas où la condition est vérifiée. Des exemples se trouvent dans notre processus :
 - Evènement « Demande reçue » : le jeton n'avancera pas tant que la demande ne sera pas reçue. Cela se traduit sous SIMIO par le fait que la demande ne sera pas arrivée à destination. La suite du flux ne pourra donc s'exécuter ;
 - Evènement « Médicament reçue » : même idée que ci-dessus pour les médicaments cette fois ;
 - Evènement « Prescription validée » : la préparation de la demande nominative par le biais du serveur SIMIO adéquat, ne pourra commencer tant que cette condition ne sera pas vérifiée ;
 - Evènement « Préparateur devant armoire » : si celui-ci n'est pas présent devant l'armoire, l'activité associée ne peut avoir lieu.
- ❖ Les évènements messages : étant une conséquence de la transmission d'informations dans le processus, ceux-ci n'apparaîtront pas dans notre modèle, car nous ne nous occupons pas de leur modélisation. Dans le cas contraire, ils apparaîtraient très certainement eux aussi sous forme de processus ;
- ❖ Les évènements erreurs : ces évènements sous BPMN agissent comme des interrupteurs de processus. Ils peuvent modéliser une panne, un contrôle défaillant, un objet manquant, etc. En tout état de cause ils interrompent le déroulement du processus, chose pouvant une nouvelle fois se traduire de différentes manières sous SIMIO suivant le type d'erreur :
 - Erreur de type panne/incident technique : il existe sous SIMIO une option « Reliability logic » pour les serveurs. Celle-ci permet de configurer des pannes sur les serveurs suivant différentes logiques. Il est aussi possible d'utiliser un processus avec un bloc « Fail » ;
 - Erreur de type contrôle défaillant : généralement traduite sous formes de pourcentages, une telle erreur peut être représentée sous SIMIO par des nœuds dont la condition de sortie est « By link weight », chaque chemin possible étant alors associé à un pourcentage précis ;

- Erreur de type produit manquant : difficilement interprétable, le moyen le plus efficace serait d'utiliser un processus add-on permettant de lancer un processus précis lors de l'exécution (processing) de l'activité liée ;
- Erreur de type rupture : cette erreur peut être vu comme une panne, dans le sens où si un stock est insuffisant alors une panne en résulte. Il serait dès lors possible de la modéliser comme la première catégorie d'erreur.

En conclusion, comme pour les activités, la traduction des événements sous SIMIO, sera elle aussi dépendante de l'objectif visé, des capacités de l'utilisateur. Cette constatation sera d'ailleurs applicable pour la prochaine catégorie d'éléments BPMN, à savoir les portes décisionnelles.

7.4.3 Comment traduire les portes décisionnelles

Nouvelle difficulté de traduction, les portes décisionnelles vont majoritairement pouvoir être classées dans deux catégories : les portes interrogeant sur la réalisation d'une activité et les portes portant sur les caractéristiques du jeton.

Bien qu'à première vue, elles ne servent que pour orienter la progression du jeton dans le processus, les questionnements présents dans ces portes vont représenter autant de paramètres que nécessaires pour caractériser l'objet de flux. Il est donc nécessaire de dresser la liste des différents paramètres utilisés dans le processus (Tableau 7.18).

Tableau 7.18 : Question et réponses possibles durant le processus de préparation

Question / Paramètre	Réponse 1	Réponse 2	Réponse 3	Réponse 4
La prescription est analysée ?	Oui	Non		
Type de distribution de la demande/prescription	Globalisée	Nominative	Globale	Partielle
Type de préparation de la demande/prescription	Manuelle	Automatisée		
Utilisation d'un stockeur pour préparer la demande/prescription	Oui	Non		
Type d'armoire utilisée pour stocker la demande/prescription	Normale	Plein/vide	Automatisée	
La demande/prescription est-elle transportée par le préparateur ?	Oui	Non		
La demande/prescription est-elle livrée par le préparateur ?	Oui	Non		

Ce premier tableau concerne les paramètres qui pourront être liés aux différentes demandes et prescriptions. Chaque paramètre déterminera donc le parcours de chacune d'entre elles dans le modèle de simulation.

En plus des portes concernant les demandes/prescriptions, il y a aussi des portes concernant le réapprovisionnement des stocks, amenant elles aussi certains paramètres (Tableau 7.19) mais concernant cette fois-ci la création de demandes de réapprovisionnement et de demandes de reconditionnement, qui seront donc sur un circuit distinct des demandes et prescriptions ci-dessus.

Tableau 7.19 : Question et réponses possibles durant le processus de réapprovisionnement

Question / Paramètre	Réponse 1	Réponse 2
Des stocks doivent-ils être réapprovisionnés ?	Oui	Non
Les stocks intermédiaires seront-ils suffisant pour ce rapprovisionnement ?	Oui	Non
L'état des stocks est-il satisfaisant ?	Oui	Non

Au final, il est obtenu une liste de paramètres utilisables par le développeur pour concevoir le modèle de simulation. Une nouvelle problématique vient cependant ralentir ce développement et elle concerne le type de modèle voulu, à savoir dédié ou générique. En effet, suivant l'objectif, les paramètres identifiés ne seront pas implémentés de la même manière. Tandis que pour un modèle générique ces paramètres devront se « trouver » sur l'objet parcourant le flux, pour un modèle dédié, ces paramètres seront très certainement à la base du modèle lui-même. Prenons un exemple. Considérons l'implantation d'un service fonctionnant en mode globalisé dans un modèle générique et dans un modèle dédié. Tandis que pour le premier ce type de fonctionnement ne sera que l'un parmi les nombreux possibles (globale, nominatif informatisé ou papier, etc.), il sera au cœur de la logique du second, celui-ci tournant autour du service fonctionnant en globalisé et ne nécessitant donc pas la prise en compte des autres types de dispensations. Le modèle obtenu pour le second cas sera donc plus simple dans la logique et l'objet de flux sera systématiquement de type globalisé.

En ce qui concerne les portes décisionnelles, il est donc nécessaire de connaître en amont le type de modèle qui sera développé afin de prendre en compte dès que possible cette spécificité.

7.4.4 Comment traduire les acteurs

Abordons désormais la dernière partie de la conversion d'un modèle BPMN en un modèle SIMIO. Il s'agit certainement là, de la partie la plus simple. En effet, les divers acteurs et « pools » de BPMN se transforment en travailleur, transporteur, ressource, mais aussi en serveur. Ce sont donc les caractéristiques de chacun de ses acteurs qui vont décider de quels types il sera sous SIMIO. Ces caractéristiques sont déterminées par les questions suivantes :

- ❖ L'acteur se déplace-t-il ?
- ❖ L'acteur est-il soumis aux pannes ?
- ❖ L'acteur transporte-t-il des éléments ?

Les réponses apportées à ces trois questions peuvent être exprimées sous la forme de la matrice cubique suivante (Figure 7.28).

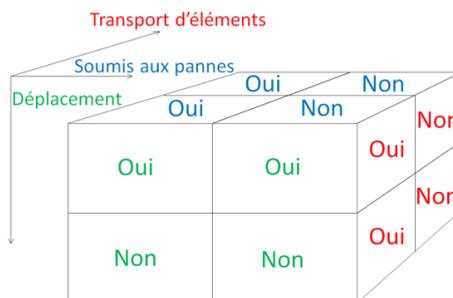


Figure 7.28 : Matrice cubique de sélection du type d'acteur sous SIMIO

Ainsi en décomposant cette matrice telle que visible sur la Figure 7.29, les acteurs (les noms étant ceux du logiciel donc en anglais) obtenus dans chaque cas sont :

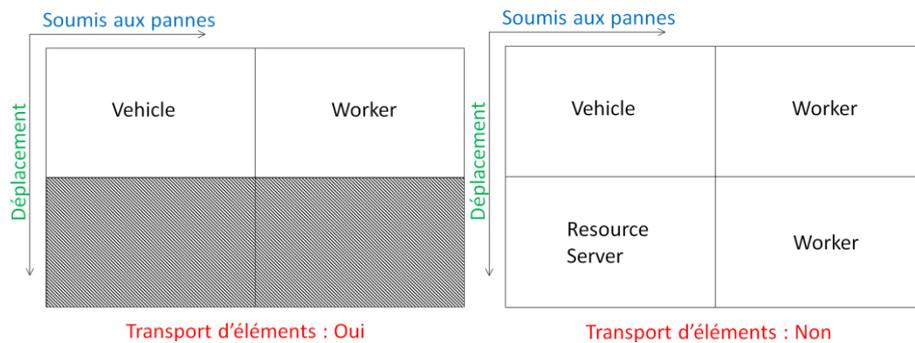


Figure 7.29 : Choix associés à la matrice cubique de la Figure 7.28

Ainsi en appliquant cette logique à nos différents acteurs nous obtenons :

- ❖ Pharmacien : Celui-ci réalise pour le moment une unique tâche (la validation), ne transporte pas et ne se déplace. Il pourrait donc être une « Resource », un « Worker » ou un « Server ». Cependant dans une optique d'évolution, où le pharmacien pourrait être amené à se déplacer (sans transporter), nous privilégierons le type « Worker » pour celui-ci ;
- ❖ Préparateur : Bien que l'aspect spatial ne soit pas intégré dans notre modèle, celui-ci devant être générique, comme nous l'avons déjà énoncé, le préparateur réalisera des tâches à divers endroits dans le modèle. Bien qu'il pourrait donc être une « Resource » ou un « Worker », mais pas un « Vehicle » n'étant pas soumis aux pannes, nous choisirons de nouveau le second afin de ne pas fermer la porte à de futures évolutions ;
- ❖ Robot de reconditionnement : Ne se déplaçant pas et pouvant être soumis aux pannes, le choix de l'acteur est ici évident, à savoir un « Server », cette entité sollicitant une aide extérieure pour réaliser son travail (un préparateur) ;
- ❖ Robot de dispensation : Même chose que pour le robot ci-dessus, d'où le choix d'un « Server » ;
- ❖ Stockeur : De nouveau, les conclusions sont les mêmes que pour les deux acteurs ci-dessus, donc celui-ci sera traduit par un « Server » ;
- ❖ Agent logistique : Cet acteur transportant jusqu'au service, le choix peut être soit un « Worker » soit un « Vehicle ». Si le raisonnement se porte sur l'agent, alors dans ce cas il sera choisi un « Worker » car pas de panne. S'il est préféré raisonner en termes de moyen de transport, alors il sera choisi un « Vehicle » pour considérer le cas des pannes. Finalement le choix se portera pour nous vers un « Vehicle », pour une nouvelle fois ne pas se fermer des portes.

7.5 CONCLUSION

Lors de ce chapitre, nous avons pu appréhender la relative complexité des objets composant le logiciel SIMIO, ainsi que présenter des éléments encadrants la conception du modèle. C'est pourquoi, dans un premier temps, nous nous sommes intéressés brièvement aux leviers de

simulation et aux critères de performance. Nous avons alors présenté un nombre, non exhaustif, de critères portant sur de nombreux éléments ainsi que les divers leviers imposés par le CHU, leviers qui devront être impérativement intégrés, d'une manière ou d'une autre, dans le logiciel de simulation.

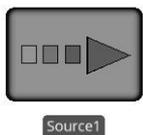
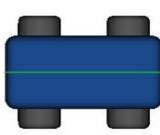
Ce chapitre s'est poursuivi en toute logique par la présentation des objets composant SIMIO. En détaillant tout d'abord les fenêtres et interfaces spécifiques à ce logiciel, nous enchaînons avec les classes d'objets utilisées dans SIMIO, pour lesquelles nous listons les caractéristiques principales et les attributs, que nous présentons dans leur entièreté (Source, Puits, Serveur, Travailleur, Chemin, etc.). Nous abordons ensuite l'une des fonctionnalités les plus intéressantes de SIMIO, à savoir la possibilité de définir des « processus » pour agir sur le comportement et l'évolution des objets précédemment définis. Par conséquent, nous détaillons les principes de la conception de ces « processus » et présentons les étapes de base qui les composent. Afin d'appréhender au mieux leur fonctionnement, nous donnons un exemple de construction d'un processus portant sur la répartition de pièces sur des machines-outils spécifiques suivant leur poids. Puis, nous terminons la présentation de SIMIO en concevant de nouveau un exemple, portant cette fois-ci sur la construction d'un modèle de simulation, qui traite de l'accueil des patients à l'hôpital (le bureau des entrées).

Ce chapitre se conclut par une explication et une présentation de règles que nous avons mises en place et ayant pour objectifs d'encadrer et de faciliter la traduction des processus conçus sous BPMN en un modèle SIMIO. Ainsi nous abordons dans cette partie comment traduire une activité, un évènement, une porte décisionnelle et un acteur.

Bien que complexe et technique, ce chapitre a permis de présenter, autant que possible, le logiciel SIMIO et son contenu, même si seule l'expérience de la manipulation de ce logiciel peut permettre à un utilisateur d'en comprendre tous les rouages.

Le Tableau 7.20 récapitule les règles de passage de BPMN à SIMIO.

Tableau 7.20 : Règles de passage principales de BPMN à SIMIO

Élément BPMN	Equivalent SIMIO
Activité « active »	
Evènement de début (générateur d'un objet de flux)	
Autre évènement :	<p data-bbox="715 712 783 741">Timer</p>  <p data-bbox="815 680 879 703">Delay1</p> <p data-bbox="874 725 916 748">Delay</p> <p data-bbox="995 712 1023 741">Ou</p>  <p data-bbox="1038 725 1102 748">Timer</p> <p data-bbox="1129 712 1326 741">lié à un processus</p>
Erreur	 <p data-bbox="815 792 879 815">Fail1</p> <p data-bbox="874 837 916 860">Fail</p> <p data-bbox="1011 837 1038 860">+</p>  <p data-bbox="1059 860 1139 882">Failure</p> <p data-bbox="810 943 1406 1003">Ou pour un serveur, une station de travail, un séparateur, un combineur :</p> <div data-bbox="810 1016 1174 1106" style="border: 1px solid #ccc; padding: 2px;"> <p><input type="checkbox"/> Reliability Logic</p> <p>Failure Type No Failures</p> <p><input checked="" type="checkbox"/> State Assignments</p> </div> <p data-bbox="810 1122 1174 1151">Ou pour un connecteur, chemin :</p> <div data-bbox="810 1167 1254 1249" style="border: 1px solid #ccc; padding: 2px;"> <p><input type="checkbox"/> Routing Logic</p> <p>Outbound Link ... Any</p> <p>Outbound Link ... By Link Weight</p> </div>
Porte décisionnelle	Utilisation couplée de processus et de paramètres de l'objet de flux
Acteur immobile (avec panne)	 <p data-bbox="954 1442 1007 1464">Resource1</p> <p data-bbox="1050 1442 1077 1464">OU</p>  <p data-bbox="1177 1442 1230 1464">Server1</p>
Acteur immobile (sans panne)	 <p data-bbox="1059 1599 1139 1621">Worker1</p>
Acteur mobile (sans panne)	 <p data-bbox="1059 1756 1139 1778">Worker1</p>
Acteur mobile (avec panne)	 <p data-bbox="1075 1935 1139 1957">Vehicle1</p>

VIII

CONCEPTION DU MODÈLE DE SIMULATION

RESUME DU CHAPITRE 8

De nombreux auteurs ont souligné et soulignent encore de nos jours que l'un des problèmes majeurs contribuant à l'échec des projets visant à la réorganisation des processus, est le manque d'outils permettant d'évaluer l'impact et la pertinence des solutions proposées.

Ainsi, et afin de ne pas répéter cette erreur maintes fois commise, nous allons à travers ce chapitre présenter le modèle de simulation que nous avons conçu avec le logiciel SIMIO.

Ce chapitre peut se révéler déroutant et quelque peu indigeste de par sa taille et son contenu très technique. Cependant il représente là l'essence même de notre travail et certainement ce dans quoi nous nous sommes le plus investis.

Ainsi, seule la première et les deux dernières sous-sections pourront se révéler intéressantes pour les pharmaciens, le reste étant plus à destination d'un public d'informaticiens. Tandis que la première sous-section présente une vue globale du modèle, l'avant-dernière s'intéresse et détaille les tables de données permettant aux pharmaciens de ne pas se préoccuper du modèle, mais uniquement de la mise en forme des données et de l'exploitation des résultats obtenus par le biais de celui-ci. Enfin, la dernière sous-section présente l'interface permettant aux pharmaciens de remplir ces tables.

SOMMAIRE DU CHAPITRE

8.1	INTRODUCTION	227
8.2	LE MODELE DE SIMULATION DANS SA GLOBALITE	228
8.2.1	Vue globale.....	228
8.2.2	Les éléments du modèle.....	228
8.2.3	Les parcours possibles.....	229
8.3	LES ELEMENTS COMPOSANT LE MODELE	230
8.3.1	Les travailleurs.....	230
8.3.2	Les demandes.....	231
8.3.3	Le Service source.....	233
8.3.4	Les demandes urgentes et leur génération.....	237
8.3.5	La séparation des prescriptions d'une demande	238
8.3.6	La décomposition des prescriptions.....	241
8.3.7	La validation des prescriptions.....	245
8.3.8	La distribution des prescriptions.....	249
8.3.9	La préparation robotisée.....	255
8.3.10	Edition et contrôle des demandes	258
8.3.11	La décomposition des demandes.....	262
8.3.12	La préparation des demandes.....	265
8.3.13	Réunion, transport et envoi	266
8.3.14	Le réapprovisionnement des stocks.....	269
8.4	LES TABLES DE DONNEES ET LES ETATS	272
8.4.1	Les états.....	273
8.4.2	Pourquoi de telles tables ?	273
8.4.3	Les tables.....	274
8.5	Interface de saisie des données.....	280
8.6	CONCLUSION	283

8.1 INTRODUCTION

Possédant désormais la connaissance nécessaire à l'utilisation du logiciel SIMIO, et connaissant les règles permettant le passage d'un modèle BPMN à un modèle de simulation sous ce logiciel, il est alors possible pour nous de commencer la construction de ce dernier, en vue de tester les solutions que nous proposons par le biais de l'utilisation de notre démarche et de nos outils précédemment conçus.

Nous entrons là dans la partie technique du sujet. Cette partie, présentée dans les sous-sections 8.2 et 8.3, intéressera les personnes ayant besoin de programmer, de construire un modèle similaire. C'est pourquoi cette partie sera très dense et technique, faite par et pour des informaticiens.

Dans ces sous-sections, les nombreux éléments composant le modèle seront détaillés en suivant une logique de « zoom ». Ainsi le modèle sera présenté dans un premier temps d'un point de vue global, afin de mettre en évidence les parcours possibles pour les entités de flux. Puis, nous aborderons les nombreux éléments intégrés dans ce modèle, qui sont de natures diverses et variées : acteur, zone de travail, source, etc. où il sera détaillé pour chacun de ces éléments :

- ❖ Les propriétés qui lui sont liées, et qui caractérisent son fonctionnement ;
- ❖ Les données nécessaires à son paramétrage et à son exécution lors de la simulation ;
- ❖ Et enfin les processus, si existants, qui lui sont attachés et qui régissent son comportement.

Cette présentation des éléments du modèle sera faite suivant la logique de parcours du processus du CDM, c'est-à-dire, que nous commencerons par la génération des demandes, puis leur préparation, pour terminer par leur envoi, et cela en suivant deux lignes : une concernant les prescriptions (demandes nominatives) et l'autre concernant les demandes non-nominatives. Deux cas particuliers de demandes seront aussi abordés, à savoir celui des demandes urgentes et des demandes de réapprovisionnement, pour auxquelles deux sous-parties seront consacrées.

Cependant, il ne faut pas omettre que ce modèle de simulation doit être générique et **manipulable** par les pharmaciens. C'est pourquoi l'une de nos contributions est d'avoir mis en place un mécanisme de tables (présenté en sous-section 8.4) qui permet de s'affranchir du paramétrage direct du modèle de simulation. Les pharmaciens sont alors capables d'utiliser pleinement ce modèle sans avoir à le manipuler et donc sans devoir s'intéresser aux sous-sections 8.2 et 8.3.

La sous-section 8.4 concernera donc les nombreuses tables de données aussi bien utiles à la caractérisation des nombreux éléments du modèle, qu'à son paramétrage pour la simulation. Chacune des tables sera présentée, en énonçant son rôle et les variables la composant, afin de souligner le caractère unique de chacune d'elles.

Enfin dans la sous-section 8.5, il sera détaillé l'interface permettant de fournir à SIMIO les informations utiles et nécessaires aux simulations.

8.2 LE MODELE DE SIMULATION DANS SA GLOBALITE

Le modèle de simulation de SIMIO comporte comme nous allons le voir de nombreux éléments de diverses natures, la plupart d'entre eux ayant déjà été présentés dans le chapitre précédent. Nous étudierons dans un premier temps la vue globale que l'utilisateur lambda pourra avoir du modèle avant de détailler rapidement les éléments qui composent le modèle et les divers parcours que les entités peuvent être amenées à suivre dans ce modèle.

8.2.1 Vue globale

Notre modèle ayant pour objectif d'être générique, ce dernier doit comporter tous les éléments majeurs que nous avons identifié dans BPMN. La vue de la Figure 8.1 permet donc de visualiser les différents pôles composant ce modèle.

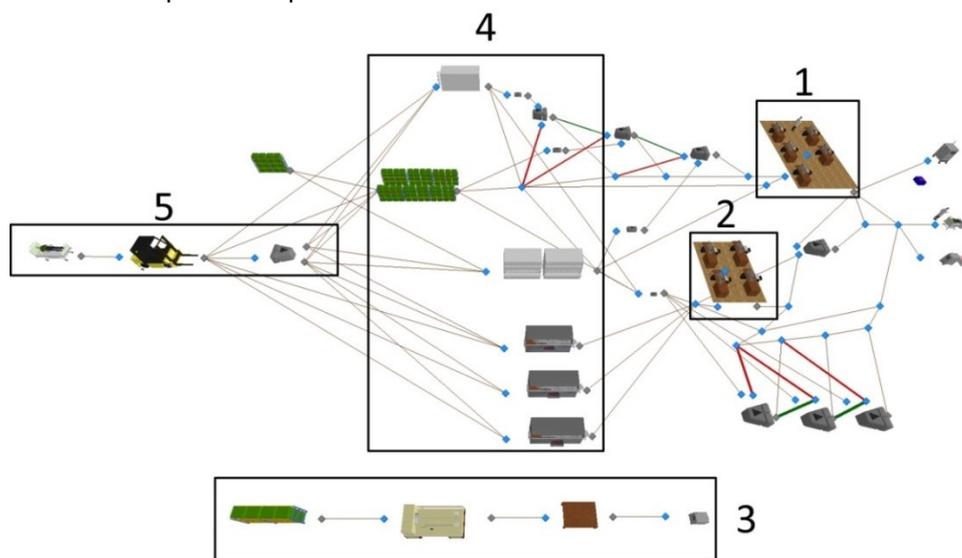


Figure 8.1 : Vue globale avec les blocs du modèle BPMN

La vue précédente permet de retrouver les différents blocs qui étaient présents dans le modèle BPMN. En effet, le bloc 1 représente l'édition et le contrôle des demandes, le bloc 2 la validation des prescriptions, le numéro 3 le réapprovisionnement des stocks, le 4 la préparation des demandes et enfin le 5 l'expédition et le transport.

8.2.2 Les éléments du modèle

La Figure 8.2 présente une vue plus détaillée des blocs 3,4 et 5. Sur cette figure, le bloc 3 qui concerne le réapprovisionnement des stocks se trouve donc décomposé en trois sous-parties, la première générant les éventuelles demandes de réapprovisionnement, la seconde s'occupant du déconditionnement (si nécessaire) et la dernière permettant de reconditionner de manière unitaire les produits avant de les réintégrer aux stocks.

Le bloc 4 permet quant à lui de préparer les différentes demandes (ou les produits qui la composent) selon le type de stockage. En effet, ce découpage comporte une partie frigorifique 4.1 (pour les produits nécessitant un tel stockage), une partie de picking manuelle 4.2 (pour solutés et produits déjà emballés), une partie picking sur stockeur mécanique 4.3 (un seul stockeur est utile le

second étant purement graphique), si un tel outil est utilisé et enfin une partie préparation automatisée avec des robots de dispensation 4.4 (chaque robot représente un cycle indépendant paramétrable).

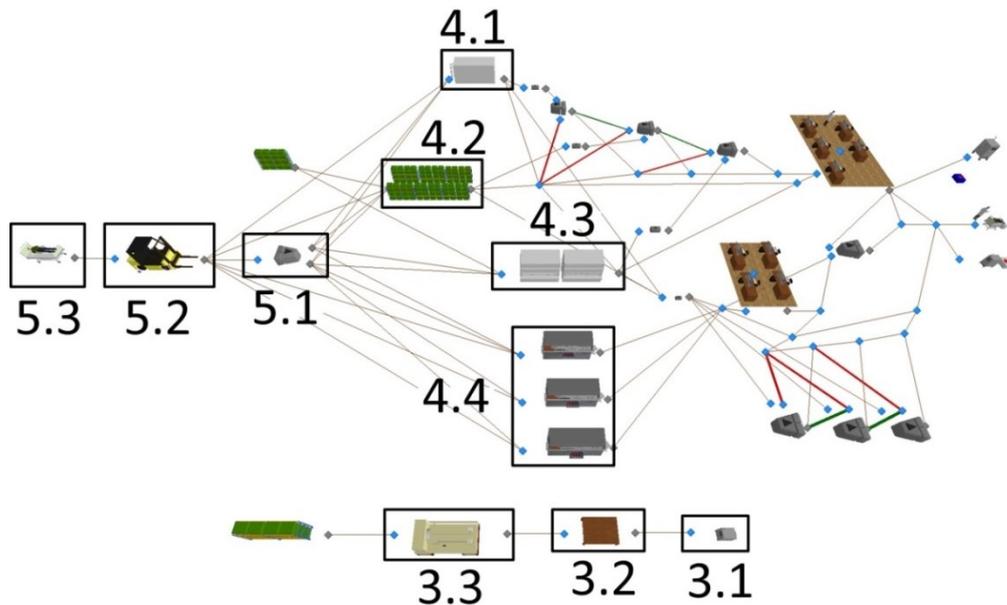


Figure 8.2 : Vue globale avec la décomposition des blocs 3, 4 et 5

Le bloc 5, est lui aussi découpé en trois sous-parties avec en 5.1, une partie permettant de regrouper les demandes entre elles (pour un service ou si une demande a été éclatée pour être préparée), en 5.2 une partie simulant le transport par la logistique jusqu’au service en 5.3.

8.2.3 Les parcours possibles

Du fait de sa généricité, de nombreux parcours différents peuvent être empruntés par les demandes et prescriptions qui parcourront notre modèle. Les choix quant au parcours à suivre sont dépendants des paramètres de la demande/prescription qui permettent aux nœuds de les orienter. La Figure 8.3 présente les parcours principaux existants. A noter que les parties concernant l’aiguillage des différentes demandes (pour le picking manuel, avec stockeur, frigo, etc.) ne sont pas représentées ici.

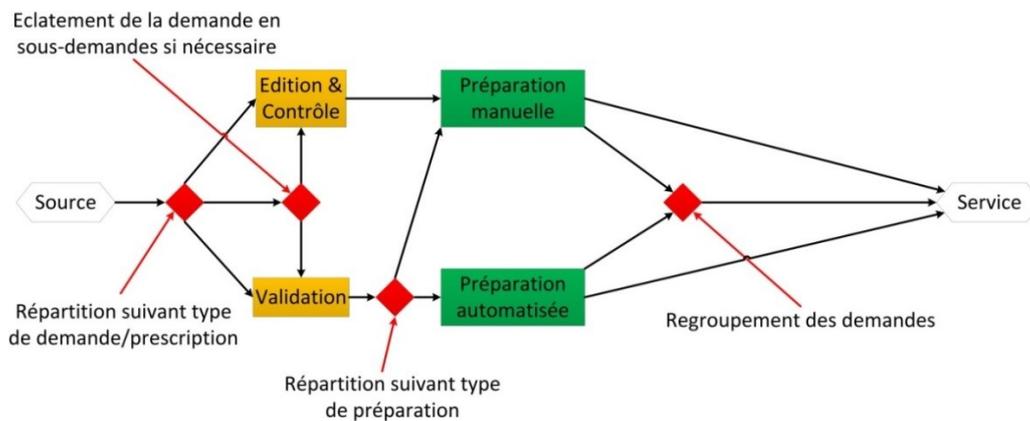


Figure 8.3 : Cheminement possible des demandes dans le modèle de simulation

Un coup d'œil rapide permet de voir que la complexité du CDM ne va pas résider dans les nombreuses étapes qui le composent, mais dans l'orientation des différentes demandes/prescriptions qui parcourent ce circuit. Cependant des parcours types peuvent être dressés :

- ❖ Source → Edition & Contrôle → Préparation manuelle → Service ;
- ❖ Source → Validation → Préparation manuelle → Service ;
- ❖ Source → Validation → Préparation automatisée → Service.

Comme il est possible de le constater, la différenciation des différents types de demandes et prescriptions aura lieu au début du parcours afin de simplifier leur orientation par la suite dans ce circuit. Nous l'expliquerons plus tard, mais les parties « Eclatement de la demande » et « Regroupement des demandes » permettent de découper et réunir une demande en plusieurs sous-demandes si les produits composant la demande de base doivent être préparés à différents endroits. Le regroupement attendra alors toutes les demandes liées avant de les expédier. Nous reviendrons sur cela dans la [partie 8.3.13](#).

8.3 LES ELEMENTS COMPOSANT LE MODELE

Nous allons maintenant aborder les différents éléments qui sont intégrés dans le modèle de simulation. Comme cela a été vu dans la partie de présentation de SIMIO, ces éléments peuvent être de différentes natures (objets fixes, nœuds, processus, etc.). Ils seront donc détaillés dans ce qui suit en abordant pour chacun d'entre eux les différents paramètres qui les caractérisent, leur type, classe, les processus et les nœuds qui leur sont liés ainsi que les données qu'ils utilisent.

8.3.1 Les travailleurs

Les travailleurs utilisés dans notre modèle ne sont guère différents. En effet, c'est l'utilisation qu'il est fait d'eux qui va réellement les cantonner dans un rôle ou dans un autre. Pouvant être « Préparateur », « Valideur » ou « Agent logistique » ces travailleurs interviennent à différents moments des processus, quand ils sont réquisitionnés par les différents serveurs composant notre modèle. Ainsi tandis que le préparateur interviendra tout au long de la chaîne, le valideur ne sera sollicité que dans le cadre de l'examen et de la validation des prescriptions, et l'agent logistique uniquement pour le transport des produits jusqu'au service demandeur. Leur représentation sous SIMIO est on ne peut plus simple (Figure 8.4 (a) et (b)), et leur population maximum est fixée au démarrage du système (par l'intermédiaire des tables de données). La population de ces travailleurs pourra donc varier durant la simulation mais ne pourra jamais excéder une certaine valeur.

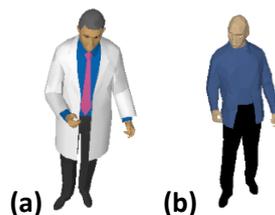


Figure 8.4 : Représentation des valideurs et préparateurs (a), et des agents logistiques (b)

Les caractéristiques de ces travailleurs sont celles définies par défaut, excepté pour le préparateur, pour qui la règle de sélection des entités le requérant est définie de la sorte :

- ❖ « Ranking Rule » : valeur la plus élevée en première (Largest value First) ;
- ❖ « Ranking Expression » : Prescriptions.Priority.

Le préparateur sélectionnera donc toujours la prescription qui a la priorité la plus élevée dans le modèle.

8.3.2 Les demandes

Les demandes sont à l’origine toutes identiques et sont basées sur la « Model_Entity » « Prescriptions ». Ce terme a été utilisé afin de faire une différence entre la « Demande » (Figure 8.5 (b)) et la « Prescriptions » (Figure 8.5 (a)) pour ne pas mélanger les deux termes lors de leur utilisation. Il s’agit uniquement d’un choix pratique. Quand nous parlerons donc des caractéristiques de la « Demande », celles-ci feront en réalité référence aux caractéristiques de l’entité modèle « Prescriptions ».



Figure 8.5 : Représentation sous SIMIO de la « Prescriptions » (a) et de la « Demande » (b)

8.3.2.1 Ses propriétés

Les propriétés de cet élément sont nombreuses et variées et permettent de caractériser de manière efficace une demande/prescription. Une partie de ces différents paramètres sont visibles dans le Tableau 8.1, la totalité étant disponible en **Annexe 5**.

Tableau 8.1 : Echantillon des propriétés de l’élément « Prescriptions »/« Demande »

Nom de la propriété	Type	Type de variable	Rôle pour la demande/prescription
Indicedemande	States	Entier	Indice du type de demande
Typedemande	States	Chaînes de caractères	Type de demande
Nbrproduitdemande	States	Réel	Nombre de produits dans la demande
Tempsprepa	States	Réel	Temps de préparation
Modetraitemendemande	States	Entier	Suivant la valeur définit le mode de préparation
Service_cible	States	Entier	Le numéro du service ayant fait la commande
Nombre_elmt_par_produit	States	Entier	Nombre d’éléments de chaque produit dans une demande
Hybridite	States	Entier	Détermine la nature hybride (=1) ou non (=0) d’une demande/prescription
Numero_prepa_necessaire	States	Entier	Récupère le numéro du préparateur devant préparer cette demande/prescription (si existant)

Nbr_pdt_en_stockeur	States	Réel	Défini le nombre de produits de la demande/prescription en stockeur
Nbr_elmts_pdt_solute	States	Réel	Défini le nombre d'éléments par soluté de la demande/prescription
Temps_prepa_pdt_frigo	States	Réel	Défini le temps de préparation des produits frigorifiques de la demande/prescription
Nbr_demande_nominative	States	Réel	Défini le nombre de prescriptions nominatives comprises dans la demande

A noter que pour faire appel à ces diverses propriétés dans le modèle de simulation, il est nécessaire de faire précéder la variable du nom de l'objet dont elle dépend. Par exemple, pour l'indice des demandes, lié à l'objet « Prescriptions », il faut écrire « *Prescriptions.Indicedemande* » pour obtenir la valeur de cet indice.

Le Tableau 8.1 présente des paramètres issus et traitants des domaines différents. Il y a en premier lieu des paramètres génériques tels que « *Indicedemande* », « *Modetraitementdemande* » ou « *Tempsprepa* ». Viens ensuite des paramètres plus spécifiques permettant de spécifier le caractère hybride d'une demande, le service qui a envoyé la demande, ou des informations relatives à la composition de la demande/prescription. Attardons-nous quelque peu sur certains de ces paramètres :

- ❖ *Nbrproduitdemande/Tempsprepa* : paramètres relatifs aux nombres de produits composant la demande/prescription et à son temps de préparation, ces données sont issues de relevés terrain et d'analyses de données. Elles font parties des paramètres de base sans lesquelles le modèle ne fonctionnerait pas ;
- ❖ *Modetraitementdemande* : suivant la méthode utilisée pour préparer la demande/prescription, la valeur de ce paramètre changera. Ainsi ce paramètre prendra la valeur « 1 » pour une préparation automatisée, « 2 » pour une préparation manuelle standard, « 3 » pour une préparation manuelle à l'aide de stockeur et enfin « 4 » pour une préparation de produits frigorifiques ;
- ❖ *Service_cible* : il est possible de spécifier le numéro du service (ou un parmi une liste) qui a envoyé cette demande/prescription ;
- ❖ *Hybridite* : nous entendons par hybridité dans notre modèle une demande qui comporte en son sein des produits préparés de manières diverses ou des prescriptions nominatives. Afin d'être le plus représentatif possible de la réalité une telle demande sera décomposée en sous-demandes ayant chacune un moyen de préparation différent. Prenons l'exemple le plus complet qui est une demande ayant des produits préparés de manière standard, avec stockeur, certains étant réfrigérés et possédant 4 prescriptions nominatives. Il sera alors obtenu (après traitement adéquat par un autre élément du modèle présenté plus tard), six sous-demandes en plus de la demande mère (Figure 8.6) ;
- ❖ *Nbr_pdt_stockeur/Nbr_elmts_pdt_solute/Temps_prepa_frigo* : une demande pouvant être découpée en sous-demandes comme nous venons de l'expliquer, il est nécessaire de fournir des informations quant aux nombres de produits/éléments et le temps pour les préparer. Les trois paramètres ici cités se retrouvent donc pour les éventuelles trois sous-demandes solutés/stockeur/frigo, pour un total de neuf paramètres supplémentaires.

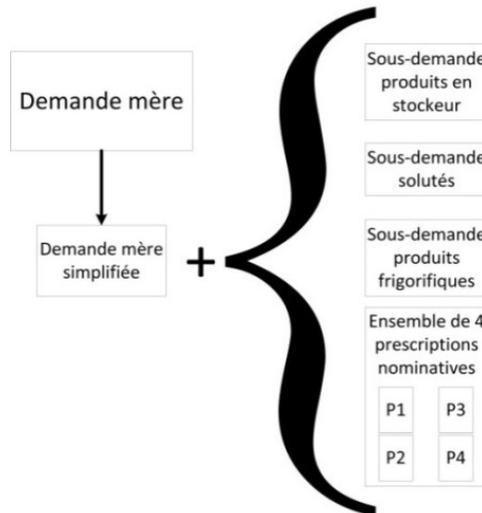


Figure 8.6 : Méthode de décomposition d’une demande en sous-demandes

Les demandes/prescriptions étant les éléments manipulés par le système, elles n’auront pas de processus et de données nécessaires à leur exécution. Ce sera l’ensemble des objets du modèle qui fourniront et changeront les données associées aux nombreux paramètres venant d’être présentés. Le premier d’entre eux est l’objet présenté dans la partie suivante à savoir le « Service ».

8.3.3 Le Service source

Le « Service » est la source de la plus grande partie des demandes qui arrivent des différentes unités de soins de l’hôpital.



Figure 8.7 : Source sous SIMIO des demandes (hors urgences et réapprovisionnement)

8.3.3.1 Ses propriétés

En tout état de cause, le « Service » est donc un objet fixe de type « Source ». Cependant et afin d’avoir un objet source plus « modulable », nous avons choisi de créer un modèle « Service » basé sur une source standard (*Générateur_demandes*) mais apportant des possibilités de gestion plus importantes. Cette source a donc été intégrée à un nouveau modèle fixe de SIMIO, dont les caractéristiques sont présentées dans le Tableau 8.2.

Tableau 8.2 : Propriétés principales de la source du Service

Catégorie	Caractéristique	Réponse	Description
Logique d'arrivée des entités	Type entité	Demande	L'entité utilisée par la source est de type « Demande »
	Mode d'arrivée	Time Varying Arrival Rate	Les arrivées dépendent d'une table temporelle
	Table Temporelle	Nouvelleprescription	
	Facteur temporel d'arrivée	Formule	Utilise une formule récupérant des données dans une table de données externe
	Entité par arrivée	1	
Table de référence pour assignation des états	Nom de la table	Donneesdemande	Table utilisée pour récupérer les données
	Numéro de la ligne	Formule	Utilise une formule récupérant le numéro d'une ligne suivant des paramètres temporels

La source génère donc des entités de type « Demande » suivant une table temporelle découpée en créneau d'une heure ou deux heures (dépendant du moment de la journée). Sur chaque créneau une seule arrivée est programmée, mais cette valeur étant multipliée par le facteur temporel d'arrivée, il est alors possible à l'utilisateur de définir sur chaque créneau combien de demandes/prescriptions (tout type confondu), il souhaite voir générées (Figure 8.8).

Starting Offset	Ending Offset	Rate (events per hour)	Heure0	Heure1	Heure2	Heure3	Heure4	Heure5	Heure6	Heure7	Heure8	Heure9
Day 1, 00:00:00	Day 1, 01:00:00	1	1	1	1	1	400	20	20	30	30	
Day 1, 01:00:00	Day 1, 02:00:00	1										
Day 1, 02:00:00	Day 1, 03:00:00	1										
Day 1, 03:00:00	Day 1, 04:00:00	1										
Day 1, 04:00:00	Day 1, 05:00:00	1										
Day 1, 05:00:00	Day 1, 06:00:00	1										
Day 1, 06:00:00	Day 1, 07:00:00	1										
Day 1, 07:00:00	Day 1, 08:00:00	1										
Day 1, 08:00:00	Day 1, 09:00:00	1										

Figure 8.8 : Vue de la table temporelle (à droite) et de la table des facteurs (à gauche)

Ainsi, à chaque création d'une demande/prescription, la source va aller chercher les données devant lui être associées dans une table de données et cela suivant un numéro de ligne bien défini. En effet, en utilisant la recherche par numéro de ligne, il y a une probabilité de tirer chaque ligne de la table de données source. Ces probabilités sont présentes dans la table elle-même, qui sera consultée à chaque nouveau tirage pour savoir dans notre cas, quel type de demande/prescription doit être généré. La table est ici « *Donneesdemande* » (Tableau 8.17) et sera présentée dans la **partie 8.4.3.6.**

La formule pour récupérer le numéro de ligne dépend du moment de la journée :

```
Math.If(DateTime.Hour(TimeNow)<Heure_Tranche0,Donneesdemande.ProbaavtH0.RandomRow,
Math.If(DateTime.Hour(TimeNow)<Heure_Tranche1,Donneesdemande.ProbaavtH1.RandomRow,
Math.If(DateTime.Hour(TimeNow)<Heure_Tranche2,Donneesdemande.ProbaavtH2.RandomRow,
Math.If(DateTime.Hour(TimeNow)<Heure_Tranche3,Donneesdemande.ProbaavtH3.RandomRow,
Math.If(DateTime.Hour(TimeNow)<Heure_Tranche4,Donneesdemande.ProbaavtH4.RandomRow,
Donneesdemande.ProbaapresH4.RandomRow))))))
```

Ainsi pour une heure comprise entre « *Tranche0* » et « *Tranche1* », une ligne sera tirée aléatoirement (*RandomRow*) suivant la probabilité « *ProbaavtH1* ».

8.3.3.2 Les données nécessaires à son exécution

Certainement l'élément du modèle qui requiert le plus de données en entrée, la source associe de nombreuses informations aux caractéristiques des demandes/prescriptions, mais aussi à certaines variables du modèle lui-même. De nouveau le nombre de caractéristiques étant important seules certaines d'entre elles seront présentées, la totalité étant en **Annexe 5**.

Tableau 8.3 : Echantillon des caractéristiques de la « Demande » fournies par le « Service »

Caractéristique	Variable de la table/Formule/Variable SIMIO
Prescriptions.Nbrproduitdemande*	Math.If((DateTime.Hour(TimeNow)+DateTime.Minute(TimeNow)/60+DateTime.Second(TimeNow)/3503.65)<12,Donneesdemande.Nbrproduit_matin,Donneesdemande.Nbrproduit_apmidi)
Prescriptions.Entite_hybride	0
Prescriptions.Retard_demande	Random.Exponential(1)
Prescriptions.Nbr_demande_nominative	Donnees_demandes_hybrides[Prescriptions.Indicedemande].Nbr_nominative
Prescriptions.Priority	1

*A noter que pour toutes les formules suivantes, l'expression « *(DateTime.Hour(TimeNow)+DateTime.Minute(TimeNow)/60+DateTime.Second(TimeNow)/3503.65)* » sera notée « **H** »

Le Tableau 8.3 précédent présente les quelques caractéristiques les moins triviales qui sont fournies par la source aux diverses demandes. Pour la première « *Prescriptions.Nbrproduitdemande* », la source va associer suivant l'heure (matin ou après-midi) un nombre de produits qui peut varier d'une demande à une autre. La seconde « *Prescriptions.Entite_hybride* » signifie qu'à la création aucune demande n'est une sous-demande. Cette différenciation interviendra après la création comme nous le verrons. « *Prescriptions.Retard_demande* » permet d'imposer un retard à chaque demande lors de cette préparation. Dans l'exemple ce retard est défini suivant une loi exponentielle de moyenne « 1 ». La suivante « *Prescriptions.Nbr_demande_nominative* » se voit associée par la source une valeur de la table « *Donnees_demandes_hybrides* ». Cette donnée sera dépendante de « *Prescriptions.Indicedemande* » permettant de caractériser le type de la demande et sera le nombre de prescriptions nominatives que comportera cette demande (*Nbr_nominative*). Enfin « *Prescriptions.Priority* », mise à « 1 » lors de la création, évoluera lors de son parcours du modèle,

cette valeur permettant de prioriser la préparation des demandes quand elles sont dans des files d'attentes.

8.3.3.3 Ses processus

Le dernier élément permettant de définir le comportement d'un objet de SIMIO est comme nous l'avons déjà vu les processus. Dans notre cas, notre « Service » se verra attribuée deux processus de type « add-on ». En effet, une fois la demande/prescription créée, certaines informations vont lui être ajoutées (en plus de celles citées dans la partie précédente). Ces informations seront dépendantes, pour le premier, du jour de la semaine et du fait de l'attribution ou non de ces demandes à des services spécifiques tandis que le second concernera le temps de transport de cette demande vers le service demandeur. Les processus en question se nomment « *Choix_planning* », qui se place en tant que processus « add-on » dans « Created Entity », et « *Temps_transport_service* », qui se place dans « Exited ».

La Figure 8.9 présente le processus « *Choix_planning* » dans son entièreté. Suivant l'existence de planning de commandes précis, le processus poursuivra en « 1 » (si existant) sinon en 3. Le bloc « 1 » associera en vérifiant le jour de la semaine, un numéro de service à une demande (globale ou globalisée dans la logique). Le bloc « 3 » associera lui aussi un numéro de service à une demande mais de manière aléatoire, le numéro de service étant sélectionné dans une liste suivant des probabilités pouvant être définies. Enfin le bloc « 2 » associe, si existant, un préparateur spécifique à la demande. L'exemple parfait, est quand certains préparateurs s'occupent de services spécifiques. Dans ce cas-là il y a alors mise en adéquation des informations.

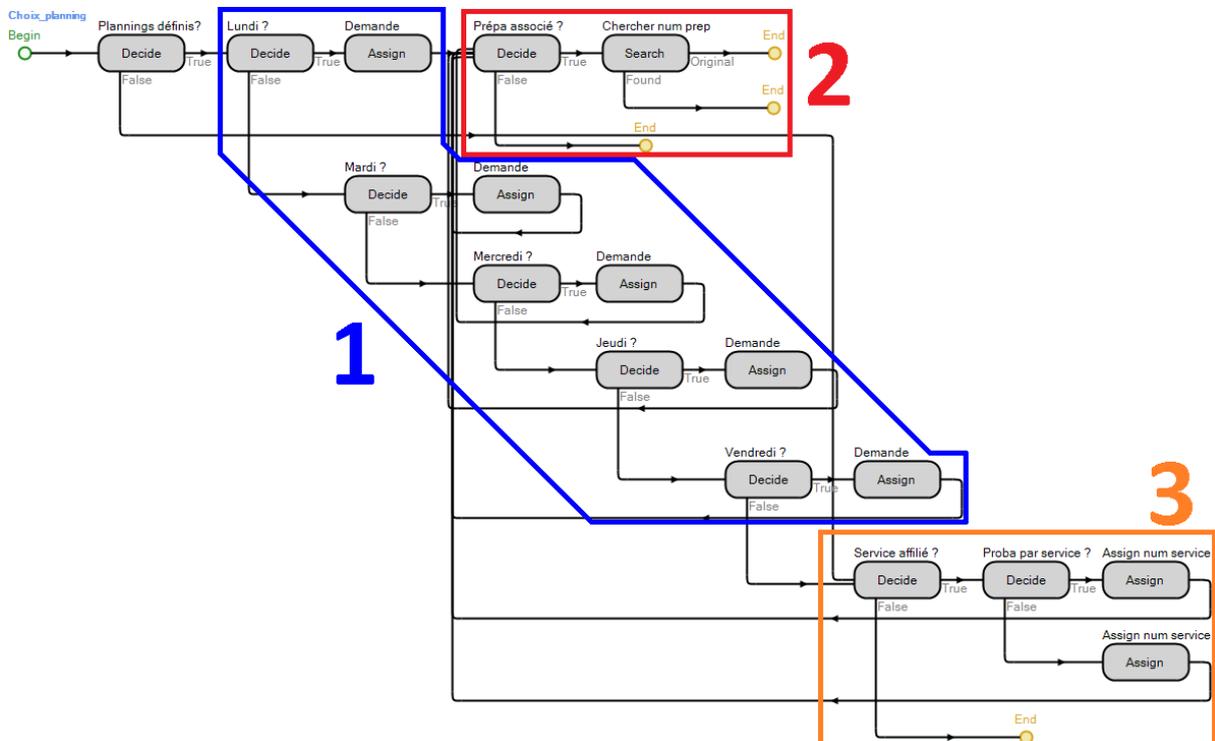


Figure 8.9 : Vue du processus « *Choix_planning* » de la Source

La Figure 8.10, présente le second processus « *Temps_transport_service* ». Dans celui-ci, le premier bloc en rouge recherche dans la table « Transport », si le numéro du service associé à la demande (si existant bien entendu) est présent dans cette table. Si tel est le cas, alors le bloc orange est sollicité pour rechercher le temps de transport pour rejoindre ce service de la pharmacie. Sinon, le bloc vert sera sollicité, ce dernier assignant une valeur de transport par défaut à la demande.

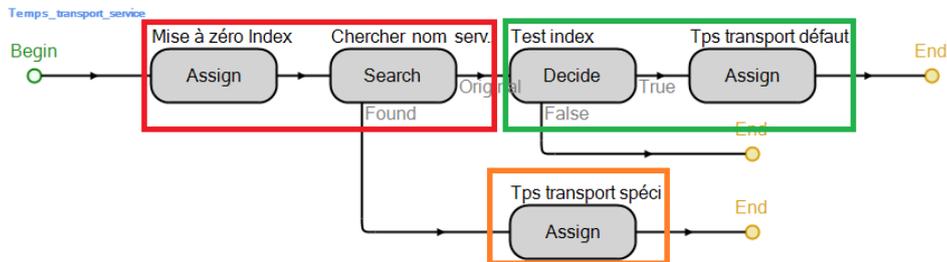


Figure 8.10 : Processus définissant le temps de transport d’une demande

8.3.4 Les demandes urgentes et leur génération

Ce type de demande est représentatif d’une situation particulière, lors de laquelle des services sollicitent en urgence la pharmacie pour se voir accorder des médicaments ou produits pharmaceutiques spécifiques. Ce cas se rencontre notamment aux services des urgences mais aussi dans les autres unités de soins, lors d’une aggravation subite de la santé d’un patient.

Afin de rester cohérent avec les éléments « Service » et « Demande » présentés précédemment, nous avons fait le choix de réutiliser au maximum ces deux éléments pour gérer la situation des demandes urgentes. Par conséquent, nous réutilisons l’entité « Prescriptions » pour générer une nouvelle entité « Demande_urgente » qui est en tout point équivalente à la « Demande », et que nous ne représenterons donc pas.

Le cas de la source générant ces demandes urgentes est un peu plus complexe. Nous avons réutilisé la « Source » pour concevoir une source « Generateur_urgences », mais contrairement à l’objet de flux, les changements apportés vont être beaucoup plus conséquent, même s’ils se résument en grande partie à la suppression de fonctions, comme nous allons le voir.



Figure 8.11 : Vue de la source urgente, de la demande urgente et du nœud de sortie de la source

8.3.4.1 Les données

Parmi les données qui ne vont plus être nécessaires ou qui seront amenées à changer pour la génération des demandes urgentes, il y a en premier lieu, celles liées à la création de demandes pour

chaque jour de la semaine. En effet, les demandes urgentes pouvant arriver tout au long de la journée, les variables « *Nbr_demande_lundi* », « *Nbr_demande_mardi* », etc. vont disparaître. Il ne sera donc sollicité que la création de demandes aléatoirement sur la journée.

La donnée « *Possible_hybride* » sera dans tous les cas mise à zéro, étant donné que les demandes urgentes ne posséderont pas de sous-demandes.

La donnée « *Prescription.Priority* » (Tableau 8.3) sera quant à elle mise à 5 (ou toute valeur plus élevée), cela permettant de donner la plus haute priorité possible à ces demandes, celles-ci étant alors préparées en première lors de leur arrivée à la pharmacie.

8.3.4.2 Ses processus

Dans les processus, il y aura une modification majeure du processus « *Choix_planning* » et cela à cause une nouvelle fois de la suppression de la création de demandes spécifiques pour chaque jour (Zone 1 Figure 8.9). Le nouveau processus obtenu sera grandement simplifié, ne se limitant plus qu'au choix d'un service et au choix d'un préparateur, si de telles informations sont fournies. Le second processus lié au transport restera quant à lui inchangé.

8.3.4.3 Les tables

Des tables de données ne seront plus exploitées par la nouvelle source, et par conséquent, disparaîtront des registres de celles-ci. Ces tables sont « *Donnes_demandes_hybrides* » et « *Planning_demande_G* », la disparition de la première s'expliquant par le fait que les demandes urgentes sont en grande majorité mono-produit et ne posséderont pas de sous-demandes, tandis que la seconde est en lien avec la suppression de demandes pour chaque jour de la semaine.

Mis à part leur création, les « *Demande_urgente* » suivront les mêmes parcours que les demandes standards et solliciteront des éléments du modèle parmi ceux présentés dans ce qui suit.

8.3.5 La séparation des prescriptions d'une demande

Dans un premier temps, une fois la demande/prescription créée, celle-ci est orientée vers un premier nœud nommé « *Demande_ou_Prescription* » (Figure 8.12).

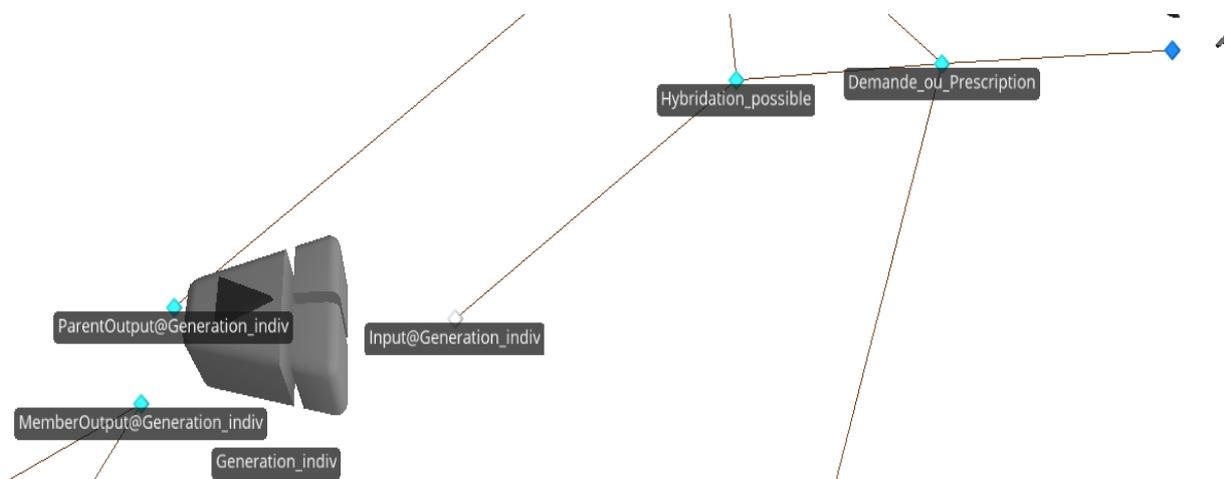


Figure 8.12 : Séparateur prescription/demande (en bas à gauche) et du nœud d'orientation (en haut à droite)

Il est alors décidé suivant la méthode de préparation (automatisée ou manuelle ou si besoin d'être validée) d'orienter les demandes/prescriptions. Tandis que les prescriptions (donc qui nécessitent une préparation automatisée ou une validation) iront directement vers un nœud « *Choix_validation* », que nous détaillerons plus tard, les demandes pourront être orientées vers un nouveau nœud « *Hybridation_possible* » quand leur caractéristique d'hybridité sera égale à « 1 ».

Dans ce cas, et suivant un pourcentage défini par la variable « *Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_nominative* », ces demandes pourront être accompagnées de prescriptions nominatives qu'il sera alors nécessaire de séparer de la demande mère. C'est le rôle du séparateur « *Generation_indiv* ».

8.3.5.1 Ses propriétés

Commençons par les deux nœuds. Leurs propriétés sont similaires exception faite de la logique de trajet (Tableau 8.4).

Tableau 8.4 : Propriétés des nœuds « Demande_ou_prescription » et « Hybridation_possible »

Catégorie	Caractéristique	Réponse
Logique de parcours		
	Capacité de voyage initiale	Infinie
	Règle de sélection à l'entrée	FIFO
Logique de trajet		
	Préférence du chemin de sorti	Aucun
	Règle de sélection du chemin de sorti	Chemin le plus court pour « Demande_ou_Prescription » Par poids pour « Hybridation_possible »

La règle de sélection de « *Demande_ou_Prescription* » n'ayant que peu d'importance, la destination étant choisie par un processus « add-on » que nous détaillerons dans la [partie 8.3.5.3](#).

En ce qui concerne le séparateur ses propriétés sont celles par défaut, excepté celles de la catégorie « Logique de séparation ». En effet, le mode de séparation est ici « Make copies » et la quantité à copier est « *Prescriptions.Nbr_demande_nominative* ». Ainsi, lorsqu'une demande mère arrive, ce séparateur réalise des copies de la demande mère suivant le nombre défini par « *Prescriptions.Nbr_demande_nominative* ».

8.3.5.2 Les données nécessaires à son exécution

La majorité des données nécessaires à ce bloc sont demandées par le séparateur. Seules deux d'entre elles sont présentes sur les chemins sortant du nœud « *Hybridation_possible* ».

Il s'agit de « *1-Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_nominative* » et « *Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_nominative* », celles-ci représentant le poids de chaque chemin sortant du nœud précédemment cité. Ainsi pour chaque type de demande, le modèle regardera s'il y a une probabilité que celle-ci comporte des prescriptions et il redirigera donc en conséquence la demande suivant le poids des deux chemins.

Une fois la demande mère copiée, bien qu'une partie des paramètres de celles-ci soient héréditaires pour les prescriptions nominatives créées (par exemple le service ayant fait la demande), un certain nombre de données sont mises à jour (Partie « State assignments » du séparateur) autant pour la demande mère que pour les prescriptions. Etant peu nombreuses, nous allons entièrement les présenter ici.

❖ Pour la demande mère :

- *Prescription.nombre_ss_entite_hybride* = *Prescriptions.Nbr_demande_nominative* : le nombre de prescriptions nominatives est sauvegardé sur la demande mère ;
- *Prescriptions.Entite_hybride* = 1 : cette variable est mise à « 1 » afin de préciser qu'elle fait partie d'un ensemble de sous-demandes ;
- *Prescriptions.Demande_parent* = 1 : la variable est cette fois égale à « 1 » afin de signifier qu'elle est une demande mère ;
- *Prescriptions.Priority* = 4 : la demande mère acquiert la plus haute priorité.

❖ Pour les prescriptions :

- *Prescriptions.Modetraitemendemande* = *Donnees_demandes_hybrides[Prescriptions.Indicedemande].Mode_prepa_nominative* : récupère dans la table « *Donnees_demandes_hybrides* » le mode de préparation (parmi les quatre existants) de la prescription générée ;
- *Prescriptions.Nbrproduitdemande* = *Donnees_demandes_hybrides[Prescriptions.Indicedemande].Nbr_prod_nominative* : récupère dans la table « *Donnees_demandes_hybrides* » le nombre de produits présent dans la prescription générée ;
- *Prescriptions.Nombre_elmt_par_produit* = *Donnees_demandes_hybrides[Prescriptions.Indicedemande].Nbr_elmts_nominative* : récupère dans la table « *Donnees_demandes_hybrides* » le nombre d'éléments par produit présent dans la prescription générée ;
- *Prescriptions.Tempsprepa* = *Donnees_demandes_hybrides[Prescriptions.Indicedemande].Temps_prepa_nominative* : récupère dans la table « *Donnees_demandes_hybrides* » le temps de préparation de la prescription générée ;
- *Prescriptions.Entite_hybride* = 1 : cette variable est mise à « 1 » afin de préciser qu'elle fait partie d'un ensemble de sous-demandes ;
- *Prescriptions.Demande_parent* = 0 : la variable est cette fois égale à « 0 » afin de signifier qu'elle est une sous-demande rattachée à une demande mère.
- *Prescriptions.Priority* = 3 : la prescription acquiert la priorité de niveau 2.

8.3.5.3 Ses processus

L'unique processus utilisé dans ce bloc est celui associé au nœud « *Demande_ou_Prescription* », se nomme « *Choix_apres_creation* » (Figure 8.13) et est placé en tant que processus « add-on » dans « *Entered* ».

Ici, le premier « *Decide* » nommé « *Prépa_auto ou Valid?* » vérifie que l'élément traversant le nœud nécessite une préparation automatisée ou a besoin d'être validé (ce qui concerne uniquement les prescriptions nominatives dans notre modèle). Si tel est le cas, il l'envoie vers le nœud

« *Choix_validation* ». Dans le cas contraire, le processus regarde si la demande est hybride. De même, si tel est le cas il envoie la demande vers le nœud « *Hybridation_possible* ». Sinon, la demande est directement envoyée au nœud d'entrée du serveur « *EditionControle* ».

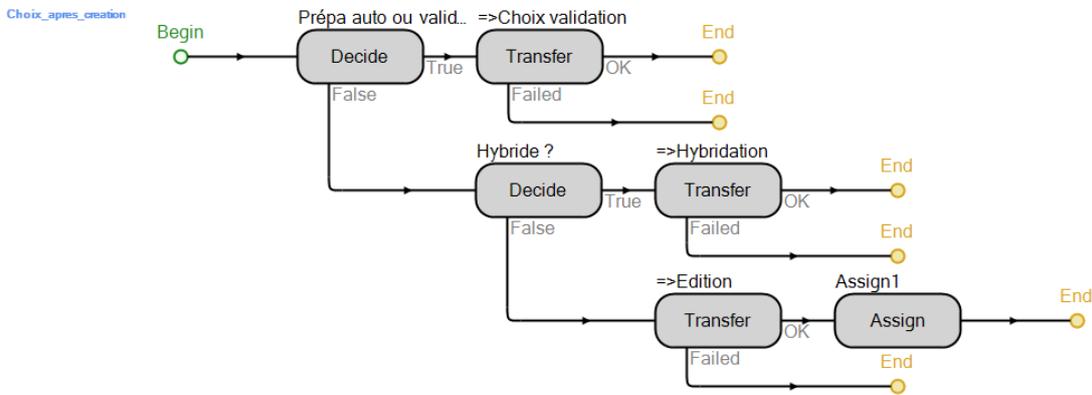


Figure 8.13 : Vue du processus « *Choix_apres_creation* »

8.3.6 La décomposition des prescriptions

Etant donné qu'il est possible de suivre la chaîne de préparation des prescriptions ou la chaîne de préparation des demandes, nous allons dans un premier temps nous intéresser à tous les éléments permettant de préparer les prescriptions. Après une éventuelle validation (partie suivante), les prescriptions peuvent être décomposées en sous-prescriptions si cela est nécessaire. En effet, tout comme les demandes, les prescriptions peuvent comporter divers types de produits, ce qui nécessite de pouvoir les éclater en diverses sous-prescriptions.

Cependant, cette situation ne pourra se trouver que pour des prescriptions qui ne sont pas issus d'une demande mère, cette double décomposition complexifiant trop la gestion de nos entités dans le modèle si elle venait à être appliquée.

Cette décomposition est réalisée par trois séparateurs chacun traitant un type de produits (frigo, solutés, stockeur), et est similaire à celle de la Figure 8.6 (Figure 8.14).

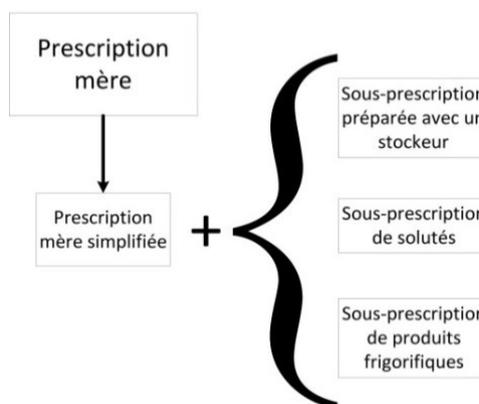


Figure 8.14 : Méthode de décomposition d'une prescription en sous-prescriptions

La Figure 8.15 présente ces trois séparateurs, accompagnés de divers nœuds permettant le voyage des prescriptions dans cette partie du modèle.

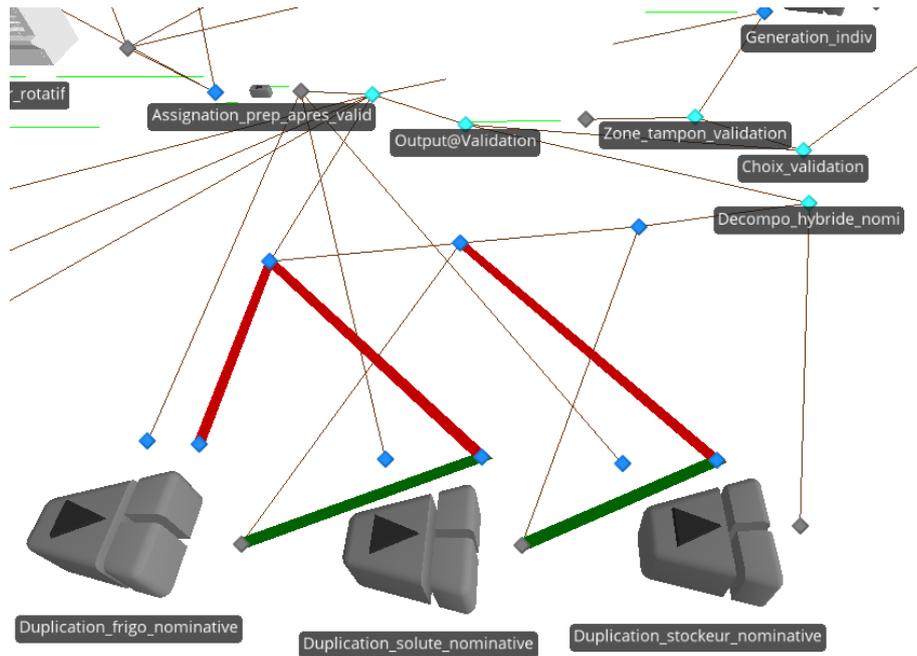


Figure 8.15 : Vue des trois séparateurs (en bas) permettant de décomposer les prescriptions

8.3.6.1 Ses propriétés

Tous les nœuds possédant des chemins de sorties multiples, à l'exception de « *Zone_tampon_validation* », sont gérés comme le nœud « *Hybridation_possible* » décrit précédemment (Tableau 8.4), c'est-à-dire en choisissant le chemin de sorti suivant le poids de celui-ci (Règle « By Link Weight »). Cependant les poids varieront d'un nœud à l'autre.

De même, les trois séparateurs utilisés fonctionnent exactement de la même manière. Ils produisent chacun une copie de la prescription avant de lui attribuer de nouvelles données que nous détaillerons dans la partie suivante. Les propriétés (partie « Separation Logic ») des séparateurs sont donc :

- ❖ Mode de séparation = « Faire copies » ;
- ❖ Nombre de copie = 1.

8.3.6.2 Les données nécessaires à son exécution

Ci-dessus, nous avons précisé que le poids de chaque chemin change selon le nœud d'origine, ce qui nécessite donc autant de données que de chemins. Le Tableau 8.5 récapitule donc les différents poids de sortie des chemins.

Tableau 8.5 : Poids appliqués aux chemins ayant pour origine l'un des nœuds présents

Nœud	Poids de sortie
Choix_validation	« 1- Prescriptions.Validationdemande » et « Prescriptions.Validationdemande »
Output@Validation	« 1-Prescriptions.Hybridite » et « Prescriptions.Hybridite »
Decompo_hybride_nomi	« 1-Donnees_demandes_hybride[Prescription.Indicedemande].Pourc_stockeur » et « Donnees_demandes_hybride[Prescription.Indicedemande].Pourc_stockeur »

Decompo_hybride_nomi_1	« 1-Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute » et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute »
Decompo_hybride_nomi_2	« 1- Donnees_demandes_hybrides [Prescriptions.Indicedemande].Pourc_frigo » et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo »
ParentOutput@Duplication _stockeur_nominative	« 1-Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute » et « Donnees_demandes_hybride[Prescriptions.Indicedemande].Pourc_solute »
ParentOutput@Duplication _solute_nominative	« 1- Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo » et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo »

Les dernières données nécessaires à l'exécution de ce bloc, sont celles requises dans le cadre de la mise à jour des prescriptions sortant de chaque séparateur. Ces données étant quasi identiques pour chaque séparateur, nous ne les présenterons que pour le séparateur créant une sous-prescription de produits frigorifiques :

❖ Pour la prescription mère :

- *Prescription.nombre_ss_entite_hybride = Prescription.nombre_ss_entite_hybride + 1* : mise à jour du nombre de sous-prescriptions, une seule étant créée à chaque fois;
- *Prescriptions.Entite_hybride = 1* : cette variable est mise à « 1 » afin de préciser qu'elle fait partie d'un ensemble de sous-demandes ;
- *Prescriptions.Demande_parent = 1* : la variable est cette fois égale à « 1 » afin de signifier qu'elle est une demande parent ou mère.
- *Prescriptions.Priority = 4* : cette prescription devient prioritaire devant les autres ;
- *Prescriptions.Nbrproduitdemande = Prescription.Nbrproduitdemande - Nbr_pdt_ssentite_frigo_nomi* : le nombre de produits inclus dans la sous-prescription créée sera retiré du nombre de produit de la prescription mère ;
- *Prescriptions.Tempsprepa = Math.If(Prescriptions.Tempsprepa < Tps_prepa_ssentite_frigo_nomi, 0, Prescription.Tempsprepa - Tps_prepa_ssentite_frigo_nomi)* : de même que pour le nombre de produit, le temps de préparation de la prescription mère se voit soustraire le temps de préparation.

❖ Pour les sous-prescriptions :

- *Prescriptions.Modetraitementdemande = 2, 3 ou 4* : suivant le séparateur le mode de traitement sera différent pour la sous-prescription créée ;
- *Prescriptions.Nbrproduitdemande = Donnees_demandes_hybrides[Prescriptions.Indicedemande].Nbr_frigo* : récupère dans la table « *Donnees_demandes_hybrides* » le nombre de produits (frigorifique, solutés ou stockeur suivant le séparateur) dans la sous-prescription générée ;
- *Nbr_pdt_ssentite_frigo_nomi = Prescriptions.Nbrproduitdemande* : dans cette variable est enregistrée la valeur de la variable précédente (même chose pour « *Nbr_pdt_ssentite_solute_nomi* » et « *Nbr_pdt_ssentite_stockeur_nomi* ») ;

- *Prescriptions.Priority = 3* : cette sous-prescription devient prioritaire devant les prescriptions de base ;
- *Prescriptions.Nombre_elmt_par_produit = Donnees_demands_hybrides[Prescriptions.Indicedemande].Nbr_elmts_frigo* : récupère dans la table « *Donnees_demands_hybrides* » le nombre d'éléments par produit présent dans la prescription générée ;
- *Prescriptions.Entite_hybride = 1* : cette variable est mise à « 1 » afin de préciser qu'elle fait partie d'un ensemble de sous-prescriptions ;
- *Prescriptions.Demande_parent = 0* : la variable est cette fois égale à « 0 » afin de signifier qu'elle est une sous-prescription rattachée à une prescription mère ;
- *Prescriptions.Tempsprepa = Donnees_demands_hybrides[Prescriptions.Indicedemande].Temps_prepa_frigo* : le temps de préparation est récupéré dans la table « *Donnees_demands_hybrides* » pour les produits frigorifiques (même chose en stockeur et pour les solutés) ;
- *Tps_prepa_ssentite_frigo_nomi = Prescriptions.Tempsprepa* : dans cette variable est enregistrée la valeur de la variable précédente (même chose pour « *Tps_prepa_ssentite_solute_nomi* » et « *Tps_prepa_ssentite_stockeur_nomi* »).

8.3.6.3 Ses processus

Dans cette portion du modèle seul le nœud « *Zone_tampon_validation* » utilise des processus pour régir sa logique. Contrairement aux différents éléments présentés jusqu'à maintenant, ce nœud n'utilise pas un mais deux processus « add-on ». Le premier processus « *Temps_attente* » – placé dans « *Entered* » – fait patienter les prescriptions devant être validées jusqu'à ce qu'un des valideurs soit disponible. Le second « *Niveau_analyse* » – placé dans « *Exited* » – permet de récupérer les données liées aux divers niveaux d'analyses des prescriptions (par exemple le temps moyen d'une analyse de niveau trois – la plus élevée).

Commençons par détailler le plus simple, à savoir celui concernant l'analyse des prescriptions (Figure 8.16). Celui-ci est basé sur une logique extrêmement simple. A l'aide de blocs « *Decide* », un niveau d'analyse est associé à chacune des prescriptions devant être validée. Cette association est faite suivant des probabilités (*Pourcentage_Niv1_analyse*, *Pourcentage_Niv2_analyse*, *Pourcentage_Niv3_analyse*), et ce niveau d'analyse est caractérisé par un temps (fixe ou variable).

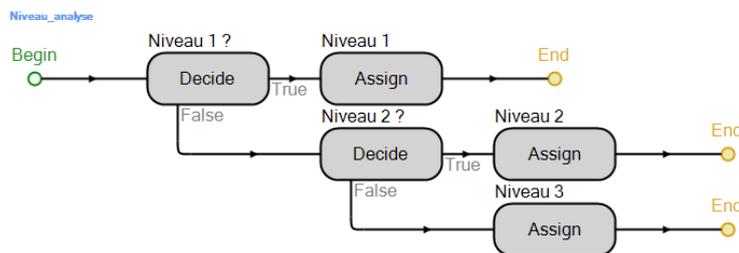


Figure 8.16 : Vue du processus définissant le temps de validation

Le second processus (Figure 8.17) est lui aussi assez basique dans son fonctionnement. En vérifiant dans un premier temps que la prescription arrive en journée (bloc rouge) et non de nuit – imposant alors d'attendre le début de la présence des valideurs (bloc vert en bas à droite) – le processus va vérifier tour à tour la présence du valideur adéquat pour examiner la prescription. Ainsi

les différents blocs bleus, ainsi que le noir, servent à vérifier que le valideur 2, 3,4,5 ou 6 est bien présent pour réaliser cette tâche.

Prenons un exemple : Soit deux valideurs V1 et V2, l'un travaillant entre 8h et 10h et le second entre 14h et 16h. Une prescription arrivant à 11h ne pourra être envoyée vers le V1. On passe donc du premier bloc vert au premier bloc bleu. Le bloc « Decide » ici présente vérifie l'heure d'arrivée de la prescription (11h) et constate que le V2 n'a pas encore commencé son travail (début à 14h). Il fera donc patienter la prescription trois heures jusqu'à la disponibilité de V2.

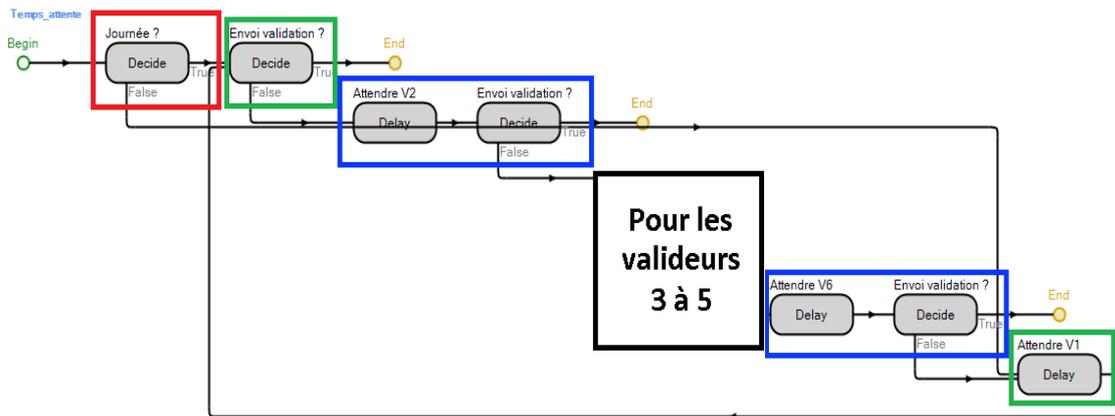


Figure 8.17 : Vue du processus de gestion de la disponibilité des valideurs

8.3.7 La validation des prescriptions

Maintenant que nous avons vu le traitement des prescriptions, intéressons-nous à la validation. Comme déjà dit, la validation sera dépendante du niveau d'analyse de la prescription et de la présence de valideurs pour réaliser cette tâche. Le pôle validation de notre modèle est un serveur standard (Figure 8.18) dont la capacité de validation (le nombre de valideurs disponibles) varie en fonction du temps.

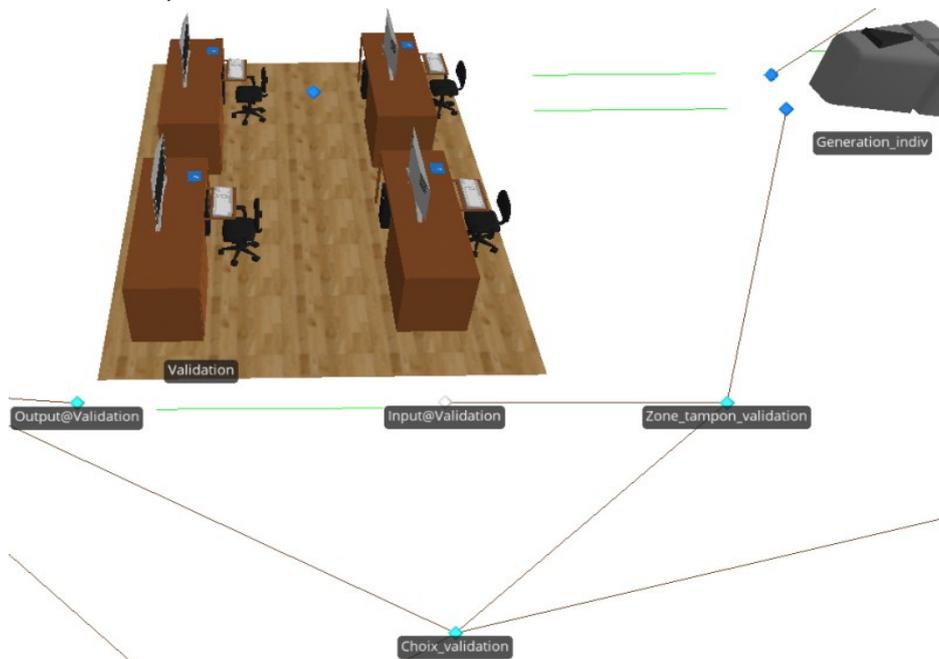


Figure 8.18 : Vue de la zone validation

8.3.7.1 Ses propriétés

Le serveur « Validation » possède comme tout autre un nœud d'entrée et un nœud de sortie. Les deux n'ont rien de particulier et leurs propriétés sont celles définies par défaut. De même le serveur n'a pas été grandement modifié. La seule de ses propriétés qui change est le « *Processing time* » défini par le biais d'une formule mathématique :

$$\text{Math.If}(\text{Prescriptions.Niveau_analyse_valid}==1, \text{Donneessimulation}[1].\text{Temps_analyse_niv_01}, \\ \text{Math.If}(\text{Prescriptions.Niveau_analyse_valid}==2, \text{Donneessimulation}[1].\text{Temps_analyse_niv_02}, \\ \text{Donneessimulation}[1].\text{Temps_analyse_niv_03}))$$

Cette formule fait donc le lien entre niveau d'analyse et temps d'analyse, et se lit :

- ❖ Si l'analyse doit être de niveau « 1 » (*Prescriptions.Niveau_analyse_valid==1*)
 - Alors le temps d'analyse est « *Donneessimulation[1].Temps_analyse_niv_01* »
- ❖ Si l'analyse doit être de niveau « 2 » (*Prescriptions.Niveau_analyse_valid==2*)
 - Alors le temps d'analyse est « *Donneessimulation[1].Temps_analyse_niv_02* »
- ❖ Sinon le temps d'analyse est « *Donneessimulation[1].Temps_analyse_niv_03* »

Les véritables changements apportés à ce serveur seront donc en fait dans la partie des processus « add-on ». En utilisant quatre processus différents, son comportement en est grandement modifié. Nous détaillerons cela dans la [partie 8.3.7.3](#).

8.3.7.2 Les données nécessaires à son exécution

Nous allons aborder ici les données que nous retrouverons dans les processus de la partie suivante et dans les différentes tables de données détaillées plus tard. Le Tableau 8.6 présente ces nombreuses informations :

Tableau 8.6 : Données nécessaires dans le cadre de la validation

Utilisation	Nom	Rôle
Formule « Processing time »	Donneessimulation[1]. Temps_analyse_niv_01	Temps d'analyse d'une prescription de niveau 1 (même chose pour niveau 2 et 3)
Processus « Chargementdonneesvalideur »	Gestion_valideurs[1]. Nombre_valideurs	Nombre de valideurs de type 1 (Gestion_valideurs2].Nombre_valideurs pour type 2, et ainsi de suite)
Processus « Capacite_validation »	Gestion_valideurs[1]. Horaire_debut	Début horaire de travail du valideur de type 1 (Gestion_valideurs[2].Horaire_debut pour type 2, et ainsi de suite)
	Gestion_valideurs[1]. Horaire_fin	Horaire de fin de travail du valideur de type 1 (Gestion_valideurs[2].Horaire_fin pour type 2, et ainsi de suite)
	Gestion_valideurs[1]. Nombre_valideurs	Ci-dessus

Processus « Selectionvalideur »

Gestion_valideurs[1]. Horaire_fin	Voir ci-dessus
Gestion_valideurs[1]. Nombre_valideurs	Voir ci-dessus

Les processus de la validation sont donc majoritairement liés à des contrôles temporels et de disponibilité. L’avantage d’utiliser ces processus est que l’on gagne en flexibilité par rapport à l’utilisation des plannings intégrés. Nous reverrons d’ailleurs ce type de gestion pour toutes utilisations d’entités de type travailleur.

8.3.7.3 Ses processus

Détaillons maintenant le contenu des processus utilisés par le serveur « Validation ». Ces processus sont de nouveau assez simples dans leur fonctionnement, mais certains ont nécessité une certaine gymnastique dans leur conception afin de les simplifier, et d’obtenir les processus finaux.

Le premier processus « add-on » est « *Chargementdonneesvalideur* » (Figure 8.19) et se place dans « Run Initialized ». Celui-ci est utilisé pour créer le bon nombre de valideurs et cela pour chaque type.

Ainsi, le processus va contrôler – bloc « Decide » – si chaque type de valideur existe et combien ils doivent être. Ensuite par le biais du bloc « Create », il créera autant de valideurs que nécessaire.

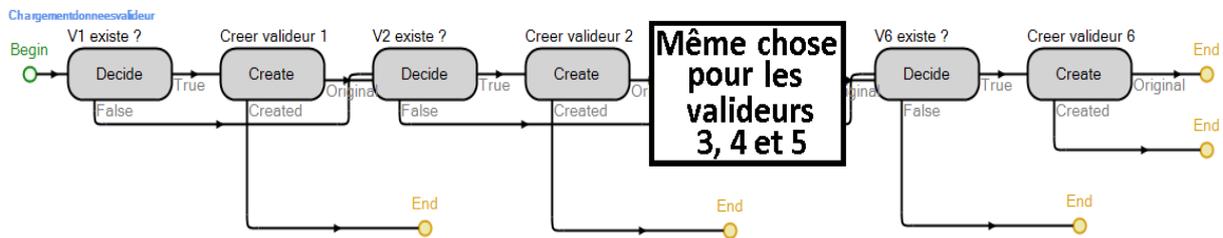


Figure 8.19 Vue du processus « Chargementdonneesvalideur »

Le second processus intervenant est « *Capacite_validation* » (Figure 8.20) et se place en « Entered ». Il permet de mettre à jour selon l’heure le nombre de valideurs en train de travailler. Il ira donc contrôler sur chaque créneau horaire le nombre de valideurs présents. Ce contrôle est réalisé à l’aide d’un bloc « Assign », qui met à jour la variable « *Validation.CurrentCapacity* » et utilise la formule suivante :

```

Math.If(H < Gestion_valideurs[1].Horaire_debut,0,
Math.If(H <= Gestion_valideurs[1].Horaire_fin,Gestion_valideurs[1].Nombre_valideurs,
Math.If(H <= Gestion_valideurs[2].Horaire_fin,Gestion_valideurs[2].Nombre_valideurs,
Math.If(H <= Gestion_valideurs[3].Horaire_fin,Gestion_valideurs[3].Nombre_valideurs,
Math.If(H <= Gestion_valideurs[4].Horaire_fin,Gestion_valideurs[4].Nombre_valideurs,
Math.If(H <= Gestion_valideurs[5].Horaire_fin,Gestion_valideurs[5].Nombre_valideurs,
Math.If(H <= Gestion_valideurs[5].Horaire_fin,Gestion_valideurs[5].Nombre_valideurs,0))))))
    
```

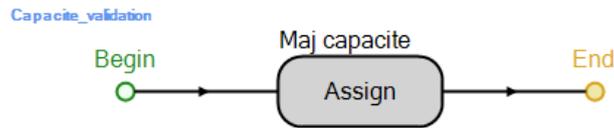


Figure 8.20 : Vue du processus « Capacite_validation »

Vient ensuite le processus permettant de sélectionner un valideur pour examiner la prescription. De base, cette fonction est intégrée au serveur mais dans notre cas elle ne se révèle pas assez flexible pour nous convenir d'où l'utilisation de ce processus. En effet, la seule méthode pour arriver au même résultat que ce processus est de coupler la sélection d'un travailleur avec un planning, chose que nous voulons éviter impérativement, afin de pouvoir contrôler à notre guise ces plannings. Nous reviendrons sur ce point plus longuement avec les tables de données.

Ce processus se nomme donc « *Selectionvalideur* » (Figure 8.21) et prend place dans « Processing ». Ce processus est constitué d'un enchaînement de tests composé d'un bloc « Decide » et d'un bloc « Seize » (les blocs en bleu). A chacun de ces blocs le processus teste si un des valideurs du type associé est disponible. Ce test concerne l'heure à laquelle la demande est faite. Par exemple : `DateTime.Hour(TimeNow) >= Gestion_valideurs[1].Horaire_debut && DateTime.Hour(TimeNow) < Gestion_valideurs[1].Horaire_fin`

Dans ce cas si l'heure intégrée au modèle est comprise entre les horaires de début et de fin de travail du valideur de type 1, alors si l'un d'eux est disponible, il commencera à examiner la prescription. Si aucun d'eux n'est libre, alors la prescription patientera. Enfin, si disponibilité il y a, le valideur est assigné et l'heure à laquelle il sera réquisitionné sera enregistrée (bloc rouge) : `Prescriptions.Heure_affectation_valideur = H`.

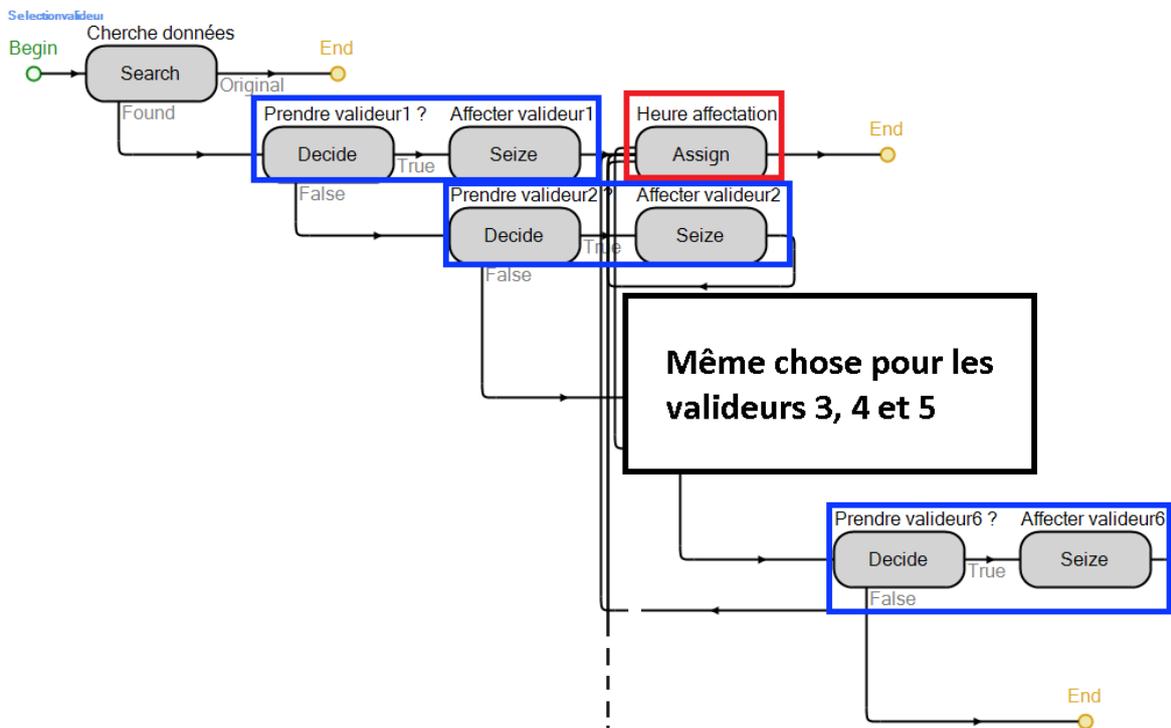


Figure 8.21 : Vue du processus « Selectionvalideur »

Pour terminer, le dernier processus est bien entendu la libération du valideur réquisitionné pour examiner la prescription. Ce processus (Figure 8.22) consiste simplement en la libération de la dernière ressource mobilisée, par la prescription dans notre cas, à l’aide d’un bloc « Release ». Ce processus se nomme « *Liberer_valideur* ».

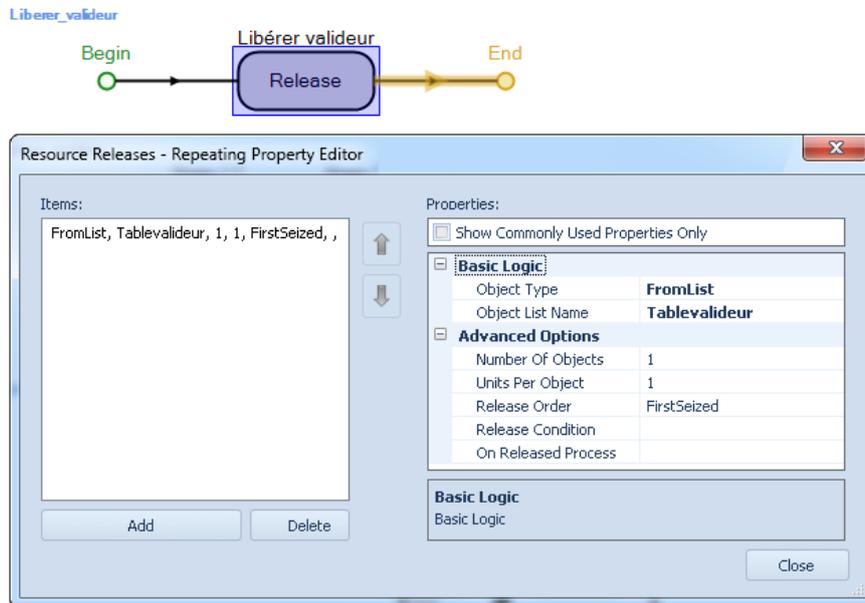


Figure 8.22 : Vue du processus « liberation_valideur »

8.3.8 La distribution des prescriptions

En sortie de l’étape de validation, ou à la suite directe de la séparation des prescriptions, il est désormais temps de s’attaquer à la préparation des prescriptions. Néanmoins, avant de se mettre de suite à cette tâche, le modèle de simulation intègre encore une étape avant celle-ci. Cette étape consiste en la distribution des prescriptions sur les divers pôles de préparation et en la possible attribution d’un préparateur dédié. Bien que la première partie ne représente pas la réalité, elle s’avère nécessaire pour le bon fonctionnement du modèle. En effet, une fois prête à être préparées, ce bloc enverra les prescriptions vers les stockeurs, les frigos, la zone de picking manuelle, mais aussi vers les robots. Dans le cas des trois premières zones suscitées, il sera nécessaire de réquisitionner un préparateur pour que la prescription puisse être préparée.

8.3.8.1 Ses propriétés

Ce bloc (Figure 8.23) est composé d’un nœud spécial « *Choix_mode_preparation* » et d’un serveur « *Assignment_prep_apres_valid* ». Tandis que le nœud est sollicité par toutes les prescriptions sortant de la validation et celles n’étant pas préparées manuellement, le serveur ne sera utilisé que pour les prescriptions nécessitant elle une préparation non robotisée.

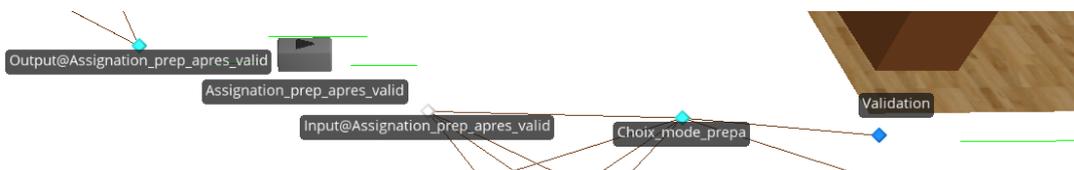


Figure 8.23 : Vue du bloc de répartition des prescriptions dans le modèle

Ces deux objets n'ont pas subi de modifications dans leur conception. En effet, toutes les propriétés de ces deux éléments sont celles par défaut, à l'exception du « Processing Time » du serveur qui est égal à zéro. Cela s'explique par le fait, que ce serveur va servir ici uniquement pour sa gestion des processus « add-on », seul véritable changement apporté à ce bloc du modèle.

8.3.8.2 Les données nécessaires à son exécution

Tout comme le bloc validation, les données qui vont être utiles seront celles qui permettront aux divers processus de s'exécuter. Nous allons donc de nouveau présenter dans le Tableau 8.7 ces diverses informations :

Tableau 8.7 : Données nécessaires pour la distribution des prescriptions

Utilisation	Nom	Rôle
Processus		
« Choix_mode_prepa »		
	Prescriptions. Modetraitementdemande	Voir partie 8.3.5.2
	Horaire_debut_cycle1	Définie l'horaire du début du cycle de préparation automatisée numéro « 1 » (même chose pour 2 et 3)
	Horaire_fin_cycle1	Définie l'horaire du fin du cycle de préparation automatisée numéro « 1 » (même chose pour 2 et 3)
	Cycle1_nombre_robot	Définie le nombre de robot de préparation fonctionnant suivant ce cycle (même chose pour 2 et 3)
Processus		
« Creerpreparateur »		
	Donneessimulation[1]. Debut_journee	Définie le début de la journée de travail de la pharmacie (ou des préparateurs)
	Donneessimulation[1]. Fin_journee	Définie la fin de la journée de travail de la pharmacie (ou des préparateurs)
	Donneessimulation. Nbrpreparateurmax	Définie le nombre de préparateurs maximum pouvant exister dans le système
	Nombre_prepa_dispo	Définie le nombre de préparateurs actuellement disponible donc sans tâche active
Processus		
« Selection_prepa »		
	Donneessimulation[1]. Planning_preparateur	Définie si oui (égal à « 1 ») ou non (égal à « 0 ») des plannings de préparation sont définis
	Donneessimulation[1].T1	Définie l'heure correspondant à la tranche T1 (par exemple T1 = 6 heures)
	Planning_G	Définie si des préparateurs sont attribués à des services en particulier
	Table_prepa. Service_prepa	Liste les services associés à chaque préparateur
	Table_prepa[Jour_prepa + Tranche_horaire]. Prepa_dispo_jour	Définie quels préparateurs sont disponibles sur chaque tranche horaire et cela pour chaque jour de la semaine

Les processus de ce bloc sont en grande partie dédiés à la recherche et à l’affectation des ressources (robot ou préparateur) qui vont préparer les prescriptions. Ces processus se basent de nouveau sur des tests temporels pour rechercher les affectations à faire et vérifient à chaque nouvelle incrémentation la disponibilité ou non des ressources demandées.

8.3.8.3 Ses processus

Abordons maintenant les processus qui utilisent les données précédentes pour fonctionner. Le premier processus que nous allons aborder est celui lié au nœud « *Choix_mode_preparation* » qui comme son nom l’indique permet de répartir les prescriptions suivant leur mode de préparation. Ce processus est réellement le premier d’une catégorie de processus bien plus complexe (en terme de nombres de blocs différents et de formules pour les tests) qui seront rencontrés à chaque fois qu’une décision doit être prise quant à la définition de la destination de l’entité parcourant le nœud (une prescription ou une demande).

Ce processus (Figure 8.24) est composé de deux scénarios différents. Le premier intervient quand le test associé au bloc rouge est négatif. Dans ce cas « *Prescriptions.Modetraitementdemande* » est différent de « 1 », et donc la préparation n’est pas automatisée. La prescription est alors envoyée vers le serveur – bloc « *Transfer* » en jaune – pour se voir attribuer un préparateur grâce aux deux processus qui seront décrits par la suite.

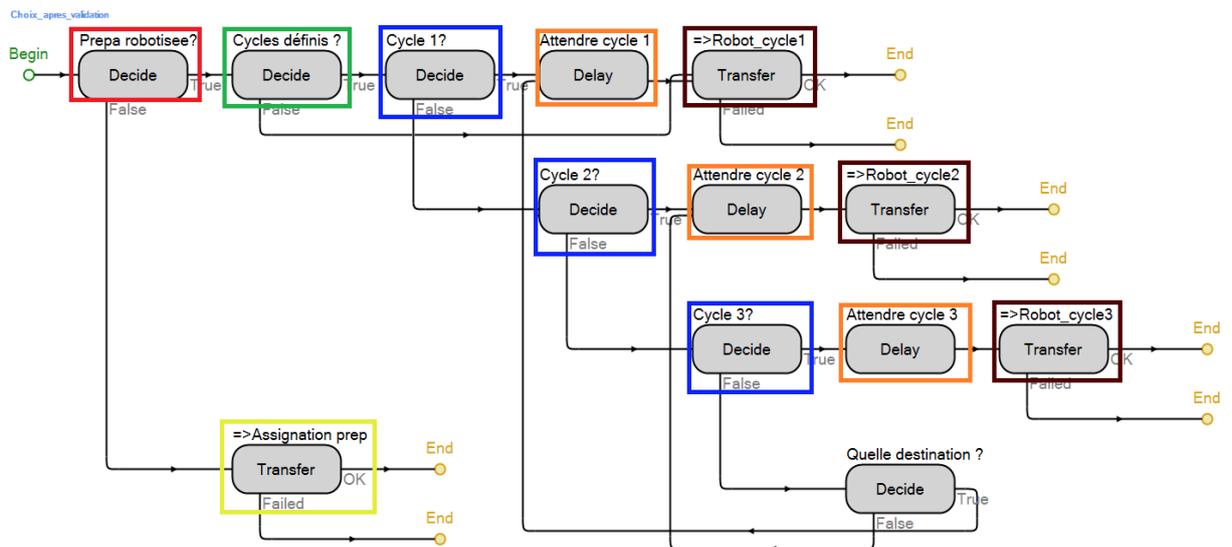


Figure 8.24 : Vue du processus « Choix_mode_prepa »

Si le test du bloc rouge s’avère vrai, alors une nouvelle série de tests va permettre de déterminer à quel robot, la prescription va être envoyée. Si le test du bloc vert lui succédant est vrai « *Cycle_existant = 1* » alors différents cycles de préparations seront testés pour savoir où envoyer la prescription. S’il s’avère faux (pas de cycles définis), la prescription est envoyée automatiquement au robot du cycle 1 correspondant au bloc « *Transfer : => Robot_cycle1* » en haut à droite.

Pour tester les différents cycles, deux caractéristiques seront évaluées, à savoir, est-ce que le cycle est actif (inéquation ci-après) et est-ce que des robots fonctionnent suivant ce cycle (« *Cycle1_nombre_robot > 0* » pour le robot 1. De même pour les 2 et 3). A noter que, par définition le

cycle 1 sera toujours le *premier* cycle ayant lieu dans une journée de travail, suivi des cycles 2 et 3, ce qui amène à avoir des formules de tests différentes dans les blocs bleus :

- ❖ Cycle 1 : le plus complexe car il faut comparer les horaires du cycle 1 à l'heure du modèle et aux horaires du cycle 3 pour éviter les chevauchements :

```
(H <= Horaire_fin_cycle1 && Horaire_fin_cycle3 <= H && Horaire_fin_cycle3 <= Horaire_debut_cycle1)
//
H >= Horaire_fin_cycle3 && Horaire_debut_cycle1 <= Horaire_debut_cycle3)
//
(Horaire_debut_cycle1 < Horaire_debut_cycle3 && Horaire_fin_cycle1 < Horaire_fin_cycle3 && H <=
Horaire_fin_cycle1)
&& Cycle1_nombre_robot > 0
```

- ❖ Cycle 2 : le cycle 2 possède le test le plus simple car le cycle 1 est forcément passé et seul son horaire de fin est à prendre en compte :

```
H <= Horaire_fin_cycle2 && Cycle2_nombre_robot > 0
```

- ❖ Cycle 3 : pour appliquer le cycle 3, il faut vérifier que le cycle 2 soit terminé et que le cycle 3 ne le soit pas encore :

```
H >= Horaire_fin_cycle2 && Cycle3_nombre_robot > 0)
//
H <= Horaire_fin_cycle3 && Cycle3_nombre_robot > 0)
```

Une fois ces tests réalisés, nous connaissons vers quel cycle la prescription sera envoyée. Dès lors un temps d'attente peut lui être imposé si le cycle en question n'a pas commencé. C'est le rôle des trois blocs orange qui font patienter la prescription d'un temps différent suivant le cycle voulu :

- ❖ Temps attente du cycle 1 :

```
Math.If( Horaire_debut_cycle1 <= H && Horaire_fin_cycle1 >= H, 0,
Math.If(Horaire_debut_cycle1 < H, 24+Horaire_debut_cycle1 - H, Horaire_debut_cycle1 - H))
```

- ❖ Temps attente du cycle 2 :

```
Math.If(Horaire_debut_cycle2 <= H && Horaire_fin_cycle2 >= H, 0,
Math.If(Horaire_fin_cycle2 < H, 24 + Horaire_debut_cycle2 - H, Horaire_debut_cycle2 - H))
```

- ❖ Temps attente du cycle 3 :

```
Math.If( Horaire_debut_cycle3 <= H && Horaire_fin_cycle3 >= H, 0,
Math.If( Horaire_fin_cycle3 < H, 24 + Horaire_debut_cycle3 - H, Horaire_debut_cycle3 - H))
```

Les trois formules précédentes permettent donc de calculer le temps restant à attendre avant que le cycle de production automatisée adéquat ne commence.

Enfin les trois blocs marron servent aux transferts des prescriptions vers les robots de production. Ces transferts peuvent s'apparenter à l'envoi informatique des prescriptions informatisées vers la mémoire du robot en vue de leur production.

Le second processus de ce bloc est plus simple que celui que nous venons de présenter. Il consiste en la « création » de préparateurs quand le besoin s’en fait sentir dans le système. Comme nous allons le voir cette création n’est pas infinie et est soumise à certaines contraintes.

Ce processus se nomme « *Creerpreparateur* » (Figure 8.25) et débute par un contrôle – bloc rouge – pour savoir si la pharmacie est actuellement dans une période de travail, ce qui signifie que l’heure du modèle est entre les horaires de début et de fin de journée définies par l’utilisateur. Cela se traduit par la formule :

$$H \geq \text{Donneessimulation}[1].\text{Debut_journee} \ \&\& \ H < \text{Donneessimulation}[1].\text{Fin_journee}$$

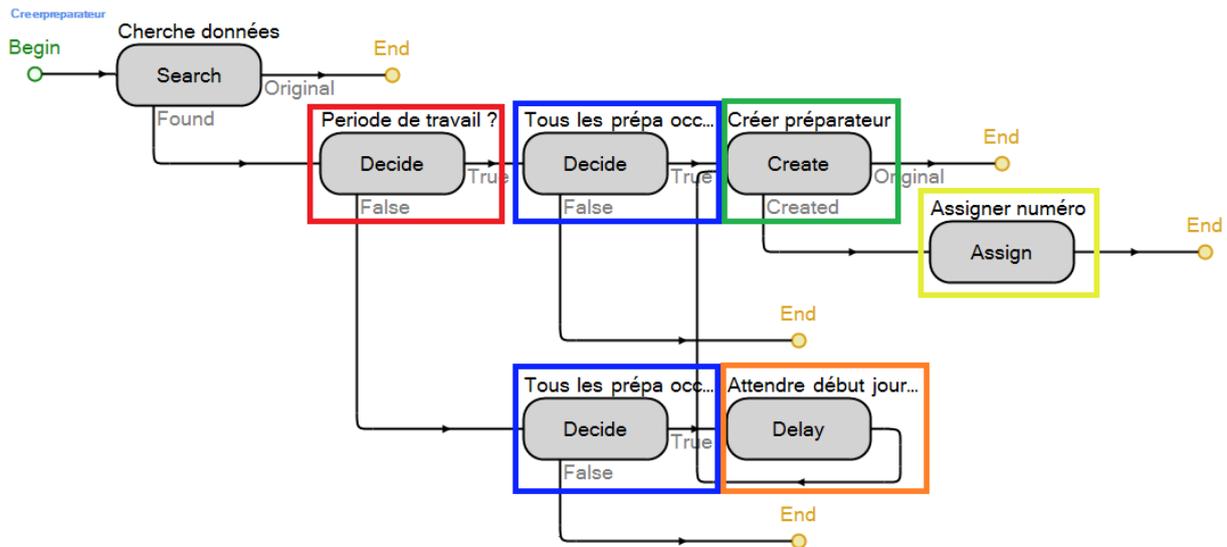


Figure 8.25 : Vue du processus « *Creerpreparateur* »

Dans les deux cas (hors période de travail ou pas), l’étape suivante consiste en un contrôle de l’occupation des préparateurs et du nombre de ces derniers. En effet, si tous les préparateurs sont occupés et si leur nombre maximum n’est pas atteint alors un nouveau préparateur sera créé. Si le nombre maximal est atteint alors la prescription (ou la demande, ce processus étant utilisé ailleurs) devra patienter que l’un d’eux se libère.

L’impact au niveau de la période de travail se fera sentir sur cette création, étant donné qu’une demande de création, sera différée dans le temps pour attendre le début de la prochaine session de travail. Il s’agit du bloc « *Delay* » en orange qui utilise la formule suivante pour différer la création :

$$\text{Math.If}(\text{DateTime.Hour}(\text{TimeNow}) < \text{Donneessimulation}[1].\text{Debut_journee}, \\ \text{Donneessimulation}[1].\text{Debut_journee} - H + 0.01, 24 + \text{Donneessimulation}[1].\text{Debut_journee} - H + 0.01)$$

La création du préparateur sera réalisée par le bloc « *Create* » en vert, tandis que le bloc « *Assign* » en jaune, permet d’associer un numéro au nouveau préparateur créé « *Worker.Priority = Preparator.Population.NumberInSystem* », ainsi que de mettre à jour le nombre de préparateur dans le système « *Nombre_prepa_dispo = Nombre_prepa_dispo+1* ».

Le dernier processus de ce bloc renoue quelque peu avec la complexité. En effet, il est cette fois question de sélectionner le préparateur adéquat et cela peut être fait selon plusieurs critères différents. Ce processus se nomme « *Selection_prepa* » (Figure 8.26) et commence par un test pour

savoir si oui ou non des plannings de présence préparateur existent (bloc rouge « Decide »). Si ce n'est pas le cas (variable égale à zéro), alors il sera associé aléatoirement un préparateur à la prescription/demande qui en requiert un.

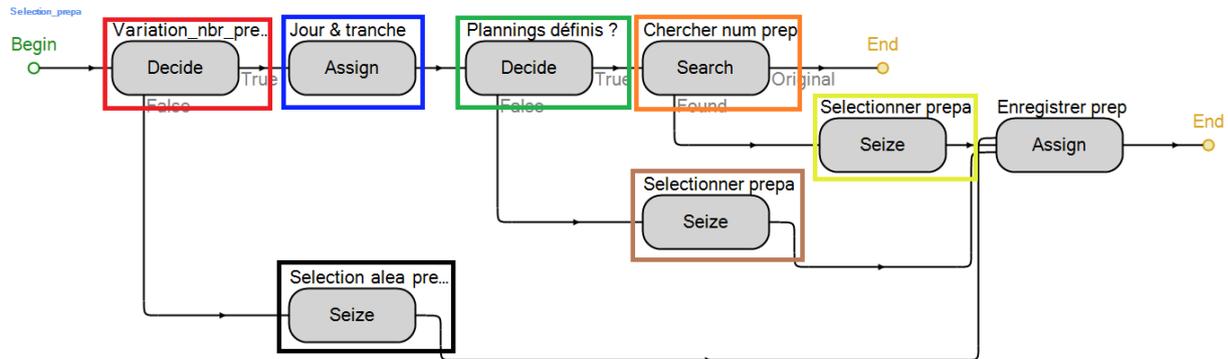


Figure 8.26 : Vue du processus de sélection des préparateurs « Selection_prepa »

Si des plannings préparateur sont définis alors, le processus va commencer par contrôler la tranche horaire et le jour correspondant au moment où la requête d'un préparateur est faite. Il s'agit du bloc « Assign » en bleu. Le jour défini par « Jour_prepa » est obtenu par la formule suivante, avec la valeur « 2 » pour le lundi, « 3 » pour le mardi, etc. :

```
Math.If(DateTime.DayofWeek(TimeNow) == 2, 0, Math.If(DateTime.DayofWeek(TimeNow) == 3, 6,
Math.If(DateTime.DayofWeek(TimeNow) == 4, 12, Math.If(DateTime.DayofWeek(TimeNow) == 5, 18,
Math.If(DateTime.DayofWeek(TimeNow) == 6, 24, 30))))
```

Tandis que la tranche horaire (*Tranche_horaire*) l'est par :

```
Math.If( H < Donneessimulation[1].T1, 1, Math.If( H < Donneessimulation[1].T2, 2, Math.If( H <
Donneessimulation[1].T3, 3, Math.If( H < Donneessimulation[1].T4, 4, Math.If( H < Donneessimulation[1].T5,
5, Math.If( H < Donneessimulation[1].T6, 6, 0))))))
```

Une fois ces deux valeurs connues, un nouveau bloc « Decide » (en vert) permet d'orienter le processus. S'il existe des préparateurs dédiés à certains services alors ce test sera vrai – « *Planning_G=1* » – et dans ce cas, le bloc « Search » en orange, récupérera le numéro du service associée à la prescription/demande, cherchera dans la table « *Table_prepa* » et enregistrera la ligne correspondante (Sortie « Found » du bloc « Search »). Dès lors le bloc jaune « Seize » permettra de sélectionner dans cette ligne, et grâce aux deux valeurs calculées plus haut, le bon préparateur, grâce à la formule :

```
String.Contains( Table_prepa[Jour_prepa+Tranche_horaire].Prepa_dispo_jour,
String.Format("[{0}]",Candidate.Worker.Priority))==1
&& Candidate.Worker.Priority==Prescriptions.Numero_prepa_necessaire
```

Les deux premières lignes de cette formule permettent de vérifier que le préparateur sélectionné est bien en service sur le créneau souhaité et la dernière ligne vérifie que le préparateur sélectionné est bien celui requis par la prescription/demande. Cette formule teste donc un à un chaque préparateur du système.

Dans le cas où aucune association service/préparateur ne serait faite, le bloc « Seize » en marron serait alors sollicité et il y aurait une sélection aléatoire de l'un des préparateurs en service sur le créneau horaire souhaité grâce à la formule :

```
String.Contains( Table_prepa[Jour_prepa+Tranche_horaire].Prepa_dispo_jour,
String.Format("{0}", Candidate.Worker.Priority)) == 1
```

8.3.9 La préparation robotisée

Faisant suite directe au nœud « *Choix_mode_preparation* », la préparation robotisée repose comme nous l'avons déjà dit sur trois types de cycles différents. Revenons quelque peu sur ce fait, une certaine subtilité prenant place ici. En effet, il aurait été possible de raisonner en terme de robots et non pas en terme de capacité de production pour modéliser cette préparation. Cependant utiliser un tel raisonnement aurait limité les capacités de simulation, un robot ne pouvant fonctionner que sur un cycle, tandis qu'un cycle peut être utilisé, suivi par plusieurs robots, d'où notre choix.

Afin d'apporter plus de flexibilité dans les possibilités de simulation, trois types de cycles sont possibles et paramétrables comme nous le verrons dans les propriétés. Chaque cycle est modélisé par un robot sous SIMIO (Figure 8.27), étant chacun un serveur standard possédant un nœud d'entrée et de sortie.

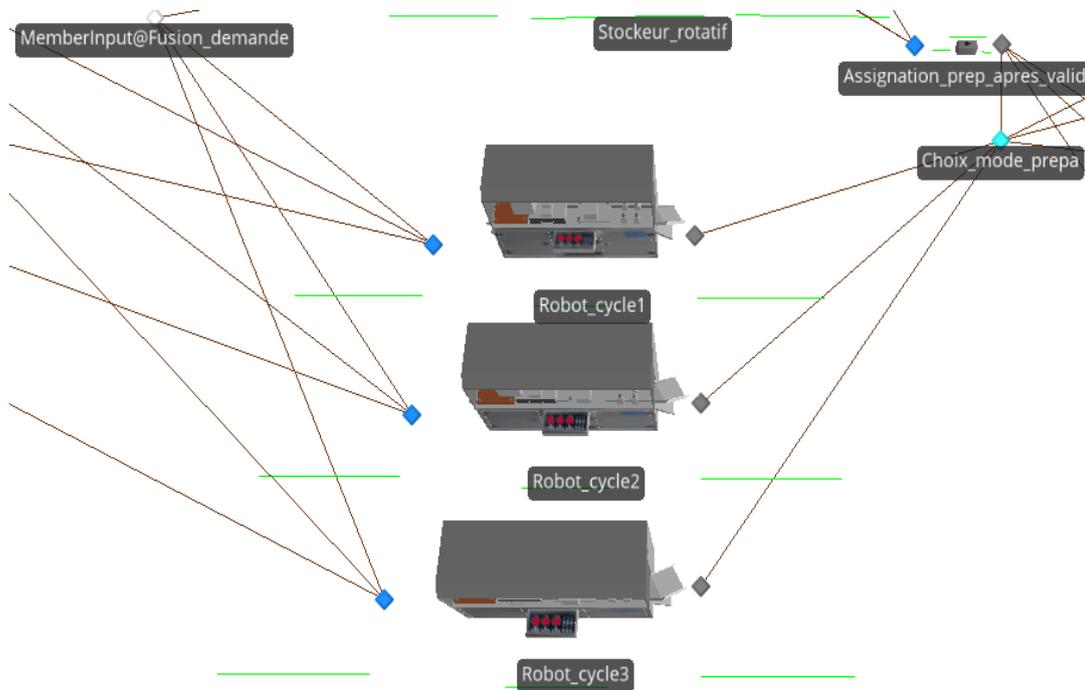


Figure 8.27 : Vue des serveurs représentant chacun un cycle de préparation

8.3.9.1 Les propriétés des cycles/robots

Les propriétés des serveurs et des nœuds sont globalement ici les propriétés par défaut de ces éléments. Les seules qui changent sont les propriétés « *Processing Time* » et « *Transfer-In Time* » de

la catégorie « Process Logic » des serveurs. En effet, ces valeurs font référence au temps de préparation de la prescription et au temps d'attente de celle-ci. Chacune de ces propriétés est définie par une formule mathématique, ce qui donne dans le cas du serveur « Robot_cycle1 » :

- ❖ Temps de production : Dépendant de la cadence du cycle et du nombre de produits et d'éléments de la prescription :

$$((Prescriptions.Nbrproduitdemande * Prescriptions.Nombre_elmt_par_produit) * 60) / Cadence_cycle_1$$

- ❖ Temps d'attente avant préparation : Suivant que l'arrivée de la prescription se fasse avant ou après le cycle, la formule du calcul du temps ne sera pas la même (troisième et dernière ligne de la formule suivante). Si la prescription arrive avant le cycle elle attendra « *Horaire_debut_cycle1 - H* », et « *24 - H + Horaire_debut_cycle1* » si elle arrive après :

$$\begin{aligned} & \text{Math.If}((\text{Cycle1_nombre_robot} + \text{Cycle2_nombre_robot} + \text{Cycle3_nombre_robot}) > 0, \\ & \quad \text{Math.If}(\mathbf{H} \geq \text{Horaire_debut_cycle1} \ \&\& \ \mathbf{H} < \text{Horaire_fin_cycle1}, 0, \\ & \text{Math.If}(\mathbf{H} \leq \text{Horaire_debut_cycle1}, \text{Horaire_debut_cycle1} - \mathbf{H}, 24 - \mathbf{H} + \text{Horaire_debut_cycle1})), 0) \end{aligned}$$

8.3.9.2 Les données nécessaires à son exécution

Les seules données requises en plus de celles utiles dans les deux formules ci-dessus sont celles utilisées dans le processus des nœuds de sorties de ces trois serveurs. Le Tableau 8.8 présente ces diverses données :

Tableau 8.8 : Données utilisées en sortie des serveurs de préparation robotisée

Utilisation	Nom	Rôle
Processus		
« Choix_apres_preparation »		
	Prescriptions.Entite_hybride	Voir partie 8.3.5.2
	Prescriptions.Demande_Parent	Voir partie 8.3.5.2
	Prescriptions.Indicedemande	Voir Tableau 8.1
Propriétés		
« Logique de production » des serveurs		
	Prescriptions.Nbrproduitdemande	Voir Tableau 8.1
	Prescriptions.Nombre_elmt_par_produit	Voir partie 8.3.5.2
	Cadence_cycle_1*	Définie le nombre de doses par heure pouvant être préparées par les divers robots de dispensation fonctionnant sur ce cycle
	Horaire_debut_cycle1*	Voir Tableau 8.7
	Horaire_fin_cycle1*	Voir Tableau 8.7

*La même chose s'applique pour les cycles 2 et 3

L'unique donnée mise à jour en sortie de ces serveurs est la priorité de la prescription (*Prescriptions.Priority*) qui est désormais égale à « 2 ».

8.3.9.3 Les processus

L'arrivée des prescriptions étant régie par le nœud « *Choix_mode_preparation* », les serveurs n'ont ici que la préparation à réaliser, et ne possède dans ce cas aucun processus. Cependant leurs nœuds de sortie devant orienter les prescriptions selon qu'elles soient seules ou faisant partie d'un groupe de demandes/prescriptions, ces derniers posséderont un processus dédié à cette tâche.

Ce processus se nomme « *Choix_apres_preparation* » (Figure 8.28), sera utilisé à la sortie de tous les serveurs de préparation du modèle et peut orienter les entités passant par ces nœuds sur quatre voies différentes. En effet, le premier bloc « *Decide* » en rouge permet de définir si oui ou non la demande fait partie d'un groupe de demandes/prescriptions (*Prescriptions.Entite_hybride=1*). Dans ce cas, un second bloc « *Decide* » – en orange – prend le relais et vérifie si la demande/prescription est le parent ou un sous élément de ce groupe (*Prescriptions.Demande_Parent = 1*). Si tel est le cas, alors la prescription/demande est envoyée dans la file d'attente des parents, sinon dans celles des enfants.

Si la demande/prescription était un élément unique, c'est-à-dire ne faisant pas partie d'un groupe, alors elle est envoyée vers le bloc « *Decide* » en vert pour déterminer si par hasard, il s'agirait de produits revenants des services (*Prescriptions.Indicedemande=100*), pouvant parfois être remis en stock. Sinon, la demande/prescription est envoyée au service logistique pour livraison.

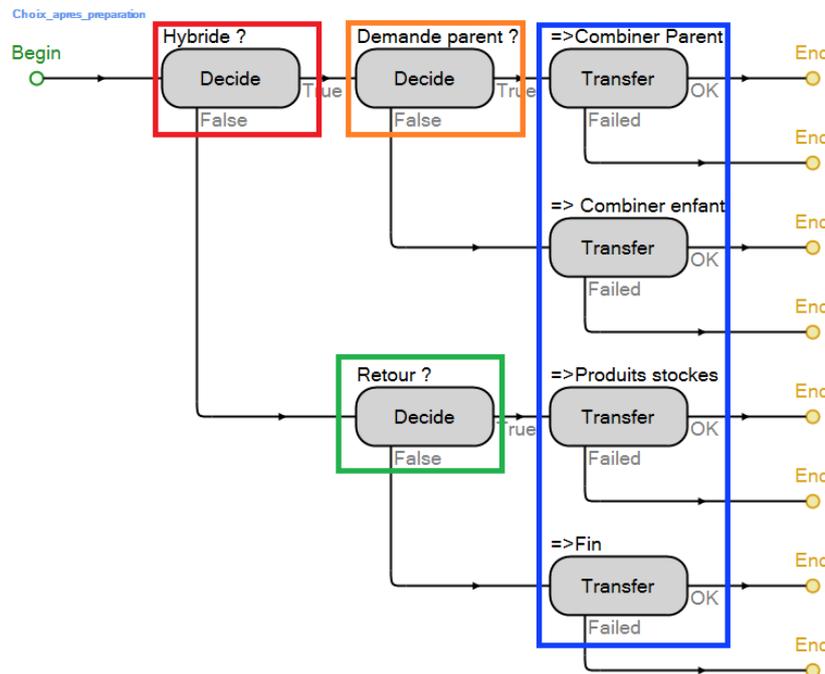


Figure 8.28 : Vue du processus « *Choix_apres_preparation* »

Nous en avons donc terminé avec le processus de préparation automatisé des prescriptions. Les autres procédés de préparation étant commune aux demandes standards et prescriptions, nous allons reprendre de zéro le processus de préparation des demandes afin de l'étudier.

8.3.10 Edition et contrôle des demandes

Nous voici donc au début du processus de préparation des demandes une fois le tri fait entre toutes les entités arrivant dans le modèle ([partie 8.3.5](#)). Ce bloc (Figure 8.29) a une logique très similaire à la validation des prescriptions, exception faite que ce contrôle est obligatoire pour toutes les demandes de services.

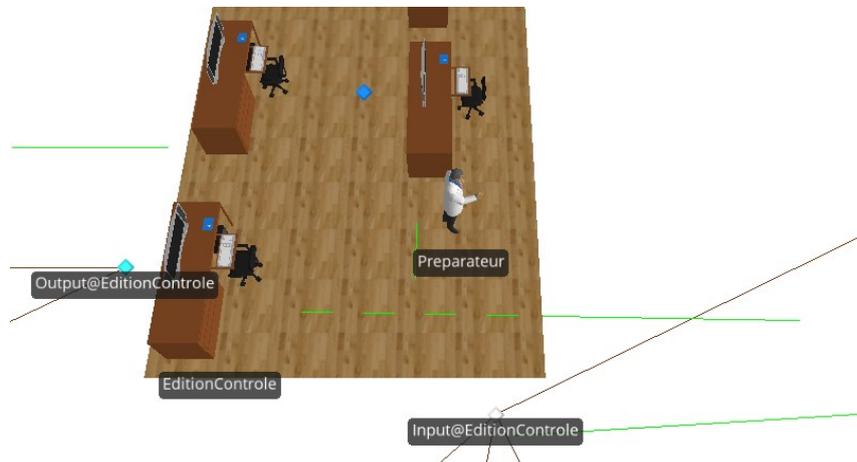


Figure 8.29 : Vue du bloc « Edition/Contrôle »

Toutes les demandes arrivent donc au nœud d'entrée de ce bloc « *Input@EditionContrôle* » avant de patienter jusqu'à ce qu'un préparateur s'occupe de les éditer et d'en contrôler le contenu. Une fois cela fait, la demande sort de ce bloc par le nœud de sortie « *Output@EditionContrôle* » où elle sera orientée dans le processus. Alors que les nœuds ont subi peu de modifications par rapport à la configuration par défaut, l'édition et le contrôle sont modélisés une fois de plus par un serveur qui a subi certains changements. Comme nous le verrons dans ce qui suit, un travail a été fait au niveau de la gestion de la capacité de travail de ce bloc, ainsi qu'au niveau de la gestion des préparateurs.

8.3.10.1 Les propriétés de ce bloc

En plus des processus, seules trois propriétés du serveur sont différentes de la configuration par défaut. Il s'agit de :

- ❖ La capacité initiale du serveur (*Initial Capacity*) qui est mise à zéro ;
- ❖ Du temps d'attente (*Transfer-In Time*) mise aussi à zéro ;
- ❖ Du temps de contrôle et d'édition de la demande (*Processing Time*) défini par :

$$\text{Math.Abs(Random.Normal(Duree_moyenne_controle, 0.1))}$$

L'action réalisée par le ce bloc n'est donc pas difficile en elle-même. Il sera vu dans la partie processus que la réelle difficulté de cette partie était la gestion de l'allocation des préparateurs. Mais tout d'abord faisons le tour des données nécessaires à son exécution.

8.3.10.2 Les données du serveur et des processus

Trois, voire quatre processus sont sollicités dans ce bloc en plus de l'unique variable du serveur qui concerne le temps de contrôle d'une demande permettant d'en valider le contenu. Le Tableau 8.9 suivant fait un bilan de ces diverses informations :

Tableau 8.9 : Données nécessaires pour le serveur « EditionContrôle »

Utilisation	Nom	Rôle
Propriété « Logique de production » du serveur	Duree_moyenne_controle	Voir partie 8.3.5.2
Processus « Capacite_edition »	Donneessimulation[1]. Debut_journee	Voir Tableau 8.7
	Donneessimulation[1].T1	Voir Tableau 8.7
	Donneessimulation[1]. Planning_preparateur	Voir Tableau 8.7
	Donneessimulation. Nbrpreparateurmax	Voir Tableau 8.7
	Table_prepa[Jour_prepa + Tranche_horaire]. Prepa_dispo_jour	Voir Tableau 8.7
	Nombre_prepa_dispo	Voir Tableau 8.7
	Donneessimulation[1]. Fin_journee	Voir Tableau 8.7
Processus «Attente_select_prepa »	Nombre_prepa_dispo	Voir Tableau 8.7
	Planning_G	Voir Tableau 8.7
	Preparateur[Prescriptions.Numero_prepa_necessaire]. ResourceState	Définie le statut occupé (égale à « 1 ») ou disponible d'un préparateur (égale à « 0 »)
Processus «Choix_apres_edition »		
	Prescriptions.Indicedemande	Voir Tableau 8.1

Globalement, les données demandées ici sont déjà communes à d'autres processus. Ce constat va commencer à être de plus en plus évident étant donné que tout du long de la construction de notre modèle nous avons tenté de rationaliser les diverses données et de les limiter au maximum. Ainsi nombre de processus ou de serveurs réutilisent des données déjà sollicitées ailleurs, et même pour les premiers des pans entiers d'autres processus.

8.3.10.3 Ses processus

Comme dit précédemment les processus de ce bloc sont au nombre de trois. Cependant l'un d'eux fait appel à un processus déjà présenté, ce qui revient à dire que quatre processus sont utilisés ici.

Par ordre d'utilisation le premier à être sollicité est « *Capacite_edition* » (Figure 8.30), qui comme son nom l'indique gère la capacité de travail du bloc « EditionContrôle ». Cette gestion de la capacité de travail, revient en fait à gérer le nombre de préparateurs présents à tout instant dans le système. En effet, un préparateur qui contrôle une demande devra la préparer dans la foulée afin

d'avoir une continuité dans la préparation et assurer une meilleure sécurité. Du coup si seuls sept préparateurs contrôlent et éditent des demandes, seuls sept préparateurs pourront les préparer.

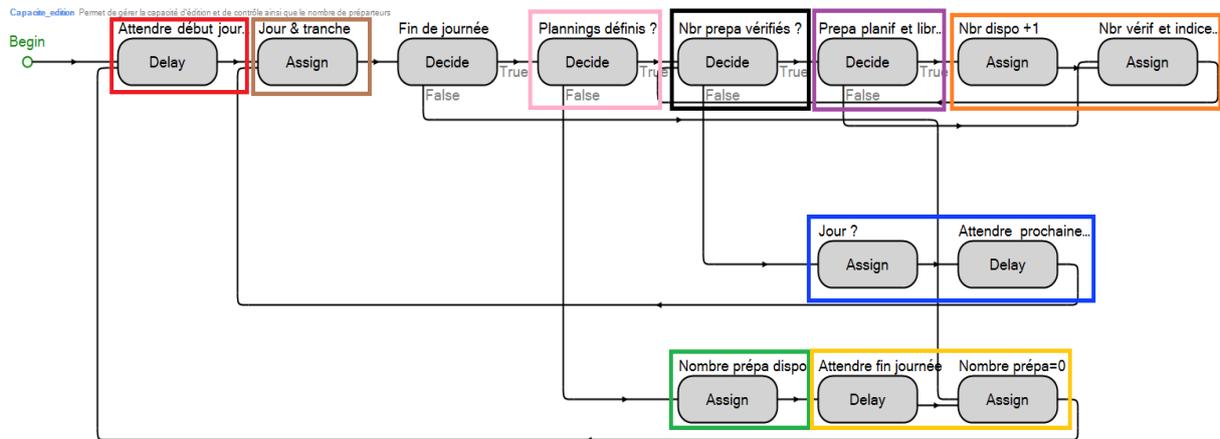


Figure 8.30 : Vue du processus « Capacite_Edition »

Contrairement aux valideurs dont le nombre ne varie pas en permanence, il est nécessaire ici de recalculer sur chaque créneau horaire le nombre de préparateurs et leur disponibilité.

Utilisons une description sous la forme d'un algorithme pour en comprendre le fonctionnement :

- ❖ Bloc rouge : Evaluation du temps restant avec le début de la journée de travail ;
- ❖ Bloc marron : Evaluation pour déterminer le jour et la tranche horaire actuellement simulé ;
- ❖ Test pour déterminer si la fin de journée est atteinte :
 - Si oui :
 - Bloc rose : Si des plannings de présence préparateurs sont définis (*Donneessimulation[1].Planning_preparateur = 1*) :
 - Bloc noir, violet et orange : Contrôler les préparateurs présents sur la tranche horaire simulée ;
 - Bloc bleu : Mettre le nombre de préparateurs disponibles à jour ;
 - Bloc bleu : Mettre en attente le processus jusqu'à la prochaine tranche horaire ;
 - Répéter les actions précédentes lors de la nouvelle tranche ;
 - Sinon
 - Bloc vert : Définir le nombre de préparateurs disponibles à son maximum : $\ll \text{EditionControle.CurrentCapacity} = \text{Donneessimulation[1].Nbrpreparateurmax} \gg$
 - Mettre le processus en attente jusqu'à la fin de journée ;
 - Mettre le nombre de préparateurs disponible à zéro ;
 - Sinon : Mettre le nombre de préparateurs disponible à zéro ;

Le second processus ici utilisé est celui permettant de différer l'affectation d'un préparateur à une demande si aucun d'entre eux n'est disponible (Figure 8.31). En effet, il a été difficile de mettre en adéquation une demande de préparateur et leur affectation, étant donné qu'une demande de

produits requérant un préparateur était automatiquement transférée dans la station de travail (de production) même si le préparateur ne lui avait pas été affecté. Du coup celle-ci bloquait une capacité de travail de la station qui aurait pu être utile à d'autre. Pour contourner ce problème nous avons donc intégré dans ce processus deux fonctions et défini ce processus comme un processus « add-on ». Il prend place juste après la requête de saisie d'un opérateur par la demande mais avant que cette dernière ne soit transférée dans la station de travail (Catégorie « Before Processing » dans « Add-on process Triggers »).

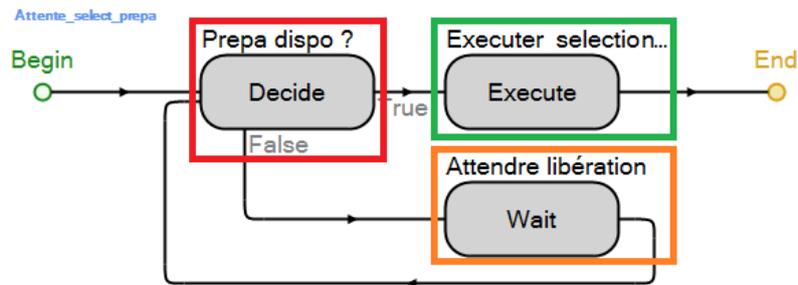


Figure 8.31 : Vue du processus d'attente d'un préparateur « Attente_select_prepa »

La première fonction est un bloc « Decide » en rouge, permettant de vérifier si un préparateur est disponible pour préparer la demande. Si indisponibilité il y a, alors la demande devra attendre qu'une autre demande ou prescription est finie d'être préparée (donc en attente pour être regroupée, mise à disposition du service logistique ou rangée) pour demander de nouveau un préparateur ou bien que la capacité de travail du bloc « EditionControle » est augmentée. Il s'agit de la seconde fonction qui consiste en l'utilisation d'un bloc « Wait » (en orange).

Dès qu'une disponibilité est validée grâce à la formule ci-dessous alors le bloc « Execute » en vert est sollicité. Ce bloc permet de faire appel à un autre processus, permettant d'éviter les redondances, qui sera dans notre cas le processus « Selectionprepa » présenté dans la **partie 8.3.8.3**.

```
Nombre_prepa_dispo > 0
&& Math.If( Planning_G==1, Preparateur[Prescriptions.Numero_prepa_necessaire].ResourceState == 0, Tampon == 0)
```

Le troisième et dernier processus de ce bloc est celui rattaché au nœud de sortie « Output@Editioncontrole », et il se nomme « Choix_apres_edition » (Figure 8.32).

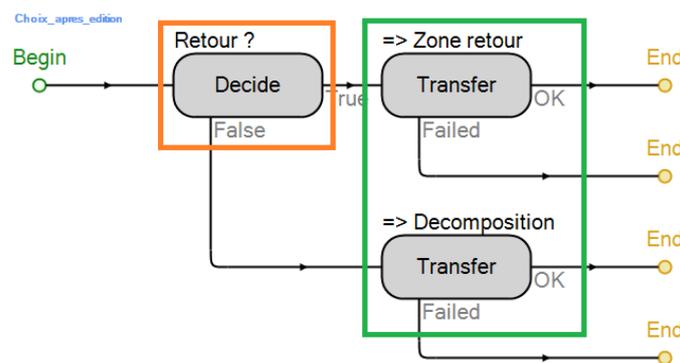


Figure 8.32 : Vue du processus « Choix_apres_edition »

Ce processus est extrêmement simple. Après avoir contrôlé que les produits ne sont pas des retours à la pharmacie « *Prescriptions.Indicedemande = 100* » par le biais du bloc « Decide » en orange, les demandes sont soit orientées en conséquence vers un nœud « *Zone_retour* » ou vers un nœud « *Decompo_hybride* » permettant de scinder les demandes en sous-demandes suivant le type de produits incorporés dans celles-ci, ce qui va être l'objet de la prochaine partie.

8.3.11 La décomposition des demandes

Comme nous venons de le dire, cette portion du modèle, commençant avec le nœud « *Decompo_hybride* » et se terminant avec le nœud « *Choix_global* », permet de décomposer une demande en une ou plusieurs sous-demandes suivant les produits qu'elle contient. Le fonctionnement est exactement le même que pour la décomposition des prescriptions. Par conséquent, nous allons présenter ce bloc de manière identique à celui des prescriptions.

La décomposition des demandes est elle aussi réalisée par trois séparateurs chacun traitant un type de produits (frigo, solutés, stockeur), et amenant à avoir une décomposition similaire à celle de la Figure 8.6 sans la présence des prescriptions. La Figure 8.33 présente ces trois séparateurs, accompagnés des divers nœuds permettant le voyage des demandes dans cette partie du modèle et des serveurs permettant l'affectation des préparateurs.

8.3.11.1 Ses propriétés

Tous les nœuds « *Decompo_hybride* » et « *ParentOutput@Duplication* » choisissent le chemin de sorti suivant le poids de celui-ci (Règle « *By Link Weight* »), avec bien entendu une variation du poids d'un chemin à l'autre. Nous présenterons les poids des différents dans la partie sur les données.

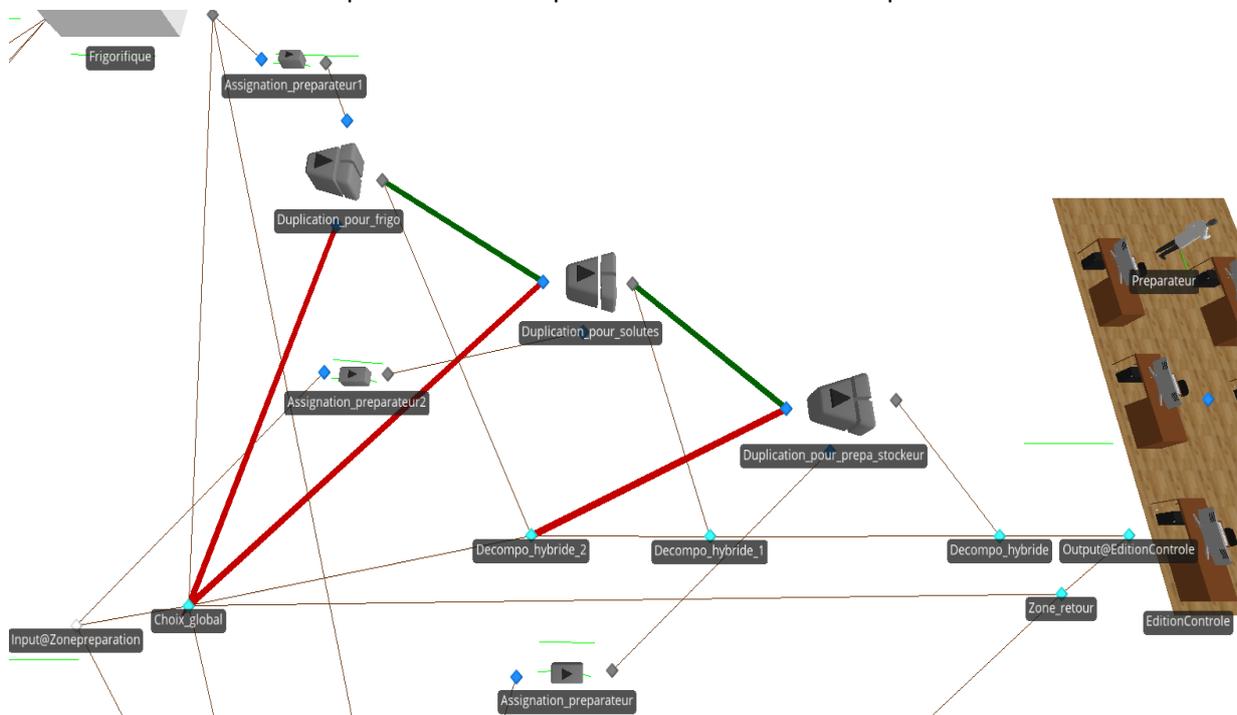


Figure 8.33 : Vue des séparateurs utilisés pour décomposer les demandes en sous-demandes

Les trois séparateurs « *Duplication_pour_solutes/frigo/prepa_stockeur* » utilisés fonctionnent exactement de la même manière que ceux des prescriptions, et possèdent d’ailleurs des propriétés très similaires. Par conséquent, comme pour les prescriptions, ils produisent chacun une copie de la demande avant de lui attribuer de nouvelles données que nous détaillerons dans la partie suivante. Les propriétés (partie « Separation Logic ») des séparateurs sont donc :

- ❖ Mode de séparation = « Faire copies » ;
- ❖ Nombre de copie = 1.

En plus de ces éléments en commun avec la portion sur les prescriptions, des serveurs permettant d’assigner des préparateurs aux sous-demandes créées peuvent être trouvés en sortie de chaque séparateur. Il aurait été possible de faire comme les prescriptions avec un unique serveur permettant d’assigner les préparateurs pour toutes les sous-demandes, mais afin d’avoir un modèle lisible et pour économiser la création d’un nouveau processus (étant limiter dans nos possibilités du fait de la version test) nous avons préféré utilisé trois serveurs d’affectations qui fonctionnent cependant d’une manière parfaitement identique. Les seuls changements des propriétés de ses serveurs nommés « *Assigant_preparateur* », « *Assigant_preparateur1* » et « *Assigant_preparateur2* » sont :

- ❖ Capacité initiale (Initial Capacity) égal à 0 mais mis à jour à « *Donneessimulation.Nbrpreparateurmax* » après l’initialisation par le processus de chargement des données;
- ❖ Règle de sélection des entités en attente (Dynamic selection rule) : « *Smallest Value first* » ;
- ❖ Valeur associée à la règle de sélection précédente (Value Expression) : « *Candidate.Prescriptions.Preparateurassocie.Worker.ResourceState* » permettant de sélectionner les entités dont les préparateurs sont disponibles (ResourceState = 0).

8.3.11.2 Les données nécessaires à son exécution

Ci-dessus, nous avons précisé que le poids de chaque chemin change selon le nœud d’origine, ce qui nécessite au moins autant de données que de chemins. Le Tableau 8.10 récapitule les différents poids de sortie des chemins.

Tableau 8.10 : Poids des chemins utilisés pour l’orientation des demandes

Nœud	Poids de sortie
Decompo_hybride	«1-Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_stockeur» et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_stockeur»
Decompo_hybride_1	« 1- Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute» et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute»
Decompo_hybride_2	« 1- Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo» et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo»
ParentOutput@Duplication_pour_prepa_stockeur	« 1- Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute» et « Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_solute»

ParentOutput@Duplication _pour_solutes	« 1- Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo» et «Donnees_demandes_hybrides[Prescriptions.Indicedemande].Pourc_frigo»
---	--

Les autres données nécessaires à l'exécution de ce bloc, sont celles requises dans le cadre de la mise à jour des prescriptions sortant de chaque séparateur. Etant donné que ce seront les mêmes données que pour la séparation des prescriptions, nous ne les présenterons que pour le séparateur créant une sous-demande de solutés :

- ❖ Pour la demande mère : Exactement les mêmes que pour les prescriptions au ajustement près dû au type de produits séparés.
- ❖ Pour les sous-prescriptions : Même chose avec en plus :
 - *Prescriptions.Prepareurassocie = Copie_prepareur_hybride* : Le préparateur nécessaire à la préparation de la demande mère est sauvegardé par « *Copie_prepareur_hybride* » lors de l'entrée de la demande dans le séparateur. Cette donnée est ensuite transmise à la sous-demande afin d'avoir le même préparateur pour les deux.

8.3.11.3 Ses processus

Cette nouvelle portion du modèle n'inclut l'utilisation que de deux processus « add-on », l'un par le nœud « *Choix_global* » pour orienter les demandes dans le système et le second par les serveurs d'affectation des préparateurs.

Le premier processus, nommé « *Choix_apres_assignment_duplication* » (Figure 8.34) et associé au nœud, fonctionne comme les nombreux processus présentés précédemment et qui servaient à orienter les entités dans le modèle. Ainsi en couplant des blocs « *Decide* » (en rouge) et des blocs « *Transfer* » (en vert) les entités seront envoyées soit vers le stockeur, soit vers le frigo ou vers la zone de picking. Ces deux « *Decide* » testeront la valeur de « *Prescriptions.Modetraitementdemande* » pour connaître le mode de préparation associé à la demande et donc définir sa destination.

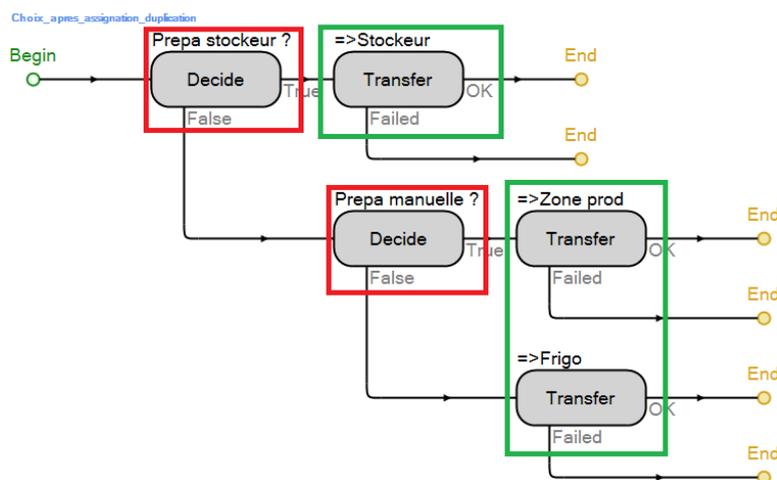


Figure 8.34 : Vue du processus « *Choix_apres_assignment_duplication* »

Le second processus, lié aux serveurs « *Assignment_prepareur* » (Figure 8.35), est des plus simples. Une fois que la demande peut saisir un préparateur, ce processus se déclenche et par le

biais du bloc « Seize » en bleu, en réquisitionne un. Pour cela il sélectionne le préparateur défini par « *Prescriptions.Prepareurassocie* ». Ensuite par le biais du bloc « Assign » en orange, il met le nombre de préparateurs disponibles à jour grâce à la formule :

$$\text{Math.If}(\text{Nombre_prepa_dispo} + 1 > \text{EditionControle.CurrentCapacity}, \text{Nombre_prepa_dispo}, \text{Nombre_prepa_dispo} + 1)$$

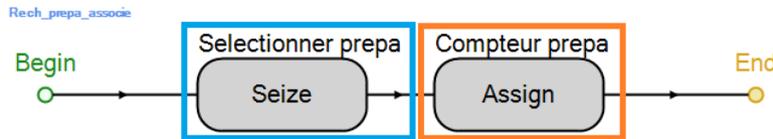


Figure 8.35 : Vue du processus « Rech_prepa_associe »

Cette étape étant passée, nous allons pouvoir aborder la préparation des demandes de chaque type.

8.3.12 La préparation des demandes

Trois objets existent dans le modèle pour simuler la préparation des demandes et des prescriptions par les préparateurs. Afin de gagner une fois de plus en simplicité et clarté, nous avons choisi trois serveurs, chacun associé à un type de produits, mais strictement identiques en termes de données, de propriétés et de processus sollicités. La Figure 8.36 présente ces trois serveurs, le premier étant associé au picking manuel, le second au stockeur mécanique et le dernier à une chambre froide.

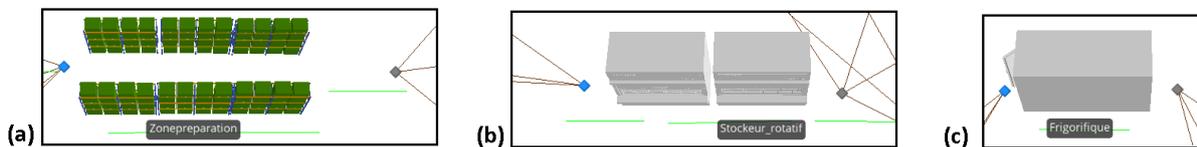


Figure 8.36 : Vue des serveurs de picking manuel (a), de stockeur mécanique (b) et de la chambre froide (c)

8.3.12.1 Leurs propriétés

Les propriétés ayant changées dans ces serveurs sont celles en lien avec la capacité et le temps de production des demandes (Catégorie « Process Logic »). Les changements de la catégorie « States Assignments » sont uniquement liés à de la mise à jour de compteur (pour le nombre de demandes), ce qui n’est pas essentiel ici.

Ainsi, les changements prenant place dans la première catégorie sont :

- ❖ Capacité initiale (*Initial Capacity*) égale à zéro ;
- ❖ Temps d’attente (*Transfer-In Time*) égal à zéro ;
- ❖ Temps de production (*Processing Time*) défini par « *Prescription.Tempsprepa* ».

8.3.12.2 Les données nécessaires à l’exécution

Contrairement à ce qui aurait pu être anticipé, le nombre de données ici nécessaires est très limité. En effet, l’utilisation des préparateurs et les capacités de production étant gérées en amont,

l'utilisation des serveurs est ici limitée à sa plus simple expression. Les données nécessaires au fonctionnement de ces serveurs sont :

- ❖ « *Prescriptions.Tempsprepa* » : le temps de préparation associé à la demande ;
- ❖ « *Nombre_prepa_dispo* » : suivi en temps réel du nombre de préparateurs disponible dans le système.

8.3.12.3 Le processus de libération

Etant en fin de chaîne de préparation, une fois les demandes et prescriptions prêtes le préparateur est donc libre d'aller en préparer une nouvelle. Pour cela, une fois la préparation terminée, le processus « *Liberation_prepa* » est exécuté. Ce processus (Figure 8.37) est de type « add-on », se place en « After Processing » et peut être vu comme le processus miroir de « *Selectionprepa* ». Par conséquent, contrairement à ce dernier le premier bloc n'est plus un « Seize » mais un « Release » en vert, accompagné une nouvelle fois par un bloc « Assign » permettant de mettre à jour le nombre de préparateurs disponibles dans le système (en rouge).

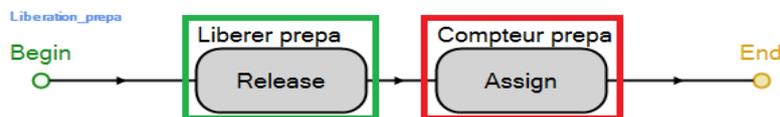


Figure 8.37 : Vue du processus de libération des préparateurs « *Liberation_prepa* »

Il y a un second processus utilisé par ces serveurs, mais celui-ci ayant été déjà présenté, nous ne le referons pas. Il s'agit du processus « *Choix_apres_preparation* » présenté dans la [partie 8.3.9.3](#).

8.3.13 Réunion, transport et envoi

Nous allons maintenant traiter la dernière partie de la chaîne de préparation en une seule fois, c'est-à-dire de la fin de la préparation à l'arrivée dans le service. Cette portion du modèle (Figure 8.38) comporte trois objets, à savoir un combineur, un serveur et un puits. Le combineur permet de réunir, regrouper les demandes/prescriptions mères avec leurs sous-demandes/prescriptions. Le serveur quant à lui représente le transport des demandes/prescriptions vers le service de soins, c'est-à-dire la logistique. Enfin le puits, représente le service de soins, et permet de faire disparaître les demandes/prescriptions du système, celles-ci étant alors supposées traitées.

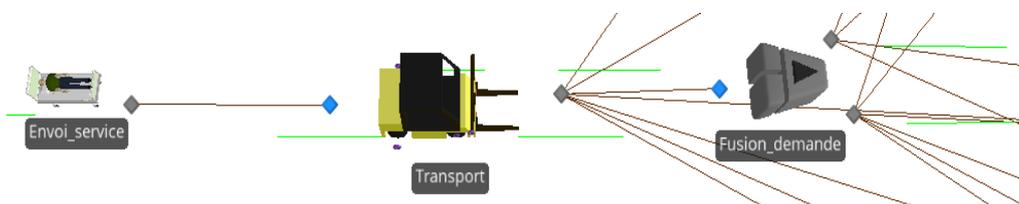


Figure 8.38 : Vue des trois éléments réunissant les demandes et les expédiant

Les demandes et prescriptions arrivent donc des différents serveurs de préparation et suivant la valeur de leurs propriétés « *Prescriptions.Entite_hybride* » et « *Prescriptions.Demande_Parent* », elles seront envoyées directement au combineur « *Fusion_demande* » ou au serveur « *Transport* ». Que les demandes/prescriptions, passent ou non par le combineur, elles passeront par le serveur « *Transport* » avant de disparaître dans le puits « *Envoi_service* ».

8.3.13.1 Les propriétés de ce bloc

Dans cette dernière portion du modèle seul le serveur et le combineur subissent des changements au niveau de leurs propriétés. Tandis que pour le combineur celles-ci seront en lien avec les aspects parent/enfant, celles du serveur correspondront aux temps de transport des demandes/prescriptions. Ainsi, nous avons :

Tableau 8.11 : Propriétés associées aux éléments de regroupement, de transport et d'expédition

Objet	Catégorie	Caractéristique	Réponse
Combineur			
« Fusion_demande »			
	Matching Logic	Batch Quantity	Prescriptions.Nombre_ss_entite_hybride
		Matching Rule	Match Members And Parent
		Member Match Expression	Prescriptions.Numero_demande
		Parent Match Expression	Prescriptions.Numero_demande
	Process Logic	Processing Time	0
Serveur « Transport »			
	Process Logic	Initial Capacity	Table_Transport[1].Nbr_agent_logistique * Table_Transport[2].Nbr_agent_logistique (après mise à jour)
		Processing time	Prescription.Temps_transport

A noter que la capacité du serveur « *Transport* » est ici fonction de deux paramètres, le nombre d'agents logistiques et la capacité de transport de ces agents. Ces deux paramètres sont donc multipliés pour obtenir la capacité de transport total du serveur.

8.3.13.2 Les données nécessaires

En plus des données nécessaires pour les propriétés des objets ci-dessus, se rajoutent les nombreuses informations qui sont sauvegardées par le processus « add-on » associé à la source. Le Tableau 8.12 présente ces diverses informations.

Tableau 8.12 : Données associées aux éléments de regroupement, de transport et d'expédition

Utilisation	Nom	Rôle
Combineur		
« Fusion_demande »		
	Prescriptions.Nombre_ss_entite_hybride	Définie le nombre de sous-demandes/prescriptions que peut avoir une demande/prescription mère
	Prescriptions.Numero_demande	Définie le numéro de la demande dans la simulation

Serveur
« Transport »

Temps_transport_defaut/spe	Définie le temps de transport vers le service des demandes/prescriptions
Nbr_agent_logistique	Nombre d'agents transportant les demandes

Processus
« Enregistrer_donnees »

Prescriptions.Numero_demande	Voir ci-dessus
Prescriptions.TimeInSystem	Définie le temps passé par la demande/prescription dans le modèle
Prescriptions.Nbrproduitdemande	Voir partie 8.3.5.2
Prescriptions.Typedemande	Voir Tableau 8.1
Prescriptions.Nombre_elmt_par_produit	Voir Tableau 8.1
Prescriptions.Tempsprepa	Voir Tableau 8.1
Prescriptions.Prepareteurassocie.Name	Définie le nom du préparateur (par exemple « Prepareteur[1] »)

Le combineur utilisera donc le nombre de sous demandes/prescriptions d'une demande/prescription pour savoir combien d'entités ce dernier aura à regrouper. Ainsi en scrutant un à un les numéros de demande des entités enfants (héréditaire quand une sous-demande/prescription est créée), il sera en mesure de les combiner avec l'entité parent adéquate.

8.3.13.3 Les processus

Deux processus sont utilisés avec cette partie du modèle. Le premier concerne l'assignation du temps de transport. Bien que ce processus soit utilisé au niveau de la source, les informations obtenues ne seront utilisées qu'ici, c'est donc pourquoi nous ne le présentons que maintenant.

Ce processus (Figure 8.39) va chercher dans la table « Table_Transport » le nom/numéro du service associé à la demande. Si la demande n'a pas de service associé alors le bloc « Search » essaiera de trouver « [0] » sinon il cherchera « [Nom/Numéro service] » dans cette table. Si cette information y est présente alors on poursuivra dans le bloc orange où il sera récupéré le temps de transport lié à ce service. Sinon, on passera par le bloc vert ou un temps de transport par défaut sera associé à la demande.

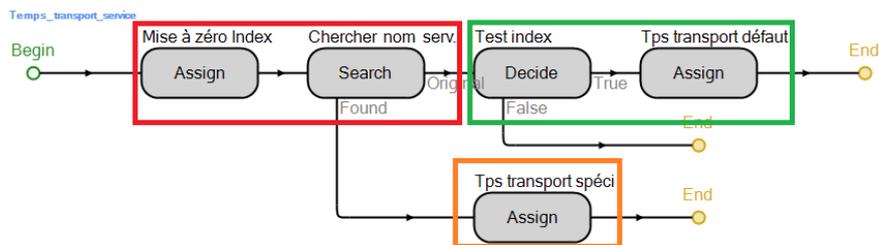


Figure 8.39 : Vue du processus recherchant le temps de transport des demandes

Le second processus est lui utilisé pour l’enregistrement des données. En effet, afin de pouvoir garder une traçabilité des diverses entités parcourant le modèle, la seule méthode s’offrant à nous était de rajouter un processus qui pour chaque entrée d’une entité dans le puits de fin, enregistre diverses informations concernant cette entité. Ainsi le processus « *Enregistrer_donnees* » (Figure 8.40) commence par mettre à jour (bloc rouge) les nombreuses variables qui vont être enregistrées et qui correspondent aux différentes propriétés présentées dans le Tableau 8.12. Ensuite, suivant que la demande/prescription est été traitée automatiquement ou manuellement – bloc « *Decide* » en orange – le processus mets à jour le nombre de lignes du fichier Excel – blocs « *Assign* » en vert – dans lequel il écrit afin de pouvoir sauvegarder – blocs « *ExcelWrite 1* » en bleu – sur une nouvelle ligne de ce fichier les informations préalablement mises à jour.

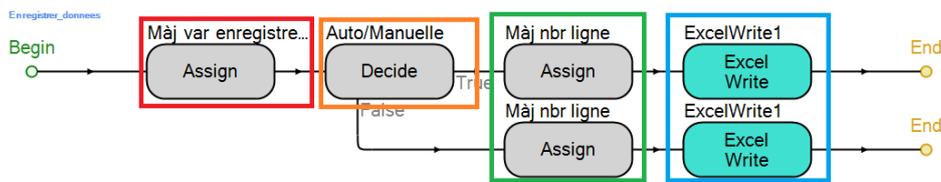


Figure 8.40 : Vue du processus d’enregistrement des données « *Enregistrer_donnees* »

8.3.14 Le réapprovisionnement des stocks

Parmi les nombreuses évolutions qui s’implantent de plus en plus dans les hôpitaux, se trouvent les robots de reconditionnements unitaires permettant de ré ensacher des produits sous un nouveau format plus complet en termes d’informations disponibles. Cette fonctionnalité a donc été intégrée à notre modèle, sous la forme d’une chaîne indépendante du reste de la production. Cette chaîne est représentée par le cadre numéro 3 sur la Figure 8.1, découpée en activités sur la Figure 8.2 (cadre numéro 3.1, 3.2 et 3.3).

Cette chaîne est composée d’une source générant des demandes de réapprovisionnement (numéro 3.1 Figure 8.41), d’un premier serveur modélisant le déconditionnement des produits pharmaceutiques (numéro 3.2 Figure 8.41), d’un second serveur représentant le robot de reconditionnement unitaire (numéro 3.3 Figure 8.41) et enfin d’un puits symbolisant la mise en rayon, dans les stockeurs, etc. des produits pharmaceutiques reconditionnés.

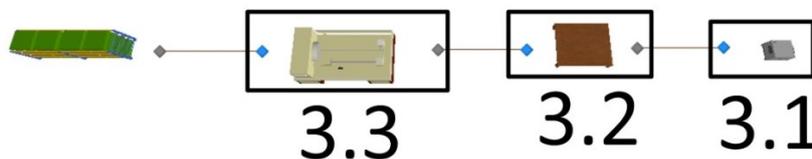


Figure 8.41 : Vue de la chaîne de reconditionnement unitaire des produits

Contrairement au reste du modèle où la gestion des entités de flux est complexe, la linéarité de cette chaîne permet de travailler et de modéliser facilement le fonctionnement de cette dernière. Ainsi, l’élément de flux sera ici intitulé « *Demande_reapro* » et se révélera simple.

8.3.14.1 Leurs propriétés

La grande simplicité des éléments composant cette chaîne permet de ne pas avoir à travailler et modifier en profondeur les éléments de base de SIMIO. L’élément de base « *Demande_reapro* »

se trouve à n'avoir que deux propriétés majeures, qui sont le nombre de produits à réapprovisionner « *Nbr_produit_reapro* » et la quantité à réapprovisionner pour ces produits « *Nombre_elmts_demande_reapro* ». Ces deux propriétés seront de la forme « Réel » et seront obtenues par lecture de tables de données.

Le Tableau 8.13 présente les propriétés ayant été modifiés pour les éléments fixes de cette chaîne.

Dans ce tableau, il est possible de constater que les demandes de réapprovisionnement seront déclenchées quand l'évènement « *Nouv_demande_reapro* » sera lancé. Il est alors généré une demande de réapprovisionnement, et la chaîne de production est alors lancée. La demande passe dans ce cas par un serveur « *Deconditionnement* », de temps de production « *Demande_reapro.Temps_decond* », puis par un serveur « *Reconditionnement* », de temps de production « *Demande_reapro.Nombre_elmts_demande_reapro/Cadence_robot_recon* », le même préparateur exécutant les deux tâches. Les produits reconditionnés sont alors remis en stock.

Tableau 8.13 : Propriétés liées au reconditionnement

Objet	Catégorie	Caractéristique	Réponse
Source « Générateur_reapro »	Entity Arrival Type	Arrival Mode	On event
		Triggering Event	Nouv_demande_reapro
		Triggering Event Count	1
		Entities Per Arrival	1
	Table Reference Assignments	Processing Time	0
Serveur « Deconditionnement »	Process Logic	Processing time	Demande_reapro.Temps_decond
	Secondary Ressource	Object Name	Preparateur
		Selection Goal	Random
Serveur « Robot_recond »	Process Logic	Processing time	Demande_reapro.Nombre_elmts_demande_reapro/Cadence_robot_recon
	Secondary Ressource	Object Name	Preparateur
		Selection Goal	Smallest Distance

8.3.14.2 Les données nécessaires à son exécution

Il se trouve que le nombre de données nécessaires à la simulation de cette chaîne est finalement assez restreint. Le Tableau 8.14 les présente.

Comme déjà dit, les demandes de réapprovisionnement générées par la source ne nécessitent que peu d'informations. De même, le serveur « *Deconditionnement* » n'utilise pas d'information supplémentaire et le serveur « *Reconditionnement* », représentant le robot, n'a besoin que de sa cadence de production en plus.

Enfin, il est utilisé un processus « *Comptage* » qui s'assure de générer l'évènement déclencheur d'un réapprovisionnement – quand un certain nombre de produits a été consommé dans la pharmacie – et la bonne mise en marche de cette chaîne de production.

Tableau 8.14 : Données associées au reconditionnement

Utilisation	Nom	Rôle
Source		
« Générateur_réappro »		
	Nombre_elmts_demande_reappro	Nombre d'éléments par produit dans la demande de réapprovisionnement
	Temps_decond	Temps pour déconditionner les produits de la demande
Serveur		
« Reconditionnement »		
	Cadence_robot_recon	Cadence de reconditionnement du robot (en dose/heure)
Processus « Comptage »		
	Nombre_elmts_consommés	Nombre d'éléments consommés par les demandes préparées
	Seuil_reappro	Seuil au-dessus duquel une demande de réapprovisionnement est émise

8.3.14.3 Les processus

Dans cette chaîne de reconditionnement sont utilisés trois processus. Le premier « *Comptage* » (Figure 8.42) est en réalité présent au niveau du transport. En effet, pour être en mesure de compter la consommation des produits nous devons être au plus proche de la sortie des zones de production afin d'éviter un décalage entre ce qui est compté et ce qui est réellement consommé.

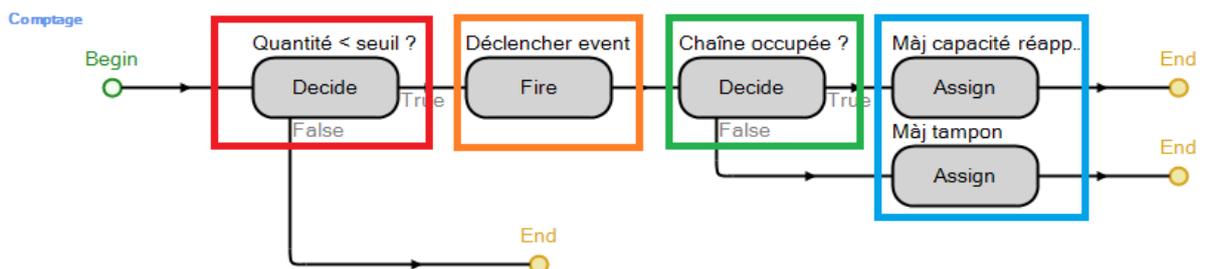


Figure 8.42 : Vue du processus surveillant la consommation de produits

Ce processus vérifie – bloc rouge – à chaque envoi d’une demande que le total des produits consommés ne dépasse pas le seuil défini. Si tel est le cas le bloc orange déclenche un évènement initialisant la création d’une demande de réapprovisionnement. Dans le cas où la chaîne de réapprovisionnement serait déjà occupée, le bloc vert décidera de soit mettre ces demandes en tampon, soit de la mettre en production directement si la chaîne est libre.

Le deuxième et troisième processus ici utilisés sont relativement simple, ces deux derniers permettant de gérer le passage des demandes du serveur « *Deconditionnement* » au serveur « *Reconditionnement* », en mettant à jour leur capacité de production respectives.

Ainsi, dans un premier temps, après avoir déconditionné les produits, le serveur associé voit sa capacité de production mise à zéro (Figure 8.43), afin de transférer cette dernière vers le serveur « *Reconditionnement* ».

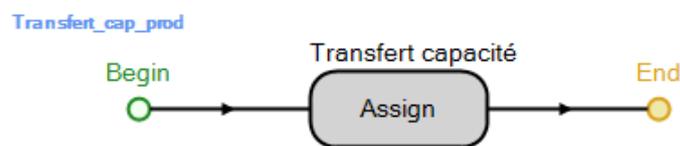


Figure 8.43 : Vue du processus mettant à jour la capacité de production

Une fois le reconditionnement terminé, le second processus rentre en action (Figure 8.44). Si une demande de réapprovisionnement est en attente – test effectué avec le bloc « *Decide* » - alors le déconditionnement des produits pour cette demande est alors lancé : « *Deconditionnement.CurrentCapacity=1* ». Sinon, la chaîne de réapprovisionnement est arrêtée et le préparateur travaillant dessus libéré.

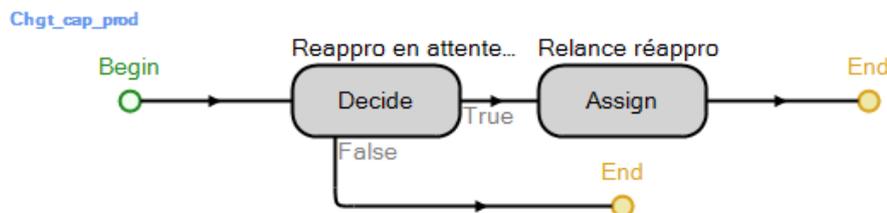


Figure 8.44 : Vue du processus gérant la capacité de production du serveur « *Reconditionnement* »

Nous en avons maintenant terminé avec la conception du modèle de simulation en termes d’objets et de processus. Il reste cependant encore deux points très importants à aborder, mais qui seront néanmoins plus courts en termes de présentation.

8.4 LES TABLES DE DONNEES ET LES ETATS

Deux aspects sont importants sous SIMIO (comme dans la plupart des outils de simulation) et qui sont les données d’entrées et les états manipulés lors de l’évolution du système. En effet, tandis que les premiers permettent de faire démarrer la simulation (et interviennent durant son évolution), les seconds permettent de représenter et d’enregistrer l’évolution de nombreuses variables et informations au fur et à mesure de la simulation.

Nous aborderons donc dans un premier temps les états, puis nous présenterons les différentes tables regroupant les données d'entrées.

8.4.1 Les états

Nous avons déjà abordé la notion d'état dans le chapitre précédent ([partie 7.3.2.3](#)), et même si nous n'y faisons alors référence que comme un attribut des entités, les états peuvent aussi être plus général et être « associé » au modèle dans son entièreté. Par exemple le suivi du nombre de demandes d'un certain type dans le modèle, est un « état général ».

La fenêtre « Définitions » et l'onglet « States » permettent de visualiser l'ensemble des états définis pour le modèle, et qui sont environ une centaine. Ces états couvrent un large panel d'informations provenant des tables des données d'entrées ou bien des informations propres à l'évolution du modèle. Ils peuvent aussi bien concerner le compte des entités dans le modèle, des temps, des cadences, etc. N'étant pas pertinent de présenter l'entièreté de ces états, seule une partie d'entre eux sera détaillée dans le Tableau 8.14 suivant et il s'agira de données non encore présentées.

Tableau 8.15 : Echantillon des états du modèle

Nom de l'état	Type de variable	Rôle
Association_prep_serv	Entier	Egale à « 1 » si une table préparateur/service est définie
Heure_Tranche0	Entier	Définie la fin de la première tranche horaire (ou créneau horaire)
Liste_service_definie	Réel	Egale à « 1 » si une liste de service à préparer est définie
Moyenne_pdt	Réel	Variable enregistrant le nombre moyen de produits dans les demandes créées

8.4.2 Pourquoi de telles tables ?

Afin de faire fonctionner le modèle de simulation, celui-ci nécessite de nombreuses données en entrée pour être dans la capacité de simuler le fonctionnement des objets qu'il contient. Pouvant être associées à des états spécifiques (comme ceux présentés ci-dessus), ou être simplement gardées comme variable des tables, ces données sont elles aussi de nature diverse et variée.

Notre modèle comporte treize tables de données, chaque table traduisant des informations spécifiques à un certain aspect de la simulation (gestion des préparateurs, des robots, liste de service, transport, etc.), pour un contenu de plusieurs centaines de données.

Mais pourquoi un tel volume d'informations est-il nécessaire ?

Les raisons d'une telle quantité sont doubles. Tout d'abord, et comme nous l'avons énoncé à de nombreuses reprises, notre modèle a pour but d'être **générique**, ce qui implique par conséquent divers mode de préparation, divers types de demandes, et tout ce qui en découle.

La seconde raison est que l'un des défis majeurs de cette modélisation a été de contourner les limitations du logiciel SIMIO. En effet, dans le but d'avoir un modèle de simulation entièrement paramétrable et exploitable depuis des interfaces extérieures – SIMIO n'étant alors plus qu'un outil comme un autre – il a été nécessaire d'intégrer l'entièreté des paramètres du modèle dans des tables de données afin de pouvoir les renseigner de l'extérieur. L'objectif de cette démarche est

qu'ainsi un utilisateur lambda, dans notre cas un pharmacien, pourra utiliser le modèle de simulation sans en endommager le contenu et le fonctionnement – ce qui est un impératif pour nous, un outil viable à long terme étant désiré – et sans se préoccuper du comment le modèle fonctionne.

La difficulté majeure d'une telle démarche a alors été la sélection et la formulation des données à intégrer aux tables. En effet, trouver le bon terme, la bonne façon d'intégrer et d'exploiter ces données a conduit à de nombreuses modifications et tests, le nombre de données pouvant être extraites se révélant plus que conséquent si un modèle entièrement paramétrable est l'objectif final.

8.4.3 Les tables

Abordons maintenant les tables en elles-mêmes. Actuellement au nombre de treize, ces tables couvrent les différentes informations nécessaires tout du long de l'évolution du modèle pour simuler la préparation des diverses demandes le parcourant. Chacune des tables présentées dans ce qui va suivre possèdent des informations qui lui sont propres et chacune d'entre elles traitent des aspects différents liés à la préparation.

8.4.3.1 Les données de simulation générales

Nommée « *Donneessimulation* », cette table regroupe les nombreuses informations qui ne sont pas en lien avec un domaine précis et qui n'ont pas trouvé place ailleurs. Cette table contiendra notamment les informations (Tableau 8.15) liées aux tranches horaires, aux nombres de préparateurs, de valideurs, le nombre de demandes, etc.

Tableau 8.16 : Données comprises dans la table « Donneessimulation »

Nom de la donnée	Type de variable	Rôle
PlanningG	Entier	Egale à « 1 » si des services précis sont à préparer chaque jour
Planning_preparateur	Entier	Egale à « 1 » si des plannings de travail préparateur sont définis
Liste_service_definie	Entier	Egale à « 1 » si une liste de service à préparer est définie avec une même probabilité pour chaque service
Proba_service_definie	Entier	Egale à « 1 » si une liste de service à préparer est définie avec des probabilités variables pour chaque service
Association_prep_serv	Entier	Egale à « 1 » si une table préparateur/service est définie
Heuretranche0 à Heuretranche4	Réel	Définit les limites des tranches horaires permettant de régir la probabilité d'arrivée des demandes
Nbr_demande_lundi à Nbr_demande_vendredi	Entier	Définit le nombre de demandes globales/globalisées à préparer chaque jour
Nbrpreparateurmax	Entier	Définit le nombre de préparateurs maximum du système
Nbrvalideurmax	Entier	Définit le nombre de valideurs maximum du système
Dureemoyenne	Réel	Définit la durée moyenne de contrôle d'une demande globale/globalisée
Pour_analyse_niv_01 à Pour_analyse_niv_03	Réel	Définit la probabilité de chaque degré d'analyse d'une prescription

Temps_analyse_niv_01 à Temps_analyse_niv_03	Expression	Définie le temps d'analyse pour valider une prescription suivant le niveau choisi
Debut_journee	Réel	Définie le début de la journée de travail
Fin_journee	Réel	Définie la fin de la journée de travail
Heure_debut_appel	Réel	Définie l'heure de début des appels urgents
Nbr_stockeur	Entier	Définie le nombre de stockeurs mécaniques présents dans la pharmacie
Capacite_stockeur	Entier	Définie le nombre de préparateurs pouvant travailler simultanément sur un stockeur mécanique
T0 à T6	Réel	Définie les limites des créneaux horaires régissant le planning de travail préparateur

8.4.3.2 Le taux d'arrivée des demandes

Cette table, « *Arrivee_demande* » permettra de définir le nombre de demandes/prescriptions (taux d'arrivée) qui arriveront sur chaque tranche horaire. Divisé en 19 tranches « Heure0 » à « Heure18 » de durée variable, la table peut être représentée par la Figure 8.41 suivante.

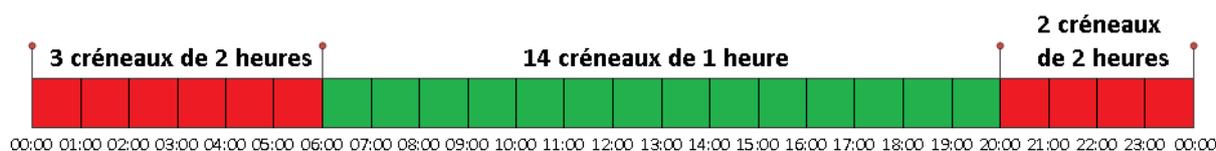


Figure 8.45 : Découpage de chaque journée sous SIMIO

La journée est donc divisée en deux. Une partie représentant le soir et la nuit (20h – 6h) en rouge et découpée par créneaux de deux heures et l'autre représentant la journée (6h – 20h) en vert. Nous avons choisi un tel découpage afin de prendre en compte les horaires de travail usuels du personnel de pharmacie auxquels nous avons ajouté un peu de marge.

8.4.3.3 La préparation automatisée

Caractérisée par la table « *Cycles_robots* », les informations nécessaires à la caractérisation de la préparation automatisée sont de diverses natures comme visible dans le Tableau 8.16.

Tableau 8.17 : Données liées à la préparation automatisée

Nom de la donnée	Type de variable	Rôle
Cycle_defini	Entier	Egale à « 1 » si des cycles de préparations automatisées sont définis
Nom_cycle_01 à Nom_cycle_03	Chaîne de caractères	Définie le nom des cycles
Cycle1_nbr_robot à Cycle3_nbr_robot	Entier	Définie le nombre de robots travaillant sur chaque cycle
Heure_deb_cycle1 à Heure_deb_cycle3	Réel	Définie l'heure de début de chaque cycle

Heure_fin_cycle1 à Heure_fin_cycle3	Réel	Définie l'heure de fin de chaque cycle
Rythme_cycle_1 à Rythme_cycle_3	Entier	Définie la cadence de travail cumulée (en dose/heure) des robots sur chaque cycle

Comme nous l'avons déjà énoncé, notre modèle peut contenir trois types de cycles différents d'où les diverses informations, relatives à ces cycles, présentes dans la table.

8.4.3.4 Le planning de travail préparateur

Géré par la table « *Gestion_preparateurs* », ce planning représente la présence des préparateurs sur chaque journée de la semaine avec les créneaux définis par T0 à T6. Cette table ne donne cependant que le nombre de préparateurs présents sur chaque créneau. Elle peut donc être vue comme une matrice Jour/Créneau d'entiers et de dimension 6*7. Chaque case de la matrice représente le nombre de préparateurs présent entre la fin du créneau précédent et le début du suivant. Ainsi, avec une matrice de la forme suivante (Figure 8.42), la case de position 1*1, précise que zéro préparateurs sont présents entre T6 et T0 le lundi et la case 1*7, que 1 préparateur est présent entre T5 et T6 le lundi.

	T0	T1	T2	T3	T4	T5	T6
Lundi	0	1	6	6	1	6	1
Mardi	0	1	6	6	1	6	1
Mercredi	0	1	6	6	1	6	1
Jeudi	0	1	6	6	1	6	1
Vendredi	0	1	6	6	1	6	1
Samedi	0	1	6	6	1	6	1

Figure 8.46 : Matrice Jour/Créneau représentant la présence des préparateurs sur la semaine

A noter que les données de cette table porte le nom de « *PrepaavtT0* » à « *PrepaavtT6* » afin d'être le plus claire possible quant à l'information demandée, et que par définition la première colonne de cette matrice sera toujours remplie de zéro, considérant ainsi qu'il n'y a jamais de préparateur avant cette heure.

8.4.3.5 La présence des préparateurs

Cette nouvelle table vient en complément de la précédente et apporte en plus la liste des services préparés par chaque préparateur. Nommée « *Table_prepa* », sa première colonne, « *Prepa_dispo_jour* », est composée des numéros des préparateurs présents sur chaque créneau horaire, tandis que la seconde colonne, « *Service_prepa* », liste les services préparés. Cette première colonne est de dimension 1*36 étant donné que les numéros des préparateurs demandés le sont pour les six jours de la semaine et sur six créneaux (T0 – T1, T1 – T2, ..., T5 – T6). La seconde est quant à elle de dimension 1* « *Nbrpreparateurmax* ».

Les informations de cette table devront nécessairement apparaître sous la forme de crochets non séparés par un espace à cause des formules utilisées pour identifier les numéros préparateurs et des services. Ainsi, il sera obtenu quelque chose de la forme « [1][2][3][4] » pour signifier que les préparateurs 1 à 4 sont présents sur ce créneau (avec la même chose pour les services).

8.4.3.6 Les demandes et prescriptions

Nous abordons maintenant la seconde table la plus importante de notre modèle à savoir la table « *Donneesdemande* ». Celle-ci regroupe l'ensemble des informations de base concernant les demandes et prescriptions. En effet, elle contient des informations telles que le nom des demandes, la probabilité de les voir apparaître sur chaque tranche horaire (« *Heuretranche0* » à « *Heuretranche4* »), le nombre de produits qu'elles contiennent, etc. Le Tableau 8.17 récapitule cela.

Tableau 8.18 : Données caractérisant les demandes parcourant notre modèle

Nom de la donnée	Type de variable	Rôle
Typedemande	Chaîne de caractères	Définie le nom de la demande
Indicedemande	Entier	Définie l'indice de la demande
ProbaavtH0 à ProbaavtH4	Entier	Définie la probabilité de trouver cette demande sur chaque tranche horaire
ProbaapresH4	Entier	Définie la probabilité de trouver cette demande après la dernière tranche horaire
Possible_hybride	Entier	Egale à « 1 » si la demande peut être hybride
Validation	Entier	Egale à « 1 » si la prescription est validée
Modetraitemet	Entier	Définie le mode de traitement de la demande (de 1 à 4)
Nbrproduit	Expression	Définie le nombre de produits de la demande
Nbrelmts	Expression	Définie le nombre d'éléments de la demande
Tempsprepa	Expression	Définie le temps de préparation de la demande

Des précisions sont à apporter sur les informations composant cette table. La Figure 8.43 donne un exemple de remplissage du début de la table où six types de demandes/prescriptions sont définis avec leurs probabilités sur chaque tranche horaire. Pour que cette partie de la table soit valide il faut que la probabilité cumulée d'apparition des demandes/prescriptions sur chaque créneau soit de 100% (86% plus 14% dans cet exemple).

	Typedemande	Indicedemande	Probaavt H0	Probaavt H1	Probaavt H2	Probaavt H3	Probaavt H4	Probaapres H4
1	InforInd	1	0	0	0	0	0	0
2	PapierInd	2	0	0	0	0	0	0
3	Globalrobot	3	86	86	86	86	86	86
4	Globale	4	0	0	0	0	0	0
5	Globalisee	5	14	14	14	14	14	14
6	Armoiresecu	6	0	0	0	0	0	0

Figure 8.47 : Exemple de remplissage du pourcentage d'apparition des demandes sur les créneaux

8.4.3.7 Les sous-demandes et sous-prescriptions

Nous avons déjà vu le cas où des demandes et prescriptions peuvent être composées de divers éléments, ce qui se traduit par la donnée « *Possible_hybride* » égale à un. Dans un tel cas, des informations seront donc demandées en complément pour traduire le contenu des sous-demandes et sous-prescriptions pouvant être générées.

L'ensemble des données nécessaires de la table « *Donnees_demandes_hybrides* » sont représentées dans le Tableau 8.18.

Tableau 8.19 : Données relatives à la création des sous-demands/sous-prescriptions

Nom de la donnée	Type de variable	Rôle
Pourc_stockeur	Réel	Définie le pourcentage de demande/prescription ayant des produits en stockeur
Nbr_pdts_stockeur	Expression	Définie le nombre de produits en stockeur
Nbr_elmts_stockeur	Expression	Définie le nombre d'éléments des produits en stockeur
Temps_prepa_stockeur	Expression	Définie le temps de préparation des produits en stockeur
Pourc_solute	Réel	Définie le pourcentage de demande/prescription ayant des solutés
Nbr_solute	Expression	Définie le nombre de solutés
Nbr_elmts_solute	Expression	Définie le nombre d'éléments par soluté
Temps_prepa_solute	Expression	Définie le temps de préparation des solutés
Pourc_frigo	Réel	Définie le pourcentage de demande/prescription ayant des produits frigorifiques
Nbr_frigo	Expression	Définie le nombre de produits frigorifiques
Nbr_elmts_frigo	Expression	Définie le nombre d'éléments des produits frigorifiques
Temps_prepa_frigo	Expression	Définie le temps de préparation des produits frigorifiques
Pourc_nominative	Réel	Définie le pourcentage de demande possédant des prescriptions liées
Validation_nominative	Expression	Egale à « 1 » si ces prescriptions doivent être validées
Nbr_nominative	Expression	Définie le nombre de prescriptions liées
Nbr_prod_nominative	Expression	Définie le nombre de produits de ces prescriptions
Nbr_elmts_nominative	Expression	Définie le nombre d'éléments des produits de ces prescriptions
Mode_prepa_nominative	Entier	Définie le mode de préparation des prescriptions
Temps_prepa_nominative	Expression	Définie le temps de préparation des prescriptions

Il apparaît donc dans cette table toutes les informations nécessaires pour créer des sous-demands, des sous-prescriptions de solutés, de produits en stockeur, de produits frigorifiques, etc. Cette table est certainement celle qui demandera les informations les plus complexes et les plus longues à obtenir, notamment en ce qui concerne la composition des demandes ainsi que les différentes valeurs pour les nombres de produits, d'éléments, etc.

8.4.3.8 Les plannings des demandes de service

Cette nouvelle table, « *Planning_demande_G* », fait le lien avec les différentes valeurs demandées dans la première table et qui se nomment « *Nbr_demande_lundi* », etc. Ainsi, en concordance avec ces valeurs, il apparaîtra en colonne – une par jour – les différents numéros des services, un seul par ligne et une nouvelle fois entre crochet.

8.4.3.9 La liste des services pouvant être préparés

Contrairement à ce qui pourrait être pensé, cette table n'est pas redondante avec la précédente. Cette fois-ci, les services listés ici peuvent être préparés mais ne le seront pas

impérativement comme précédemment, étant donné qu'ils sont sélectionnés par le biais de probabilités. Ces probabilités peuvent être la même pour toute – colonne « *Proba_uniforme* » – ou variées, chaque service ayant la sienne « *Proba_service* ». Par conséquent, la table obtenue est composée de trois colonnes « *Proba_service* », « *Numero_service* » et « *Proba_uniforme* », chaque ligne représentant un service avec les deux probabilités liées.

8.4.3.10 Les valideurs

Il peut exister dans notre modèle jusqu'à six types de valideurs différents ayant chacun des créneaux de validation définis par « *Horaire_debut* » et « *Horaire_fin* ». De plus, il existe un nombre précis, « *Nombre_valideurs* », pour chaque type. La table « *Gestion_valideurs* » a ici la forme d'une matrice de dimension 6*3, le numéro de chaque ligne faisant référence au type de valideur (la ligne une correspond donc au valideur de type 1).

8.4.3.11 Les retours de produits

Cette dernière table, « *Donnees_retours* », concerne la gestion des retours de produits des services. Cette table incorpore donc les diverses informations relatives à ces retours, et qui prennent une forme similaire aux informations de la table « *Donneesdemande* » (Tableau 8.19).

Tableau 8.20 : Données concernant le retour produits des services

Nom de la donnée	Type de variable	Rôle
Proba_retour_stock	Réel	Définie la probabilité que les produits retournés doivent être réintégrés au stock manuel
Proba_retour_stockeur	Réel	Définie la probabilité que les produits retournés doivent être réintégrés au stockeur
Nombre_produit_retour	Expression	Définie le nombre de produits retournés
Nombre_elmt_retour	Expression	Définie le nombre d'éléments des produits retournés
Temps_traitement	Expression	Définie le temps de rangement des produits dans les divers emplacements de stock

8.4.3.12 Le reconditionnement des produits

Avant dernière table, celle-ci comportera les données liées au reconditionnement des produits. Contrairement aux autres types de demandes, les demandes de réapprovisionnement ne comporteront que peu d'informations et n'en solliciteront pas un grand nombre pour leur préparation.

Tableau 8.21 : Données concernant le reconditionnement de produits

Nom de la donnée	Type de variable	Rôle
Nombre_produit_reapro	Expression	Définie le nombre de produits présent dans les demandes de réapprovisionnement
Nombre_elmts_demande_reapro	Expression	Définie le nombre d'éléments par produit
Cadence_robot_recon	Entier	Définie la cadence en dose/heure du robot de reconditionnement
Temps_decondi	Expression	Définie le temps mis pour déconditionner les produits

8.4.3.13 Le transport des demandes

Cette dernière table contiendra les données relatives aux transports des demandes préparées jusqu'aux servies. Chaque service étant plus ou moins éloignés et accessibles, le temps mis pour transporter les demandes jusqu'à eux sera donc variable.

Tableau 8.22 : Données concernant le transport des demandes

Nom de la donnée	Type de variable	Rôle
Nbr_agent_logistique	Entier	Définie le nombre d'agents logistiques transportant les demandes
Tps_transport_defaut	Expression	Définie le temps de transport par défaut
Service	Chaîne de caractères	Définie le numéro du service (entre crochet) desservi
Temps_transport_spe	Expression	Définie le temps de transport associé au service

La Figure 8.48 est un exemple de la table liée au transport. Quelques précisions sont à apporter concernant son remplissage. La première ligne de la colonne « *Nbr_agent_logistique* » représente le nombre d'agents disponibles, ici deux, tandis que la seconde ligne de cette même colonne représente leur capacité de transport. Ainsi le service logistique peut dans cet exemple transporter 10 demandes en même temps.

Dans la seconde colonne, seule la première ligne est utilisée, le temps de transport par défaut étant le même pour tous les services, ici une minute.

	Nbr_agent_logistique	Tps_transport_defaut	Service	Temps_transport_spe
1	2	1	[1000]	1
2	5	0.0	[1001]	2
3	0	0.0	[1002]	3
4	0	0.0	[1003]	4
5	0	0.0	[1004]	5
6	0	0.0	[1005]	6

Figure 8.48 : Exemple de remplissage de la « Table_transport »

La troisième colonne liste les services pouvant être livrés, service dont le nom est écrit une fois de plus entre crochet. Cette colonne est ainsi en lien direct avec la dernière « *Temps_transport_spe* », le temps précisé ici sur chaque ligne étant à lier avec le service de la même ligne. Par exemple, livrer le service « 1003 » demande quatre minutes.

8.5 INTERFACE DE SAISIE DES DONNEES

Afin de faciliter l'utilisation des tables pour les utilisateurs, nous avons conçu une interface de saisie leur permettant de renseigner les nombreuses informations nécessaires au fonctionnement du modèle de simulation.

Développée sous EXCEL en VBA, cette interface stocke dans des feuilles EXCEL les nombreuses informations saisies, informations qui sont récupérées par le logiciel SIMIO à chaque lancement d'une simulation. Pour cela, il faut lier chaque table SIMIO à une feuille EXCEL, en utilisant l'option « Bind to » (Figure 8.49) et en sélectionnant « EXCEL ».

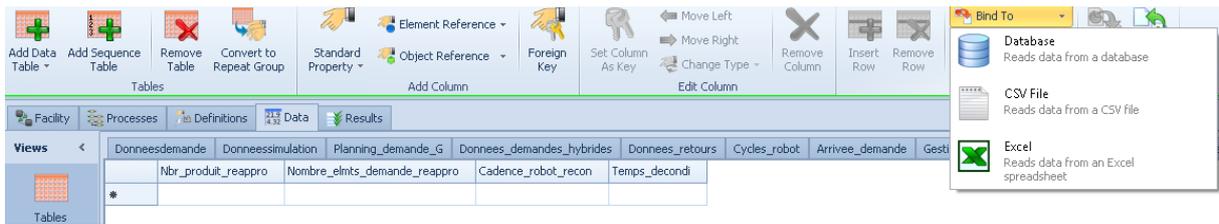


Figure 8.49 : Interface « Data » avec l’option « Bind to » sélectionnée

Une fois l’option sélectionnée, il apparaît alors une fenêtre pour aller sélectionner le fichier EXCEL et la feuille, ou les cellules voulues (Figure 8.50).

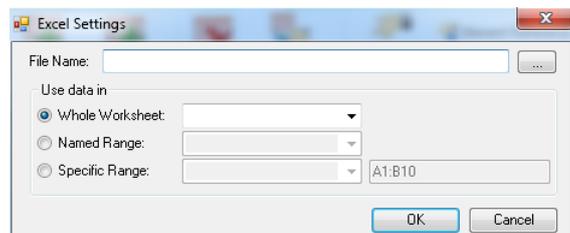


Figure 8.50 : Vue de la fenêtre de sélection du fichier EXCEL

Nous ne présenterons pas toute l’interface ici, cela étant assez conséquent, mais seulement les fenêtres les « plus » importantes. Il est à noter que les différentes vues de l’interface sont encore amenées à évoluer d’un point de vue graphique.

Commençons par la page « d’accueil » (Figure 8.51). Sur cette page sont saisies les informations telles que le début et la fin de la journée de travail, les créneaux découpant les journées, et aussi le nombre de demandes réparties sur la journée.

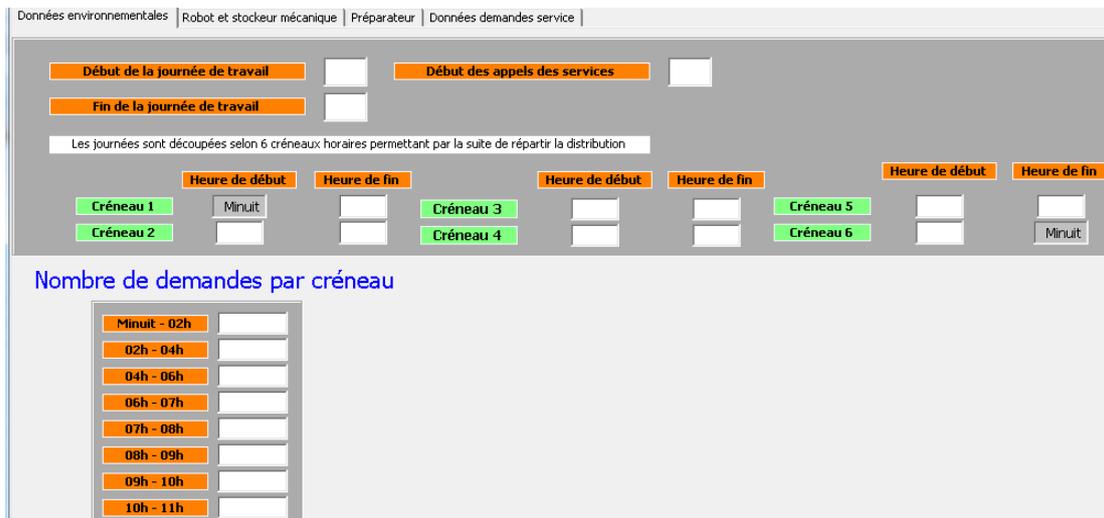


Figure 8.51 : Page d’accueil de l’interface

Le second écran (Figure 8.52) présente la page où les informations relatives aux demandes sont renseignées. Il est possible de trouver dessus le nom de la demande, sa probabilité d’apparition sur les créneaux horaires, si elle doit être validée, son nombre de produits, etc.

Figure 8.52 : Vue de la page de saisie des informations sur les demandes

Cet écran est lié avec le suivant (Figure 8.53) dans le cas où les demandes saisies sont hybrides ou possèdent des sous-prescriptions. Dans cette fenêtre il est possible de préciser la nature des sous-demandes, le pourcentage de demandes impliquées, le mode de préparation pour les sous-prescriptions, etc.

Figure 8.53 : Ecran de saisie des informations sur les sous-demandes et sous-prescriptions

Le dernier écran présenté est celui de la Figure 8.54 et qui concerne la présence des préparateurs. Il est possible de définir sur cette interface le nombre de préparateurs présents sur chaque créneau horaire.

Figure 8.54 : Vue du planning de présence préparateur

8.6 CONCLUSION

Ayant pour but de présenter la construction du modèle de simulation, ce chapitre nous a donc permis de faire un tour d'horizon des nombreux éléments et items qui composent un modèle sous le logiciel SIMIO. Dans un premier temps, la compréhension globale du modèle, les différents parcours possibles pour les entités de flux ainsi que les éléments majeurs composant le modèle ont été présentés.

Par la suite, nous avons poursuivi le détail du modèle en abordant les éléments « 3D » le composant, pour lesquels de nombreuses informations ont été fournies : propriétés, processus liés, données nécessaires. Pour commencer, nous avons observé les éléments « mobiles », à savoir les travailleurs et les demandes, où nous constatons que la complexité des demandes est importante (nombreuses propriétés et règles de gestion). Le cas particulier des demandes urgentes est alors abordé, celles-ci possédant une importance particulière dans les structures hospitalières. Comparativement aux demandes d'ordre plus général, elles se révèlent plus simples en termes de composition, mais possèdent un indice de priorité plus élevée, afin qu'elles soient préparées en premier dès leur arrivée.

Nous avons poursuivi le détail du modèle en nous intéressant ensuite aux prescriptions nominatives. Pour celles-ci, nous avons présenté comment elles sont séparées des demandes, comment elles sont décomposées, comment leur validation est réalisée, comment elles sont dirigées et orientées dans le modèle et enfin comment se passe la préparation robotisée.

Nous avons fait de même pour les autres types de demandes en détaillant comment fonctionnaient le contrôle et l'édition, l'éclatement de ces demandes sur les différents postes de préparation, avant de poursuivre avec l'analyse du regroupement et de l'envoi aux services des demandes et prescriptions préparées.

S'en suit une partie qui concerne la présentation du réapprovisionnement des stocks, en observant comment ces demandes sont générées, comment elles sont préparées et les besoins qu'elles sollicitent.

Dans l'avant-dernière sous-section, nous détaillons les nombreuses tables de données utilisées dans notre modèle, en présentant les variables composant ces tables et leurs rôles, puis dans la dernière nous présentons en partie l'interface permettant de remplir les tables précédemment détaillées.

Ces tables se révèlent donc être centrales et cela pour deux raisons. Comme nous l'avons énoncé plus haut, celles-ci permettent de travailler et d'exploiter le modèle de simulation sans avoir à manipuler ce dernier. C'est un énorme avantage pour les utilisateurs, les pharmaciens, qui ainsi n'ont pas à appréhender d'éventuelles erreurs de manipulation et n'ont surtout pas besoin de comprendre comment le modèle fonctionne, seuls les résultats les intéressent. Le second avantage est que, ce modèle étant générique, la simulation d'une structure hospitalière particulière ne se fera pas par des changements sur le modèle de simulation lui-même, mais uniquement en modifiant les données des treize tables que ce modèle exploite. **Il est ainsi possible de simuler rapidement de nombreux choix organisationnels en ne faisant varier que quelques données d'entrées.**

Il est maintenant possible d'exploiter notre démarche. Cela sera l'objet du prochain chapitre, qui va présenter et utiliser trois scénarios de simulation différents.

IX

ÉVALUATION DES OUTILS ET APPLICATION DE NOTRE DÉMARCHE

RESUME DU CHAPITRE 9

A ce stade, les pharmaciens possèdent tous les outils et informations nécessaires pour exploiter pleinement notre démarche. Par conséquent, il est temps d'utiliser nos travaux et de les mettre en application sur des cas d'études précis afin de fournir les informations pertinentes pour la prise de décisions concernant ces problèmes.

Dans cette optique, ce chapitre va mettre en application notre démarche à travers trois scénarios portant sur des problématiques importantes pour la PUI du CHU de Clermont-Ferrand. Ces problématiques concerneront la mise en place de la dispensation journalière individuelle nominative – automatisée ou manuelle – et la validation des prescriptions dans le cadre de cette dispensation.

Ainsi, pour chaque scénario, nous raisonnerons d'un point de vue pharmacien et présenterons comment le diagramme générique des processus, le diagramme générique des classes et le modèle générique de simulation s'utilisent, avant de détailler et d'analyser les résultats obtenus pour chacun de ces scénarios.

La construction de ces scénarios sera précédée par l'évaluation et un bilan portant sur la genericité et la réutilisabilité des trois outils que nous avons développés tout au long de ces travaux.

SOMMAIRE DU CHAPITRE

9.1	INTRODUCTION	287
9.2	GENERICITE ET REUTILISABILITE DES OUTILS	287
9.2.1	La généricité	287
9.2.2	La réutilisabilité	292
9.3	CADRE GENERAL.....	294
9.4	SCENARIO 1 : IMPACT DE LA DJINM.....	296
9.4.1	Contexte	296
9.4.2	Utilisation des outils et de la démarche.....	296
9.4.3	Résultats de la simulation	300
9.4.4	Conclusion	302
9.5	SCENARIO 2 : IMPACT DE LA DJINA.....	302
9.5.1	Contexte	302
9.5.2	Utilisation des outils et de la démarche.....	303
9.5.3	Résultats de la simulation	305
9.5.4	Conclusion	309
9.6	SCENARIO 3 : COMMENT REPARTIR SA VALIDATION ?.....	309
9.6.1	Contexte	309
9.6.2	Utilisation des outils et de la démarche.....	310
9.6.3	Résultats de la simulation	313
9.6.4	Conclusion	318
9.7	CONCLUSION	318

9.1 INTRODUCTION

Nous possédons désormais toutes les compétences et les outils nécessaires pour tester de nouvelles configurations ou des fonctionnements particuliers d'une pharmacie. L'objet de ce chapitre est d'exploiter notre démarche et les outils qui lui sont liés afin de visualiser leur mise en œuvre et leur utilisation. Pour cela, nous supposerons dans chaque cas, être un pharmacien qui souhaite résoudre et étudier divers problèmes liés à sa pharmacie et qui sollicitera donc notre démarche.

Ainsi, nous allons dans ce chapitre construire trois scénarios permettant de manipuler nos outils et de paramétrer le modèle de simulation, pour lesquels nous détaillerons et analyserons les résultats obtenus.

Les deux premiers scénarios porteront sur la problématique de la DJIN manuelle ou automatisée, et permettront de visualiser l'impact d'une telle démarche sur la pharmacie du CHU de Clermont-Ferrand.

Le troisième sera quant à lui aussi en lien avec cette notion de nominatif. En effet, nous y traiterons les questions liées à la validation des prescriptions, tâche pharmaceutique qui se révèle être une problématique à elle seule.

A noter que durant ces scénarios, nous ne tiendrons pas compte du type de lits ou d'unité fonctionnelle (portion d'un service et **notée UF**) dont est issue la prescription ou la demande.

La mise en place des scénarios et l'exploitation de nos outils seront précédées par une première partie portant sur l'évaluation de la généricité et de la reproductivité des dits outils. A de nombreuses reprises nous avons abordé et parlé de cette notion, et il est temps d'évaluer réellement si notre travail peut se targuer d'être générique. Pour cela, nous réutiliserons une nouvelle fois les travaux de [\[FLETCHER 09a\]](#) et [\[ROBINSON 04\]](#), et nous passerons en revue les nombreux critères énoncés afin d'infirmes ou non cette idée.

9.2 GENERICITE ET REUTILISABILITE DES OUTILS

Lors des chapitres précédents nous avons fréquemment affirmé le caractère générique de nos divers outils. Ainsi, et afin de prouver nos dires, nous allons utiliser les travaux de [\[FLETCHER 09a\]](#) pour évaluer la généricité de notre modèle des processus, de notre diagramme des classes et enfin de notre modèle de simulation.

L'évaluation portera majoritairement sur ce dernier, mais étant donné que les deux autres outils ont été nécessaires à la création du troisième, la généricité de l'ensemble sera démontrée.

9.2.1 La généricité

La littérature a maintes fois montré que la comparaison des modèles génériques et spécifiques était complexe et se faisait sur de multiples dimensions (contenu, données, etc.). [\[FLETCHER 09a\]](#) met en évidence à travers ses travaux, ces nombreuses dimensions et la complexité de découpler modèle spécifique et modèle générique, notamment dans le cas de modèle générique dédié à un domaine.

En reprenant les Figure 2.21 et 2.22, notre but étant d'obtenir un modèle générique dédié aux pharmacies, il est alors possible d'affirmer que celui-ci devrait être de « Niveau 3 » (Figure 2.21). En effet, les critères pour qu'un modèle appartienne à ce niveau sont qu'il doit être dédié à un service, dans un secteur défini, et pour de multiples centres de soins. Notre modèle étant dédié à la pharmacie – même s'il aborde d'autres services tels que la logistique et l'unité de soins – dans le secteur hospitalier et supposément applicable à de nombreux centres de soins, répond donc à ces critères.

Afin d'étayer cette constatation, nous pouvons nous appuyer sur le non-respect d'un critère dans les « Niveau 2 » et « Niveau 4 ». Le « Niveau 4 » devant être exclusivement dédié à un unique centre de soins, celui-ci ne correspond pas à notre modèle car il est applicable au moins à deux structures. De même, le « Niveau 2 » doit traiter de manière approfondie plusieurs services, ce qui en l'occurrence n'est pas notre cas, le service logistique et l'unité de soins n'étant que simplement modélisés.

Par la suite, le « Niveau 3 » vient à être scindé en deux afin d'obtenir deux sous-niveaux devant assurer une meilleure appréhension de la généricité d'un modèle. Il nous est cependant difficile de déduire, à première vue, quel sera le sous-niveau à choisir. Pour cela, nous réutiliserons les différents tableaux de critères présentés dans [FLETCHER 09a], ainsi que leur application dans [FLETCHER 09b].

9.2.1.1 Les facteurs clés de l'initiation du projet

Dans le cadre de modèle devant être générique, l'objectif et le niveau d'utilisation de ce modèle doivent être parfaitement définis avant de poursuivre la modélisation. Ainsi en se référant aux différents critères énoncés par [FLETCHER 09a], pour cette première partie, nous obtenons le Tableau 9.1.

Tableau 9.1 : Réponses aux facteurs clés de [FLETCHER 09a] pour l'initiation d'un projet

Critère	Réponse
Cadre	Modéliser le flux de médicaments traversant une pharmacie hospitalière et cela de manière locale ou plus générique
Objectif(s)	Numéros 1 à 5 (voir Figure A.4 en Annexe 3)
Niveau de perception	Des problèmes organisationnels et structurels d'ordre générique à des problèmes strictement locaux (taille d'équipe, choix matériel)
Niveau de précision	Le modèle devra être flexible et ainsi permettre de modéliser aussi bien des tâches locales que le fonctionnement dans sa globalité
Conflit avec des modèles existants	Aucun modèle équivalent connu à un niveau local ou plus général
Niveau d'utilisation visé	Doit s'adapter à des applications locales (pour simuler certaines structures en particulier) ainsi qu'à des structures fictives représentatives de divers établissements.

En s'appuyant sur ces critères, fixés en collaboration avec les parties prenantes, notre modèle pencherait plus dans ces circonstances vers le niveau 3B et cela pour deux raisons. Tout d'abord, notre modèle a pour objectif d'être applicable localement et de manière plus globale, chose que le niveau 3A ne permet pas forcément du fait de sa définition. De surcroît, tandis que le niveau

3A se voit associer un modèle dont le niveau de précision est assez faible, le niveau 3B possède quant à lui un niveau de précision bien plus élevé, ce qui correspondrait mieux à nos attentes.

Passons au deuxième lot de facteurs clés, à savoir ceux portant sur le design et la conception du modèle.

9.2.1.2 Design et conception du modèle

Les facteurs clés ici sollicités concernent les divers aspects qui toucheront au design et à la conception du modèle, des processus, etc. Le Tableau 9.2 les présente.

Tableau 9.2 : Réponses aux facteurs clés de [FLETCHER 09a] pour le design et la conception

Critère	Réponse
Niveau de réutilisabilité du modèle	Le développement des différents outils a été réalisé dans une optique de genericité la plus poussée possible, afin d’être réutilisé dans de nombreuses situations. Les divers éléments du modèle ont aussi été pensés de manière à être les plus génériques et modulable possibles
Design des processus	Réalisé conjointement avec des pharmaciens et internes, dont les connaissances se situent au niveau local, les informations nécessaires à un développement plus générales ont été récupérées par le biais de la littérature ou de contact dans diverses structures
Représentation de problèmes locaux	Le changement des données d’entrées permet de faire varier la structure simulée et donc d’étudier des problèmes à dimension locale
Justesse d’utilisation	Les deux premiers outils permettent d’approfondir les connaissances et les possibilités des utilisateurs, tandis que le modèle donnera la possibilité de simuler des configurations inédites. Cela demandera cependant de bien définir en amont le cadre de simulation
Niveau de détails	Les processus modélisés représentent les modes de dispensation et de distribution utilisés et reconnus par tous les hôpitaux de France
Objectif du design	Les processus modélisés et simulés peuvent s’adapter à une représentation locale si cela est souhaité, et si les données fournies sont adéquates
Connaissance de l’utilisateur sur la structure et le fonctionnement du modèle	Quelques utilisateurs ont été impliqués dans le développement du modèle des processus, mais dans les deux autres outils, leur niveau de connaissance sera faible

En regardant les critères liés aux « connaissances du modèle par les utilisateurs » et à la « représentation de problèmes locaux », tout en les comparant aux critères du Tableau 2.8, le bilan qui peut en être tiré est qu’une fois de plus, le modèle et les outils se rapprochent plus du niveau 3B que 3A. En effet, la connaissance du fonctionnement des outils, potentiellement basse au début, concorde avec le critère du niveau 3B. De même, la représentation des problèmes locaux passera par un paramétrage de nos données d’entrées, chaque paramétrage représentant un problème différent, ce qui se trouve être le critère du niveau 3B. Les autres facteurs clés de ce tableau suivent d’ailleurs la même ligne.

9.2.1.3 Les données

Les données représentent pour les modèles de simulations la base de leur fonctionnement, et trois facteurs clés de **[FLETCHER 09a]** permettent de les étudier (Tableau 9.3).

Tableau 9.3 : Réponses aux facteurs clés de [FLETCHER 09a] pour les données

Critère	Réponse
Qualité des données	Suivant le type de problème, les données pourront être de nature locale ou nationale. Bien que les premières peuvent être plus soumises aux erreurs, les deux se révèlent de qualité si les extractions et un travail minutieux d'interprétation est mené dessus
Accessibilité des données	Les données nationales seront aisément accessibles (rapport, site web), tandis que les données locales devront être récupérées par le biais des logiciels de gestion ou de suivi des hôpitaux. Ces dernières ne sont pas toujours facilement accessibles ce qui posera problème dans le cas d'étude locale
Données test de départ	Nos données de départ pour les tests du modèle de simulation sont issues d'extraction locale (répartition des demandes par exemple) et de données nationales (temps d'analyse des prescriptions, etc.).

Difficile de tirer des conclusions de ce tableau étant donné que les critères des facteurs clés sont identiques pour les deux niveaux. Néanmoins certaines déductions peuvent être émises sur les données. Tout d'abord, un modèle tourné vers du générique national, requerra des données nationales. Ce type de données peut être trouvé dans les nombreux rapports ministériels (DGOS, DHOS, IGAS, ANSM, etc.) ainsi que sur les sites web d'organisations tels que l'OCDE, OMS, etc. Un modèle tourné vers le local demandera quant à lui des données issues de la structure simulée. La difficulté majeure réside alors au niveau de l'obtention de ces données locales, pas toujours correctement stockées, ou bien sur lesquelles aucun travail de prétraitement n'a été réalisé, n'étant dès lors qu'un amas de données brutes. Ainsi dans le cas d'un manque local pour obtenir une information (dont la validité est vérifiable), il sera possible de suppléer cette donnée locale par son équivalent national, tout en soulignant le risque de biais encouru.

Au final, pour les facteurs clés de ce domaine, aucun niveau ne peut être privilégié, bien que l'utilisation de données locales fasse pencher la balance du côté du niveau 3B.

9.2.1.4 La validation

Difficulté majeure pour l'obtention d'un modèle éprouvé, les facteurs clés liés à la validation sont au nombre de trois pour **[FLETCHER 09a]**.

Tableau 9.4 : Réponses aux facteurs clés de [FLETCHER 09a] pour la validation

Critère	Réponse
Technique de validation	En se référant au terme utilisé par [FLETCHER 09a] , la validation a eu lieu au niveau local en « open box » et en « black box »
Représentation	CHU de Clermont-Ferrand uniquement
Précision requise	Elevé pour pouvoir en tirer des conclusions au niveau local

Ici, le cadre de validation du modèle de simulation et des deux autres outils varie, ces deux derniers ayant été plus faciles à valider sur des structures autres que le CHU de Clermont-Ferrand, seule structure sur laquelle de véritables tests ont réellement pu être menés. La configuration de ces critères vis-à-vis de ceux de la Figure A.7 fait donc de nouveau pencher le modèle vers un niveau 3B, bien qu’une nouvelle fois les deux autres outils se montrent moins facilement classifiables.

9.2.1.5 L’implémentation

Ce domaine concernera une nouvelle fois majoritairement le modèle de simulation, et sera d’importance, le manque d’implémentation des solutions proposées étant une critique récurrente sur la simulation du domaine hospitalier [LENT 12]. Le Tableau 9.5 liste les différents facteurs clés et leurs réponses.

Tableau 9.5 : Réponses aux facteurs clés de [FLETCHER 09a] pour l’implémentation

Critère	Réponse
Capacité d’utilisation des outils	Devant être utilisé par les pharmaciens, le développement des différents outils a été pensé dans une optique de simplicité d’utilisation. Les deux premiers outils sont assez triviaux, ce qui n’est pas forcément le cas du modèle de simulation qui requiert une certaine prise en main selon le niveau de simulation souhaité
Le modèle établi est opérationnel	Testé localement, les résultats des divers tests se révèlent proches de la réalité. Des tests plus génériques ont été menés afin d’évaluer le bon fonctionnement du modèle mais ces derniers mériteraient d’être approfondis
Post développement du modèle	Les outils sont d’ores et déjà opérationnels et ne devraient subir aucun post-développement
Volonté d’utilisation du modèle	Un outil d’aide à la décision a été requis, ce qui sera fourni et composé de nos trois outils. Cependant le degré de volonté d’utilisation de ces outils dépendra de la mise en valeur que nous pourrons appliquer à chacun d’entre eux afin d’en montrer les avantages
Logiciels et composants requis	Les deux premiers outils ne nécessitent pas de logiciels dédiés si aucun post-développement n’est réalisé, ce qui n’est pas le cas du modèle de simulation qui nécessite le logiciel SIMIO

Comme pour les domaines précédents, les différentes réponses aux critères n’orientent pas clairement le modèle vers un niveau en particulier. En effet, ce modèle a été commissionné, ce qui d’après la Figure A.8 (Annexe 3), le dirige vers un modèle de niveau 3A, cependant la réponse au dernier critère le dirige de nouveau vers un 3B (en se référant une nouvelle fois à la Figure A.8).

9.2.1.6 Quel niveau au final ?

En se référant aux réponses apportées pour les nombreux facteurs clés des différents domaines, il est difficile d’affecter un niveau en particulier à nos outils et notre modèle étant donné que ceux-ci « naviguent » entre deux eaux. Tandis que le modèle générique des processus, et le diagramme générique des classes pencheraient plus du côté d’un niveau 3A, du fait de leur plus grande généricité, le modèle de simulation s’oriente plus vers un niveau 3B.

Cependant, le niveau du modèle ne peut être affirmé avec exactitude, celui-ci ayant été construit d’une manière locale mais avec une réflexion globale, le laissant donc osciller entre 3A ou 3B selon l’utilisation qui en sera faite.

A noter qu'il reste un travail important à mener pour la validation du modèle de simulation, notamment à une échelle plus importante.

9.2.2 La réutilisabilité

Nous avons abordé dans ce qui précède la notion de généricité, mais plus d'un point de vue contenu et fonctions que d'un point de vue structure (en termes de code, de programmation, etc.). C'est donc à ce niveau que la réutilisabilité prend de l'importance, celle-ci s'axant sur la capacité d'un modèle à pouvoir être exploité partiellement, ou dans sa globalité, par de nouveaux utilisateurs et sur de nouveaux systèmes. Nous avons vu dans le Chapitre 2, que dans [ROBINSON 04], plusieurs auteurs tentent d'apporter des pistes et des classifications concernant cette problématique. Nous allons par conséquent les réutiliser pour étudier notre modèle de simulation.

9.2.2.1 La réutilisabilité de Robinson

Pour ce premier, comme il est possible de le voir sur la Figure 2.23, la réutilisabilité d'un modèle se décompose en quatre niveaux. En regardant chacun d'entre eux, nous pouvons déjà tirer certains enseignements. Notre modèle a été conçu par « ponction partielle de code », les nombreux exemples fournis par le logiciel SIMIO nous permettant de « coder », concevoir certaines parties ou certains processus de notre modèle.

Notre modèle a cependant été pensé dès le début dans une optique générique, classant son niveau de réutilisabilité parmi les trois derniers niveaux. Le choix du niveau sera déterminé par ce que le nouvel utilisateur souhaite réaliser et par le système devant être simulé. Techniquement, notre modèle est capable de simuler le fonctionnement de toute pharmacie hospitalière, ce qui impliquerait un niveau « Réutilisabilité d'un modèle complet ». Cependant, cela ne peut être vrai que si l'utilisateur souhaite une simulation d'ordre générique. Si ce dernier souhaite un modèle parfaitement dédié, alors notre modèle correspondra plus au niveau « Réutilisabilité de composants » voir « Réutilisabilité de fonctions ». Le Tableau 9.6 tente de synthétiser la variation du niveau de réutilisabilité de notre modèle suivant ces divers critères.

Tableau 9.6 : Variation du niveau de réutilisabilité suivant l'objectif poursuivi pour Robinson

Système étudié	Niveau de réutilisabilité Objectif de l'utilisateur	Niveau 1	Niveau 2	Niveau 3	Niveau 4
Pharmacie hospitalière	Obtention d'un modèle dédié	X	X	X	
	Obtention d'un modèle générique			X	X
Autre que pharmacie	Simulation d'une supply-chain identique dans le déroulement	X	X	X	
	Simulation d'une supply-chain quelconque	X	X		

9.2.2.2 La réutilisabilité de Taylor

Contrairement au précédent auteur, Simon Taylor ne voit que trois niveaux de réutilisabilité pour un modèle de simulation, mais cela dans le cas de l'utilisation d'un logiciel de simulation commercial pour construire ce modèle – ce qui est notre cas. En observant ces trois niveaux –

réutilisabilité des composants basiques, réutilisabilité de sous-systèmes du modèle, réutilisabilité d'un modèle similaire – il est possible de nouveau de différencier le niveau de réutilisabilité suivant l'objectif de l'utilisateur (Tableau 9.7).

Tableau 9.7 : Variation du niveau de réutilisabilité suivi l'objectif poursuivi pour Taylor

Système étudié	Niveau de réutilisabilité Objectif de l'utilisateur	Niveau 1	Niveau 2	Niveau 3
Pharmacie hospitalière	Obtention d'un modèle dédié	X	X	
	Obtention d'un modèle générique		X	X
Autre que pharmacie	Simulation d'une supply-chain identique dans le déroulement	X	X	
	Simulation d'une supply-chain quelconque	X		

Globalement, les conclusions observables ici sont les mêmes pour Robinson et Taylor. L'obtention d'un modèle dédié pourra se faire par la réutilisabilité de sous-systèmes du modèle (composants, fonctions), mais ne pourra se faire par la réutilisabilité de notre modèle dans sa globalité, ce dernier ne tenant pas compte de spécificités importantes (en termes spatial notamment). A l'inverse, une personne désirant un modèle dont l'aspect spatial n'est pas la priorité sera en mesure de pouvoir réutiliser entièrement notre modèle.

En plus de cette notion, Taylor se préoccupe d'un aspect important pour nous et qui est le temps de mise en œuvre de la réutilisabilité d'un modèle. En effet, être capable de réutiliser un modèle est bienvenu, mais si le temps nécessaire à cette démarche est trop conséquent, le bénéfice peut ne plus être pertinent. Considérons les cas de la réutilisabilité de notre modèle.

Pouvoir générer une nouvelle pharmacie en ne manipulant que des données d'entrées se trouve donc être un grand atout, et permet de gagner un temps conséquent dans la « Réutilisabilité d'un modèle similaire ». Taylor souligne d'ailleurs un point essentiel ici : il est rare d'avoir un modèle pouvant être parfaitement fonctionnel pour deux systèmes distincts si cette pluridisciplinarité n'a pas été pensée dès le début du développement, chose que nous avons tenté de faire.

Pour les deux autres niveaux, les conclusions de Taylor sont applicables directement, à savoir que la réutilisabilité de composants ou de sous-systèmes nécessitera de contrôler et vérifier le fonctionnement de chacun, ce qui sera plus ou moins chronophage suivant la situation. La seule exception pouvant être relevée serait la création d'une nouvelle pharmacie à partir de nos sous-modèles (pôle validation, pôle préparation, etc.), pour qui la réutilisation pourrait être plus rapide s'ils sont gardés en l'état.

Nous allons maintenant aborder la partie pratique de ce chapitre, qui consiste à développer des scénarios de simulation à partir de problématiques concrètes, sur lesquels nous appliquerons notre démarche.

9.3 CADRE GENERAL

A ce niveau, nous possédons tous nos outils et nous sommes dans la capacité de les utiliser pleinement. Il est donc temps de commencer à les exploiter dans l'optique de pouvoir tirer certaines conclusions quant à leur utilisation, et leur influence sur les prises de décisions. Pour cela, nous allons considérer trois scénarios différents portant sur des problématiques diverses que des pharmaciens peuvent rencontrer dans le cadre du développement et du perfectionnement de leur pharmacie. Ces trois scénarios et leurs impacts sont présentés dans ce qui suit, mais avant de se pencher sur ces derniers, il est important de faire le point sur le cadre général de ces scénarios. En effet, un certain nombre d'informations et de paramètres communs à ces scénarios doivent être présentés.

Nous nous situons pour les trois dans le cadre de l'amélioration de la pharmacie du CHU de Clermont-Ferrand, et nous ne nous concentrerons que sur les demandes traitées sur le site Gabriel Montpied : celles de Gabriel Montpied (noté **GM**), celles de l'hôpital Nord et enfin celles du Centre Médico-Psychiatrique (noté **CMP**), soit un total de 120 unités fonctionnelles représentant 1237 lits.

Ce choix impose donc certaines conditions quant aux horaires de travail des préparateurs, la gestion des prescriptions et demandes, etc. Voyons lesquels :

- ❖ Taille de l'équipe préparateur : en moyenne, l'équipe du CHU est composée de huit membres. Nous garderons donc cette valeur pour les simulations ;
- ❖ Horaire de travail des préparateurs : les préparateurs travaillent toujours de 8h à 16h avec une pause de 1h entre 12h et 13h. Chaque jour, un préparateur ne prend pas sa pause en même temps que les autres pour répondre aux demandes urgentes et un autre reste une heure de plus le soir jusqu'à 17h ;
- ❖ Arrivée des demandes : aucune demande d'UF n'est traitée entre 17h et 8h du matin le lendemain. De plus aucune demande urgente, de réapprovisionnement ou de retour produits n'est générée ;
- ❖ Répartition des demandes : chaque jour arrivent en moyenne 126 demandes provenant de 62 UF. De plus, d'après le cadencement défini, il y a en moyenne 45 UF à délivrer obligatoirement chaque jour.
- ❖ Créneaux horaires : les journées de travail sont découpées comme suit : 7h/8h, 8h/9h, 9h/12h, 12h/13h, 13h/14h, 14h/16h, 16h/17h ;
- ❖ Répartition des préparateurs sur la journée : les préparateurs sont répartis durant les journées de travail comme suit :

Tableau 9.8 : Répartition des préparateurs sur une journée de travail

Créneau	7 – 8	8 – 9	9 – 12	12 – 13	13 – 14	14 – 16	16 – 17
Nombre de préparateurs	0	8	8	1	8	8	1

- ❖ Taille de l'équipe valideur : Vu le nombre d'internes en pharmacie et de pharmaciens travaillant au CHU, nous tablons sur une équipe de 15 valideurs, pouvant être répartis comme voulu sur les journées de travail.

Abordons maintenant les demandes des UF. Celles-ci sont soumises informatiquement à la pharmacie et préparées manuellement sans l'aide de robot dans le fonctionnement de base. Nous sommes donc dans le cadre d'un fonctionnement globalisé comme nous l'avons déjà annoncé. Leurs caractéristiques sont présentées dans le Tableau 9.9 suivant. A noter que les formules apparaissant dans ce tableau et les suivants sont sous le formalisme SIMIO ([Annexe 6](#)), et sont obtenues à partir de l'analyse de l'historique des demandes et de leur composition.

Tableau 9.9 : Paramètres des demandes globalisées

Caractéristique	Réponse
Nom de la demande	Globalisée
Probabilité suivant le créneau	Dépend du % des autres types de demandes générées
Hybridité	Non
Validation	Non
Mode de traitement	Manuel
Nombre de produits différents par demande le matin	Loi discrète couplée à des lois uniformes $\text{Random.Discrete}(\text{Math.Round}(\text{Random.Uniform}(1,5)),0.20,\text{Math.Round}(\text{Random.Uniform}(6,10)),0.42,\text{Math.Round}(\text{Random.Uniform}(11,20)),0.65,\text{Math.Round}(\text{Random.Uniform}(21,30)),0.80,\text{Math.Round}(\text{Random.Uniform}(31,40)),0.86,\text{Math.Round}(\text{Random.Uniform}(41,50)),0.91,\text{Math.Round}(\text{Random.Uniform}(51,60)),0.95,\text{Math.Round}(\text{Random.Uniform}(61,70)),0.97,\text{Math.Round}(\text{Random.Uniform}(71,80)),0.99,\text{Math.Round}(\text{Random.Uniform}(81,90)),0.999,\text{Math.Round}(\text{Random.Uniform}(91,100)),1)$
Nombre de produits différents par demande l'après-midi	Loi discrète couplée à des lois uniformes $\text{Random.Discrete}(1,0.479,2,0.625,3,0.653,4,0.678,\text{Math.Round}(\text{Random.Uniform}(5,8)),0.84,\text{Math.Round}(\text{Random.Uniform}(9,10)),0.88,\text{Math.Round}(\text{Random.Uniform}(11,20)),0.93,\text{Math.Round}(\text{Random.Uniform}(21,30)),0.945,\text{Math.Round}(\text{Random.Uniform}(31,40)),0.96,\text{Math.Round}(\text{Random.Uniform}(41,50)),0.980,\text{Math.Round}(\text{Random.Uniform}(51,60)),0.99,\text{Math.Round}(\text{Random.Uniform}(61,70)),0.997,\text{Math.Round}(\text{Random.Uniform}(71,80)),0.998,\text{Math.Round}(\text{Random.Uniform}(81,90)),0.999,\text{Math.Round}(\text{Random.Uniform}(91,100)),1)$
Nombre d'éléments par produit	Loi normale : $\text{Math.Round}(\text{Random.Normal}(10,2))$
Temps de préparation de la demande	$0.7963 * \text{Prescriptions.Nbrproduitdemande} + 8.501$ (dépendant du nombre de produits de la demande)

Les UF délivrées chaque jour (selon un cadencement défini) sont au nombre de 45 en moyenne (de 41 le vendredi à 49 le mardi), chacune d'entre elles ayant la même probabilité d'être approvisionné. A noter qu'il n'y a pas de préparateurs dédiés à certaines UF.

Les informations définissant le cadre général étant maintenant connues, nous pouvons dès lors nous pencher sur la création des divers scénarios. Chacun d'entre eux est présenté en détails et est abordé d'un point de vue pharmacien. Nous entendons par cela que nous allons nous mettre à leur place et utiliser notre démarche. L'utilisation des divers outils ainsi que les nombreuses données nécessaires à la simulation de chaque scénario sont donc détaillées, avant d'analyser les résultats obtenus lors de tests.

Il est important de souligner que pour les scénarios, 100 répliques ont été réalisées dans chaque situation de chaque scénario. Par exemple pour le scénario 2 sur la DJINA, comptant quatre situations, 400 répliques ont donc été menées en tout.

9.4 SCENARIO 1 : IMPACT DE LA DJINM

Le premier scénario concerne la mise en place de la dispensation journalière individuelle nominative manuelle au CHU.

9.4.1 Contexte

Ce type de dispensation est, comme son nom l'indique, entièrement réalisé à la main, c'est-à-dire sans aucune intervention de robots. Nous ne considérons pas ici la validation des prescriptions, nous concentrant uniquement sur la préparation de ces prescriptions.

Le but est donc ici d'évaluer l'impact d'une telle dispensation sur la charge de travail des préparateurs afin de voir si, dans le cadre du CHU de Clermont-Ferrand, l'équipe en place serait suffisante pour supporter une telle démarche. Pour cela, nous augmentons par tranche de 20% (20/40/60/80) la part des demandes d'UF fonctionnant en nominatif. Nous excluons volontairement le 100% nominatif, peu réalisable dans le cadre d'un CHU, notamment à cause de la présence d'UF qui ne peuvent supporter un tel fonctionnement (réanimation, urgences, blocs opératoires, etc.). En regardant de plus près le nombre de lits concernés par ce type de dispensation, sur les sites de GM, de l'hôpital nord et du CMP, cela équivaldrait à 1100 lits soit 88.9% du total des lits de ces trois sites, mais ne représenterait que 37.5% des UF. De plus, une analyse plus fine de la répartition des UF pouvant potentiellement fonctionner en nominatif sur chaque jour de la semaine montre que ces mêmes UF ne représenteraient en moyenne que 50% des UF dispensées pour chaque journée.

Il nous faut donc revenir sur notre choix de faire évoluer de 20% en 20% la part des UF pouvant fonctionner en nominatif. Nous partons par conséquent sur un découpage 10/20/30/40% si les résultats obtenus permettent de faire évoluer ainsi cette part.

9.4.2 Utilisation des outils et de la démarche

En partant d'un fonctionnement entièrement en globalisée pour le CHU de Clermont-Ferrand, un pharmacien voulant faire évoluer ce système à l'aide de notre démarche doit passer par différentes étapes et par l'utilisation de nos différents outils.

9.4.2.1 Phase 1 et Modèle générique des processus

En premier lieu, le pharmacien va utiliser notre modèle générique des processus pour observer le processus de l'évolution qu'il souhaite faire subir à sa pharmacie, c'est-à-dire une préparation nominative des prescriptions. Le processus apparaissant en rouge sur la Figure 9.1 a été identifié durant la seconde étape de la Phase 1 de notre démarche tandis que celui en vert est l'évolution souhaitée par le pharmacien, processus identifié durant la troisième étape de cette même phase.

Sur cette figure, les différences majeures constatées lors de l'étude des deux processus sont au niveau de l'insertion de la prescription dans le système de gestion de la pharmacie (en haut à

gauche) et lors de la préparation de la demande nominative, celle-ci nécessitant des actions supplémentaires (au milieu en bas). En effet, la saisie de la prescription est désormais directement réalisée sur le logiciel de prescription – ce qui nécessitait une recopie auparavant – et lors de la préparation il est désormais nécessaire de préparer des sachets individuels, chacun d’entre eux représentant un patient et une prescription.

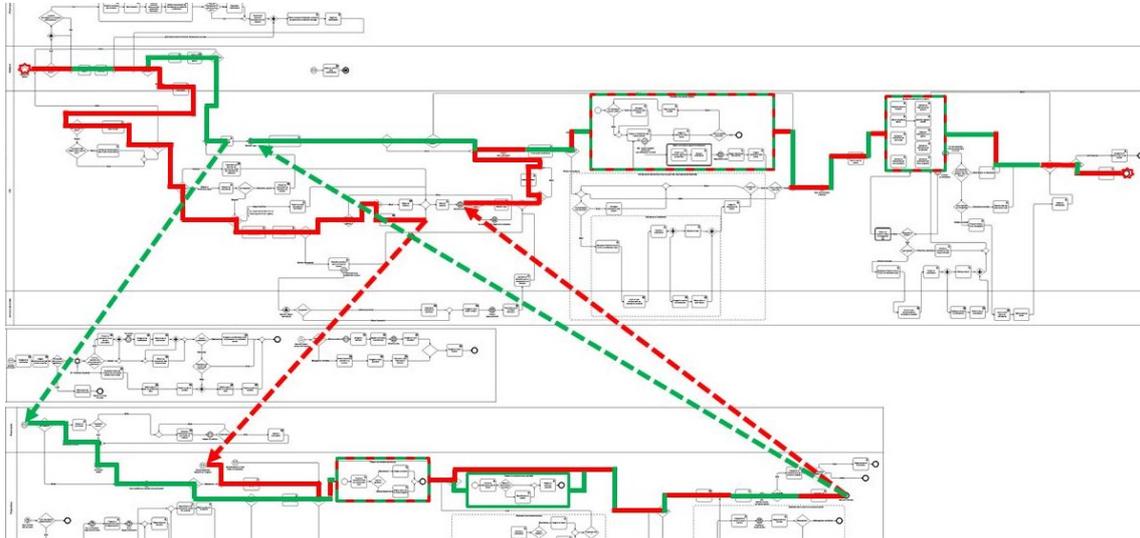


Figure 9.1 : Processus avant (en rouge) et après évolution (en vert)

9.4.2.2 Phase 2 et Diagramme des classes

Le pharmacien décide donc d’appliquer cette évolution, celle-ci ne présentant pas de contraintes structurelles ou organisationnelles insurmontables. Lors de la seconde étape de la Phase 2, le pharmacien va devoir identifier les caractéristiques des nouvelles entités grâce aux diagrammes des classes. Dans ce cas précis, aucun nouvel acteur ne vient intervenir sur le système. Seul l’objet de flux va évoluer en passant d’une demande d’UF à une demande nominative (prescription). La Figure 9.2 présente le diagramme des classes dont les portions non utilisées en DJINM ont été grisées.

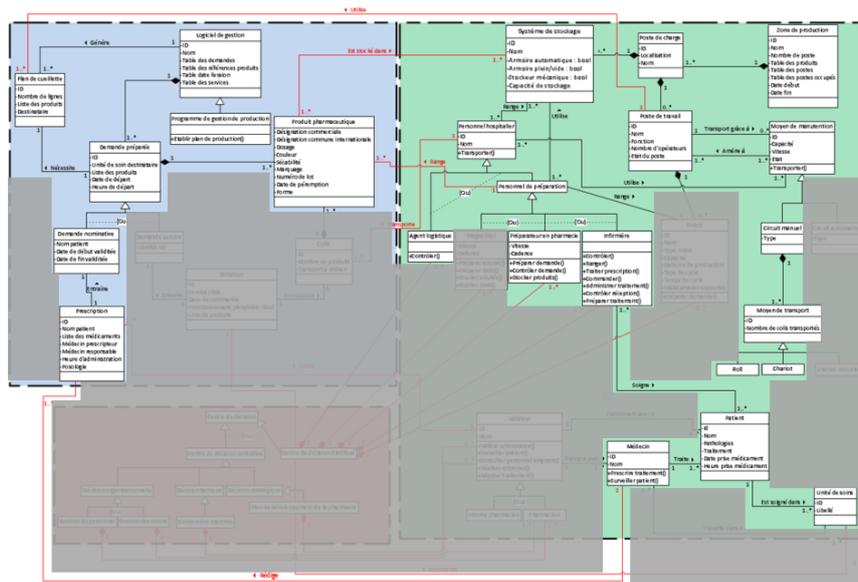


Figure 8.2 : Vue du SSPDL utilisé dans de le cas de la DJINM (parties non grisées)

En observant ce diagramme des classes (Figure 9.3), il est constaté que la demande nominative nécessite des informations complémentaires à celles de la demande d'UF (de service sur la Figure 9.3). Cependant mis à part ces informations, aucune modification majeure n'est constatée.

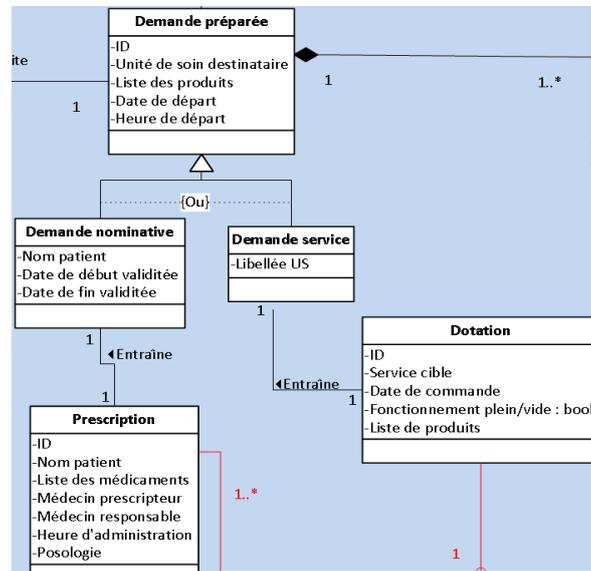


Figure 9.3 : Vue des classes « Demande nominative » et « Demande de service »

9.4.2.3 Phase 3 et Paramétrage de la simulation

Le pharmacien connaissant son nouveau processus et les changements qu'il doit apporter au niveau des entités constituant son système, il lui est alors possible de passer à la simulation de ce nouveau fonctionnement. La seule variable qu'il y a dans notre scénario de simulation est la part de demande d'UF en nominatif que nous ferons évoluer pour faire monter en charge la préparation individuelle. Le pharmacien doit donc saisir un certain nombre d'informations dans l'interface d'entrée pour pouvoir paramétrer notre modèle. En plus de celles d'ordre général qui sont présentées dans le Tableau 9.9, le Tableau 9.10 dresse le bilan des informations nécessaires à ce scénario.

Tableau 9.10 : Données nécessaires pour une simulation en DJIN manuelle

Caractéristique	Réponse
Nom de la demande	DJ_UF
Probabilité suivant le créneau	10%, 20%, 30% ou 40% selon scénario
Hybridité	Oui
Sous-demandes nominatives	
Pourcentage	100%
Validation	Non
Nombre de sous-demandes	Loi uniforme : $\text{Math.Round}(\text{Random.Uniform}(20,30))$
Nombre de produits par sous-demande	Loi normale : $\text{Math.Round}(\text{Random.Normal}(10,2))$
Nombre d'éléments par produit de la sous-demande	Loi triangulaire : $\text{Math.Round}(\text{Random.Triangular}(1,2,3))$
Temps de préparation	Loi uniforme : $\text{Math.round}(\text{Random.Uniform}(5,10))$
Mode de préparation	Manuel

Validation	Non
Mode de traitement	Manuel
Nombre de produits différents par demande le matin	Loi normale : $\text{Math.Round}(\text{Random.Normal}(15,1))$
Nombre de produits différents par demande l'après-midi	0
Nombre d'éléments par produit	Loi normale : $\text{Math.Round}(\text{Random.Normal}(5,2))$
Temps de préparation de la demande	$1,0908 * \text{Prescriptions.Nbrproduitdemande} + 11,723$

Plusieurs précisions sont nécessaires sur les différentes valeurs de ce tableau. Tout d'abord le nombre de produits d'une demande l'après-midi est de zéro car nous supposons qu'aucune demande d'UF en nominatif n'est émise sur cette période. En effet, elles le seront toutes le soir ou le matin (donc avant l'arrivée des préparateurs), c'est-à-dire après l'arrivée des patients durant l'après-midi.

De surcroît, l'impact du passage de certaine UF en nominatif va aussi avoir une influence sur les demandes qu'elles émettent, demandes qui ne vont bien entendu pas complètement disparaître, certains produits n'étant pas dispensés à titre nominatif. En étudiant les relevés de commandes et divers rapports, nous sommes parvenus à une estimation du contenu des demandes restantes hors produits nominatifs. Cette quantité serait d'environ 9.28 lignes de produits pour les UF étant désormais en nominatif. Les demandes d'une UF seront donc composées d'un nombre variable de sous-demandes nominatives et d'une demande d'UF comportant 15 lignes afin de se donner une marge de sécurité. Nous allons par conséquent utiliser de nouveau une loi normale pour modéliser le contenu de ces demandes. Celle-ci sera « $\text{Math.Round}(\text{Random.Normal}(15,1))$ » pour ces demandes. Le temps de préparation ne sera pas identique à celui des demandes globalisées, les services en nominatif utilisant dès lors des armoires plein/vide, ce qui impacte le temps, qui devient « $1,0908 * \text{Prescriptions.Nbrproduitdemande} + 11,723$ ». Ainsi pour une demande de 15 produits, le temps passe de 20,45 minutes à 28,08 minutes.

En ce qui concerne les informations des sous-demandes nominatives, divers rapports, ministériels ou émis par des hôpitaux, ont conclu qu'une prescription contient en moyenne 10 lignes, soit 10 produits, cela étant dépendant du type de l'UF émettrice et des soins procurés bien entendu. Nous utiliserons cette estimation, tout en la modulant grâce à une loi normale pour obtenir la formule du nombre de produits « $\text{Math.Round}(\text{Random.Normal}(10,2))$ ».

Le nombre de sous-demandes nominatives a été quant à lui obtenu par l'étude des cadencements de dispensation et du nombre de lits par UF. Il apparaît que les UF pouvant fonctionner en nominatif comprendront entre 16 et 80 lits. Pour traduire cela nous utiliserons une loi uniforme de bornes 20 et 30 lits (65% des UF étant comprises dans cet intervalle), le nombre moyen de lits pour ces UF étant de 25. Ainsi 10% des lits en nominatif à l'hôpital, représentent 124 lits (sur 1100 lits éligibles à ce mode de dispensation), ce qui avec 25 lits de moyenne correspond à 5 UF.

Pour obtenir le temps de préparation d'une telle sous-demande, la seule valeur relevée pour un établissement similaire au CHU de Clermont-Ferrand est celle issue de [ANAP 2012], et qui concerne le Centre Hospitalier Intercommunal de Cornouaille (noté **CHIC**) de Quimper-Concarneau. Pour celui-ci, la préparation d'une demande nominative serait située, d'après des relevés terrains, aux alentours de 5 minutes. De même, le rapport [DHOS 09] a relevé à travers ses différentes

interviews et études qu'en moyenne dans un CHU le temps préparateur alloué à la préparation des prescriptions de 100 lits est de neuf heures soit environ 5.4 minutes par lit. Par sécurité, et étant donné que la disposition spatiale des différentes infrastructures dans une pharmacie peut impacter fortement le temps de préparation d'une telle demande, nous utiliserons la loi de distribution « $Math.round(Random.Uniform(5,10))$ » afin de tenter de modéliser plus fidèlement ce temps de préparation qu'il nous est impossible de quantifier. Le temps sera ainsi compris entre 5 et 10 minutes de préparation.

Le pharmacien possède désormais toutes les données nécessaires à l'exécution de ce premier scénario à l'aide de notre modèle de simulation. Les résultats de cette simulation et leur exploitation seront présentées dans la partie suivante.

9.4.3 Résultats de la simulation

Comme nous l'avons énoncé, le but de ce scénario est de voir s'il est possible de mettre en place la dispensation individuelle nominative au CHU, cette dernière étant réalisée manuellement.

La logique aurait voulu que soit testé différents niveaux de dispensation en faisant évoluer sa part dans le système de 10% à 50% des lits. Malheureusement, comme nous allons le voir, seule la part des 10% aura été simulée, la charge pesant sur les préparateurs étant déjà à la limite.

9.4.3.1 Résultats globaux

En premier lieu, et avant d'aborder les préparateurs, regardons si le modèle de simulation a pu répondre à l'exigence de 10% des lits en nominatif – équivalent à 110 lits au total.

Sur les 100 réplifications réalisées (pour le cas des 10% uniquement), il y a eu en moyenne 4,8 UF fonctionnant en nominatif pour un volume moyen de 114,8 lits. Nous sommes donc proches de l'objectif. De même, l'analyse des demandes des UF au CHU montrait en moyenne 126 demandes passées par jour à la pharmacie, qui dans le cas de notre modèle s'établit à 120,14 en moyenne générées à chaque réplification.

Autres informations intéressantes, mais difficilement mesurables sur le système réel, le temps d'attente maximum d'une demande (hors prescription) pour être préparée sera de 1h31min – en moyenne – et le temps moyen passé par cette même demande dans la pharmacie sera compris entre 35 minutes et 4h48min – avec une moyenne de 2h40min.

En ce qui concerne les demandes « DJ_UF » produites, nous sommes à 15,23 produits en moyenne, soit une nouvelle fois proche de la valeur obtenue par analyse du système réel.

De ces premiers résultats, il est donc possible de conclure que le modèle a travaillé avec des données proches de celles de notre système réel. Hormis le temps de présence d'une demande dans le système, les nombreuses autres informations fournies par SIMIO ne se montrent pas réellement utiles quant à l'étude de ce scénario.

9.4.3.2 La charge des préparateurs

Nous abordons ici l'objectif principal de ce scénario. En effet, le critère qui permettra de dire si oui ou non, il est possible et pertinent de faire de la DJINM est la charge de travail des divers préparateurs. Il est admis qu'une charge de travail acceptable pour un opérateur est aux alentours de 80%. Voyons si cela est possible dans le cadre du CHU (Tableau 9.11).

Tableau 9.11 : Charge des préparateurs pour 10% des lits en DJINM

Préparateur	% occupation	Temps occupation (en heure)	Nombre demandes préparées
Préparateur [1]	88,63	7,82	27,86
Préparateur [2]	79,75	7,1	29,59
Préparateur [3]	78,91	7,12	28,65
Préparateur [4]	81,85	7,16	29,35
Préparateur [5]	83,32	7,07	29,03
Préparateur [6]	79,23	7,6	30,52
Préparateur [7]	78,83	7,2	27,86
Préparateur [8]	78,16	7,07	28,58

La période de travail étant de huit heures par jour avec une pause le midi devant osciller entre 30 et 45 minutes – de 0,5 à 0,75 h – il paraît évident à la vue du temps et du taux d'occupation des préparateurs que cette configuration de dispensation est dans la limite haute de ce qui est acceptable pour eux. En effet, le taux d'occupation ne tombe pas ici en dessous de 78% (Figure 9.4). La surcharge sur le « Préparateur [1] » s'explique par le fait que c'est ce dernier qui est d'astreinte le midi.

Pourcentage d'occupation

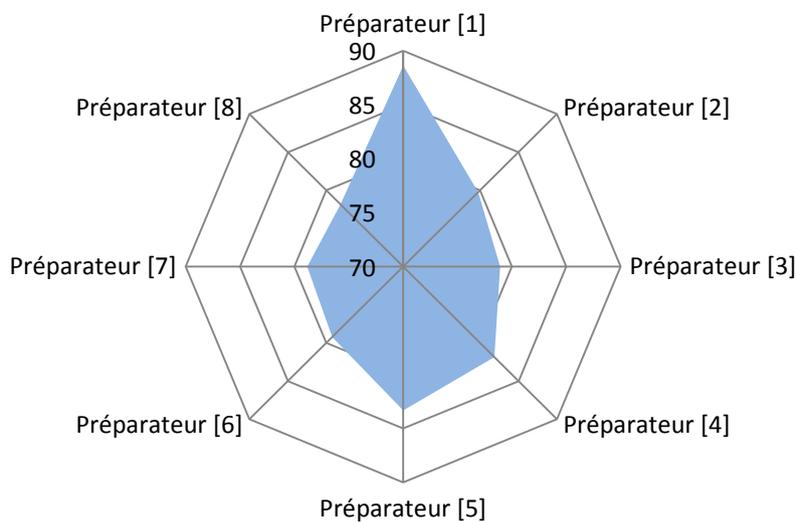


Figure 9.4 : Charge de travail des préparateurs

Enfin, en ce qui concerne le nombre de demandes (nominatives ou d'UF) traitées, il est constaté que la répartition sur les préparateurs est plutôt uniforme, cette valeur oscillant autour de 29 demandes (Figure 9.5).

Nombre demandes préparées

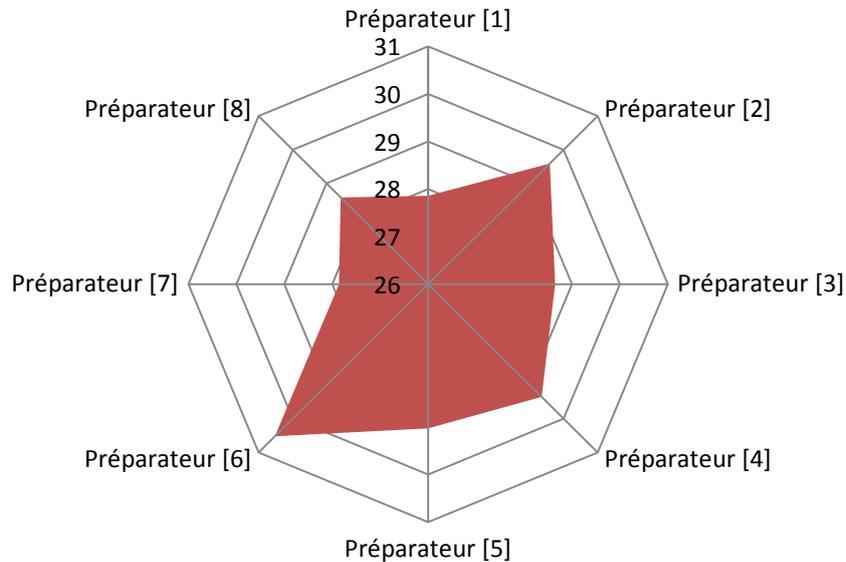


Figure 9.5 : Nombre de demandes préparées par chaque préparateur

9.4.4 Conclusion

Quelles informations est-il possible de tirer de ces premiers résultats ? Comme attendu de la part des pharmaciens, il sera difficile de mettre en place la DJINM au CHU de Clermont-Ferrand, la charge de travail des préparateurs étant difficilement compatible avec ce fonctionnement. En effet, faire passer seulement 10% des lits en nominatif, augmente d'environ 5-10% la charge pesant sur ces derniers – comparativement au fonctionnement actuel. Cette conclusion rejoint les constats faits dans la littérature, à savoir que la plupart des établissements en DJINM sont de taille réduite et avec une variabilité très faible dans le type de lits supportés.

Cependant, utiliser la DJINM n'est pas pour autant impossible au CHU, mais cela imposerait de revoir avec rigueur non seulement la taille de l'équipe préparateur – pour supporter l'impact du passage en DJINM – mais aussi de repenser la répartition des UF. En effet, le scénario ici utilisé, répartissait aléatoirement les demandes sur tous les préparateurs afin d'optimiser les capacités de production de la pharmacie, situation ne représentant pas la réalité, des UF étant assignées aux préparateurs.

9.5 SCENARIO 2 : IMPACT DE LA DJINA

Ce second scénario va concerner la mise en place de la dispensation journalière individuelle nominative automatisée, et cela de nouveau au CHU de Clermont-Ferrand.

9.5.1 Contexte

Afin de faire le parallèle avec le scénario précédent, supposons que notre même pharmacien désire désormais passer de son fonctionnement en globalisée à une dispensation de nouveau

nominative et journalière mais cette fois-ci préparée à l'aide d'un robot de dispensation. Toutes les demandes nominatives seront donc préparées par un robot, ne nécessitant plus la présence de préparateurs. De nouveau, nous ne nous préoccupons pas de la validation des prescriptions.

L'objectif de ce scénario reste cependant le même que pour le précédent, c'est-à-dire d'évaluer l'impact d'une telle dispensation sur la charge de travail des préparateurs.

Afin de voir si l'équipe en place serait suffisante pour supporter cette démarche, nous utiliserons une nouvelle fois les tranches de 10% (10/20/30/40) de demandes nominatives pour tester différentes situations et monter en charge.

9.5.2 Utilisation des outils et de la démarche

Comme précédemment les trois outils vont jaloner les actions du pharmacien.

9.5.2.1 Phase 1 et Modèle des processus

Sur la Figure 9.6 ci-dessous, sont présentés en rouge le processus initial, correspondant toujours à la dispensation globalisée, et en bleu, celui de la DJINA.

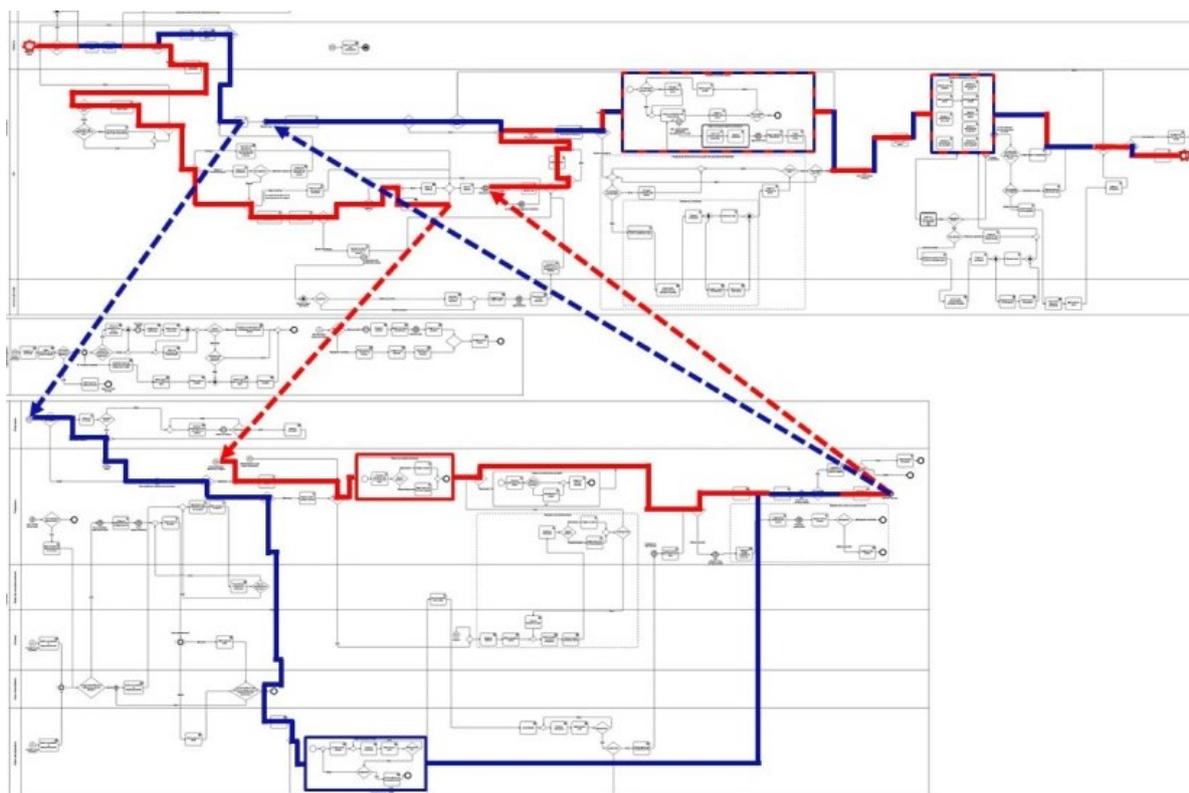


Figure 9.6 : Vue du processus actuel (en rouge) et de la DJINA (en bleu)

Les différences remarquables ici concernent comme pour le premier scénario l'insertion de la prescription dans le système de gestion de la pharmacie et la préparation de la demande nominative. La saisie de la prescription est réalisée de nouveau directement sur le logiciel de prescription tandis que la préparation des demandes nominatives transite par le robot de dispensation situé en bas de notre modèle générique des processus, ce robot préparant des anneaux ou des boîtes destinés chacun à un patient en particulier.

9.5.2.3 Phase 3 et Paramétrage de la simulation

Comme pour le scénario précédent, l'unique variable intervenant lors des tests sur ce scénario concernera la part des UF fonctionnant en nominatif. Toutes les informations présentes dans la partie 9.4.2.3 du scénario précédent seront réutilisées ici, ces deux scénarios étant très proches. En effet, seule la méthode de préparation change, ce qui impose comme seul changement lors de la saisie des données dans l'interface d'entrée, de paramétrer un robot et de passer le « *Modetraitement* » des sous-demandes nominatives en « *Auto* ». Ainsi en plus des informations des Tableaux 9.8 à 9.10, celles du Tableau 9.12 sont à rajouter :

Tableau 9.12 : Données liées au robot pour la production automatisée

Caractéristique	Réponse
Nom total de robots de dispensation	1
Cycle 1	
Nom du cycle	Cycle_standard
Début du cycle	9h
Fin du cycle	15h
Cadence des robots en dose/heure	1500
Type de cycle des robots	Cycle 1
Nombre de robots fonctionnant sur ce cycle	1

Les heures de début et de fin du cycle du robot ont été choisies en fonction des horaires des préparateurs. En effet, les préparateurs commençant à 8h et terminant à 16h, nous souhaitons que le cycle du robot ait lieu sur cette période. Le décalage d'une heure au début et à la fin permettent de laisser aux préparateurs le temps de mettre en route et en veille le robot en début et fin de journée. L'utilisation de tels horaires implique aussi que les traitements préparés seront pour une prise soir/matin/midi en théorie.

En ce qui concerne la cadence du robot, nous avons choisi cette valeur dans la fourchette basse de capacité de production des différents robots existants sur le marché (de 600 à 3600 prescriptions par jour suivant le robot et la taille des prescriptions). Enfin, nous nous limiterons à un seul robot, peu d'hôpitaux pouvant s'offrir plusieurs exemplaires de ces machines coûteuses.

Dès lors, en utilisant ces données et celles présentées dans les parties précédentes, la simulation de scénario peut démarrer, ses résultats étant présentés dans la partie suivante.

9.5.3 Résultats de la simulation

Comme pour son prédécesseur, le but était ici d'étudier l'impact du passage des services en nominatif mais cette fois-ci avec une préparation automatisée des prescriptions, ce qui se révélera bien entendu une économie de temps pour les préparateurs, ou tout du moins, un changement qui ne fera pas varier leur charge de travail.

Lors des simulations, nous avons constaté en premier lieu que la charge des préparateurs n'augmentait pas par rapport au fonctionnement actuel, celle-ci se situant aux alentours des 70%. Ce

ne sera donc pas ce critère qui sera limitant, contrairement au scénario 1, quant au choix fait sur la part d'UF en nominatif.

Par conséquent, et en toute évidence, le critère limitant sera ici la capacité de production du robot qui ne pourra traiter qu'un certain nombre de prescriptions par jour. Il est à noter que notre scénario sous-utilise quelque peu ce robot, ce dernier ne fonctionnant que six heures par jour au maximum. Ce choix se justifie, afin de ne pas trop embellir l'impact d'une telle solution sur les performances de la pharmacie. Nous préférons rester prudents quant aux conclusions qui pourraient en être tirées.

La capacité de production du robot étant limitante, nous verrons dans la partie suivante que si 40% des UF sont en nominatif, le taux d'occupation du robot aura dépassé 100%. De plus, si 30% des services sont en nominatif, le taux d'occupation ne sera pas de 100%. Il sera donc nécessaire de chercher dans cette plage, la part adéquate des services en nominatif. Nous sauterons donc la tranche de 20% des UF en nominatif et ne nous intéresserons qu'aux tranches 10/30/40%, le taux d'occupation optimale du robot se trouvant pour une part d'UF étant comprise entre 30 et 40%.

9.5.3.1 Résultats globaux

Tout comme pour le scénario 1, faisons le tour des diverses informations obtenues par le biais de la simulation. Le Tableau 9.13 synthétise un certain nombre d'informations.

Tableau 9.13 : Nombre de demandes nominatives et d'UF traitées par la pharmacie

Part d'UF en nominatif	Nombre moyen de prescriptions	Nombre moyen de DJ_UF	Nombre moyen de demandes	Temps maximum passé par une demande à la pharmacie
10%	114,8	4,8	120,14	3 h 56 min
30%	348,48	14,09	119,81	3 h 2 min
40%	481,32	19,31	117,79	3 h 31 min

Tout d'abord, le nombre de prescriptions générées dans chaque situation correspond aux attentes pour un nombre d'UF en nominatif de 10% (124 lits), 30% (372 lits) et de 40% (494 lits). Le nombre de lits reste aux alentours de 25 pour chaque UF, et enfin le nombre de demandes est de nouveau proche de ce qui est attendu.

Le second point concerne le temps maximum passé par une demande dans le système. Cette valeur est légèrement inférieure au cas de la préparation manuelle mais reste quelque peu conséquente. De plus, ce temps n'est pas ou peu influencé par la part des services en nominatif, celui-ci étant plus petit pour 30% que pour 10% et 40%.

Enfin, comme pour le premier scénario nous sommes à 15,21 produits par « DJ_UF », soit proche de la réalité.

De nouveau, nous pouvons ici conclure que le modèle a généré et utilisé des données proches de la réalité. Intéressons-nous désormais au robot et aux préparateurs.

9.5.3.2 La charge des acteurs

Contrairement au scénario précédent où les préparateurs étaient le facteur limitant du système, cette fois-ci, il s'agit du robot de dispensation et plus précisément de sa capacité de

production. Le Tableau 9.14 suivant présente ainsi la charge de travail de ces différents acteurs (à noter que le « Préparateur [8] » travaille toute la journée sur le robot).

Tableau 9.14 : Charge des préparateurs et du robot en DJINA

Acteur/Situation	% occupation			Temps occupation (en heure)			Nombre demandes préparées		
	10%	30%	40%	10%	30%	40%	10%	30%	40%
Robot	22,67	72,5	100	1h36min	4h19min	6 h	114,8	348,48	450,86
Préparateur [1]	75,99	76,10	77,91	7h35min	7h37min	7h47min	18,35	17,88	17,97
Préparateur [2]	68,44	68,92	70,77	6h50min	6h53min	7h04min	16,34	16,55	16,26
Préparateur [3]	68,46	68,72	70,75	6h51min	6h52min	7h04min	16,36	16,21	15,95
Préparateur [4]	68,80	69,01	70,92	6h53min	6h54min	7h04min	16,42	16,55	16,28
Préparateur [5]	68,42	68,89	70,25	6h52min	6h53min	7h03min	16,37	16,25	16,33
Préparateur [6]	72,55	72,87	74,65	7h15min	7h17min	7h28min	18,18	17,86	17,88
Préparateur [7]	69,39	68,51	70,54	6h56min	6h51min	7h03min	16,67	16,32	16,14

Comme nous l'avons déjà énoncé, la charge des préparateurs n'évolue pas en fonction de la part d'UF en nominatif. Cette charge de travail reste d'ailleurs dans la zone d'acceptabilité en tournant autour de 70% (Figure 9.9), excepté pour le « Préparateur [1] » toujours d'astreinte le midi.

En ce qui concerne le robot, il est bien constaté que dans les situations avec 10 et 30% des UF en nominatif, celui-ci est sous utilisé (Figure 9.9). Cependant dans le cas des 40%, il est sollicité sans interruption pendant six heures, voire même un peu plus pour terminer de préparer toutes les prescriptions, trente d'entre elles restant en moyenne en attente devant le robot à 15 heures (Figure 9.10).

Pourcentage d'occupation

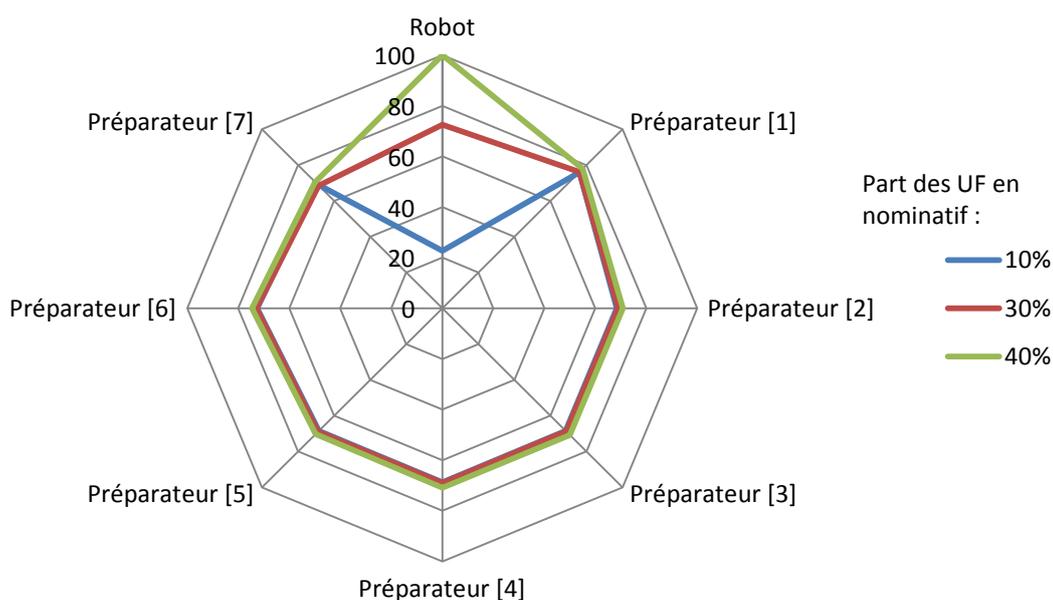


Figure 9.9 : Charges de travail des préparateurs

Nombre de demandes préparées par le robot

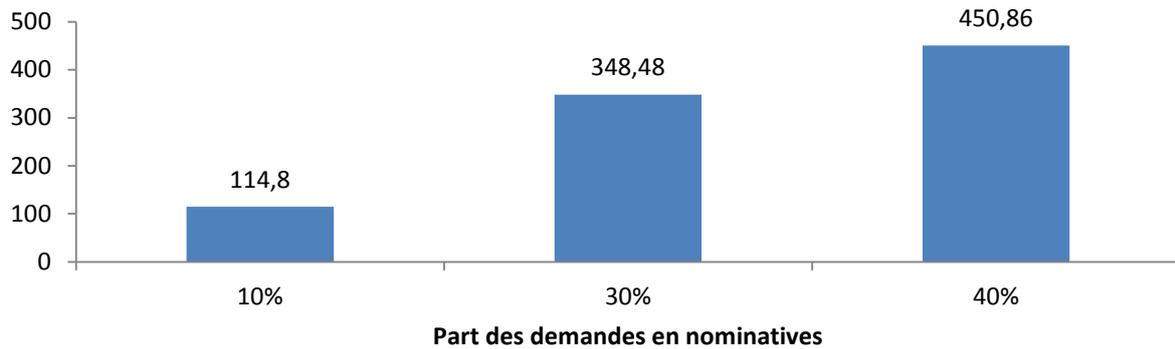


Figure 9.10 : Nombre de demandes préparées par le robot

Les préparateurs quant à eux, traitent dans leur ensemble environ 17 demandes quelle que soit la situation, ce qui représente donc en moyenne 119 demandes (Figure 9.11).

Nombre de demandes préparées (sauf robot)

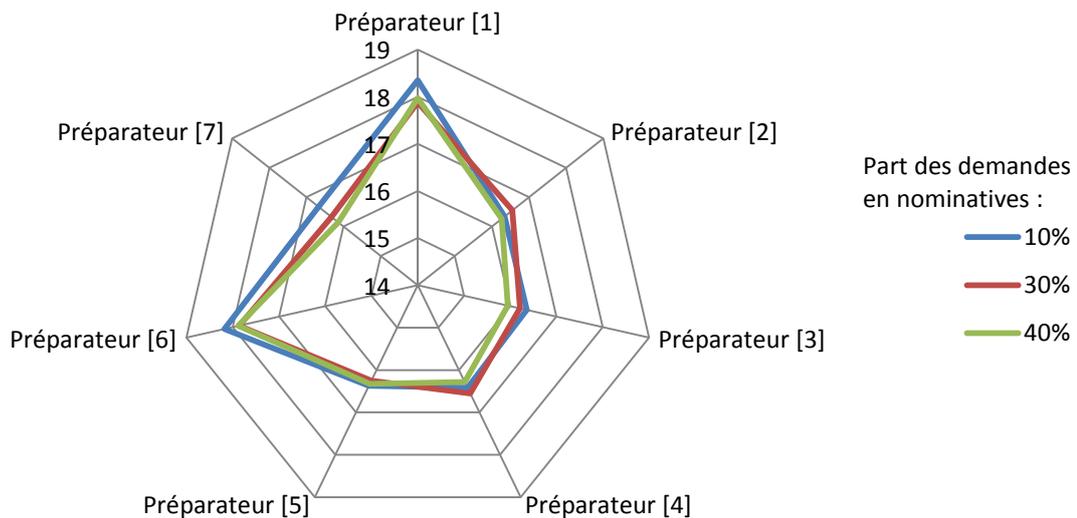


Figure 9.11 : Nombre de demandes préparées par chaque préparateur

Pour 40% des UF en nominatif, notre robot étant sollicité à 100%, il serait nécessaire de diminuer légèrement la part des UF en nominatif afin de diminuer le nombre de prescriptions arrivant à la pharmacie. En effet, contrairement à notre modèle où il est possible de jouer sur le nombre de prescriptions (équivalent au nombre de lits) liées à nos UF, ceci est impossible dans la réalité. Ainsi, une part d'UF autour de 34% en nominatif, soit 420 lits sur 1237, permettrait de solliciter le robot dans une moindre mesure et de se donner une légère marge de manœuvre.

Une rapide application de ce changement sur les données de notre modèle de simulation permet d'obtenir les résultats suivants :

- ❖ Taux d'occupation du robot : 94,33% ;
- ❖ Temps d'occupation : 5h 40 minutes ;
- ❖ Nombre de prescriptions traitées : 426,35.

Nous sommes donc bien dans la situation souhaitée, le nombre de lits préparés étant même légèrement supérieur.

9.5.4 Conclusion

Contrairement au premier scénario où dès la première part de lits en nominatif nous atteignons les limites de la capacité de production du CHU, nous avons pu cette fois-ci, grâce au robot, étudier les tranches supérieures et atteindre une part de lits en nominatif conséquente. En effet, dans le cas d'une sollicitation à 94% du robot, le CHU serait en mesure de passer 34,45% de ses lits en nominatifs soit environ 17 UF. Bien que cette dernière valeur puisse paraître faible, elle donne quand même un bon indice sur la pertinence de l'implantation des robots dans les structures hospitalières.

De surcroît, le robot utilisé ici reste de capacité de production limitée, comparativement à certaines solutions sur le marché. De même, ce dernier ne fonctionne que sept heures par jour, alors qu'il pourrait fonctionner 8 heures par jour, s'il pouvait être accompagné par un préparateur sur la même durée. De plus, suivant la période de réapprovisionnement choisie, et suivant la période de production (matin, après-midi, soir, nuit), le rendement d'un tel robot pourrait encore s'accroître.

Pour information, un cycle de sept heures permettrait de préparer environ 568 lits, soit 45,91% du total.

Dernier point, il est constaté en comparant les temps de présence moyens des demandes à la pharmacie que la méthode de préparation n'influe que peu sur ce temps. En effet, bien que les DJ_UF sortent plus rapidement en mode automatisé, le reste des demandes est présent à la pharmacie approximativement la même durée pour les deux fonctionnements. Par conséquent, une préparation manuelle des prescriptions – dans la situation des 10% - ne ralentit que peu le temps de traitement des autres demandes.

9.6 SCENARIO 3 : COMMENT REPARTIR SA VALIDATION ?

Ce troisième et dernier scénario va porter sur une portion plus précise du CDM. Celui-ci va concerner la validation des prescriptions par une équipe de pharmaciens (sénior ou junior).

9.6.1 Contexte

Nous allons cette fois-ci nous intéresser à la validation des prescriptions par les pharmaciens, étape importante dans le processus de dispensation nominative.

La validation pouvant intervenir sur divers types de dispensation – bien qu'étant obligatoire en DJIN et sur les produits dangereux – l'objectif de ce scénario n'est pas de se pencher sur la partie préparation mais uniquement sur cet aspect de contrôle des prescriptions. L'aspect préparation ne doit pas pour autant être négligé, validation et préparation étant fortement liées. En effet, il sera supposé que les deux devront avoir lieu la même journée, afin d'éviter au maximum les écarts entre ce qui est prescrit et ce qui est préparé. Par conséquent, en préambule, il est déjà possible de dire qu'une préparation exécutée rapidement nécessitera une validation toute aussi rapide. Il en sera de

même pour une préparation lente. Le mode de préparation aura donc une influence et nous privilégierons dans notre cas une préparation automatisée.

L'objectif est donc ici de déterminer avec un nombre de valideurs fixe, la meilleure répartition de ces derniers sur la journée, et cela en fonction de la capacité de production en aval.

9.6.2 Utilisation des outils et de la démarche

9.6.2.1 Phase 1 et Modèle des processus

La Figure 9.12 ci-dessous, présente le processus standard du CHU (en rouge) ainsi que le processus incorporant la validation (en bleu). La partie validation apparaissant en orange pour souligner la différence avec la Figure 9.6.

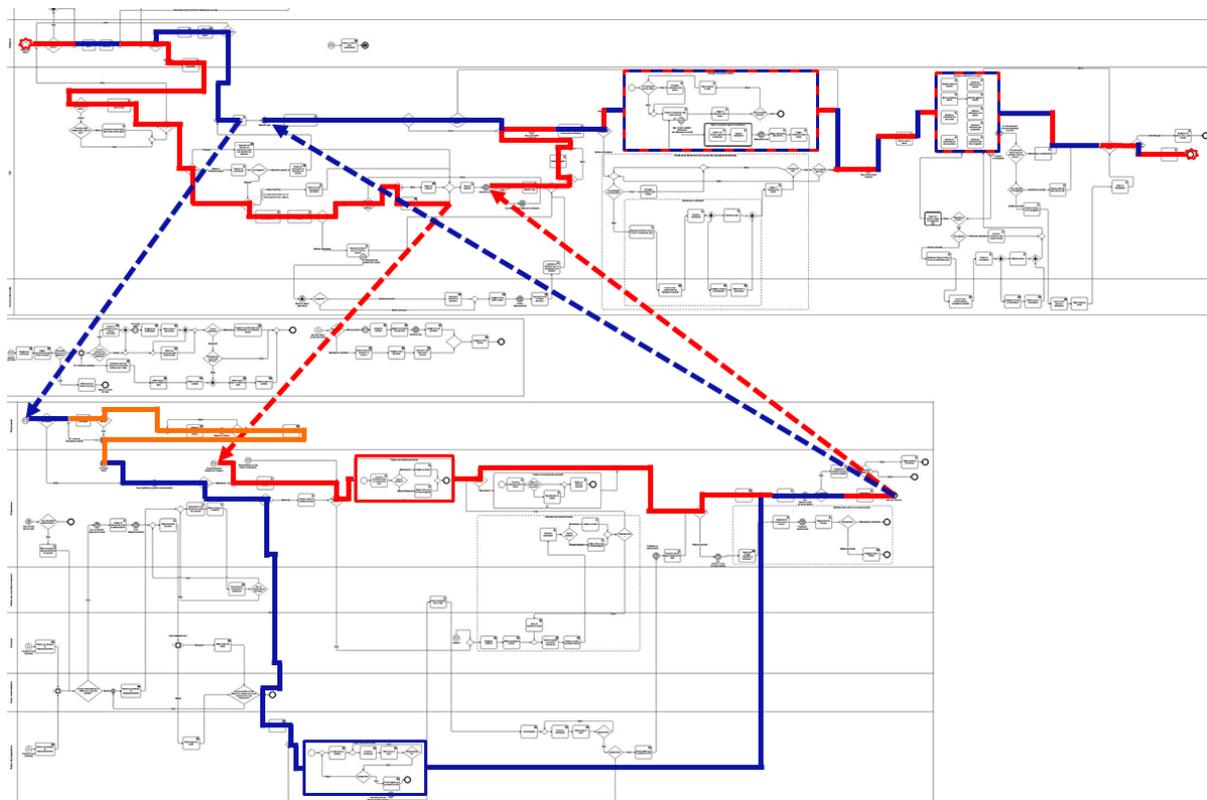


Figure 9.12 : Vue du processus de validation (en orange)

Les différences remarquables ici sont donc identiques à celles faites sur la Figure 9.6 auxquelles s'ajoutent la portion de validation en orange, qui n'autorisera la poursuite du processus que si cette étape de validation est franchie.

9.6.2.2 Phase 2 et Diagramme des classes

Lors du second scénario, les changements étaient divers (robot, demande nominative, etc.). Cette fois-ci, une nouvelle évolution technologique importante prend place dans ce fonctionnement, et qui est la validation informatique des prescriptions. Un tel changement demandera donc un déploiement informatique conséquent au niveau des services mais aussi au niveau de la pharmacie.

En supposant que cela soit réalisable, l'application de la phase aura comme conséquence l'identification d'un changement d'objet de flux (comme pour les scénarios 1 et 2) et surtout l'apparition d'un nouvel acteur agissant sur cet objet de flux, à savoir le valideur, dont les caractéristiques apparaissent sur la Figure 9.13.

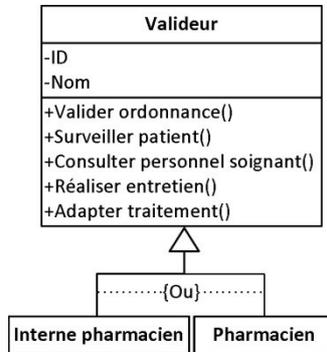


Figure 9.13 : Informations liées à la classe Valideur

La Figure 9.14 présente le diagramme des classes dont les portions non utilisées en DJINA avec validation ont été grisées.

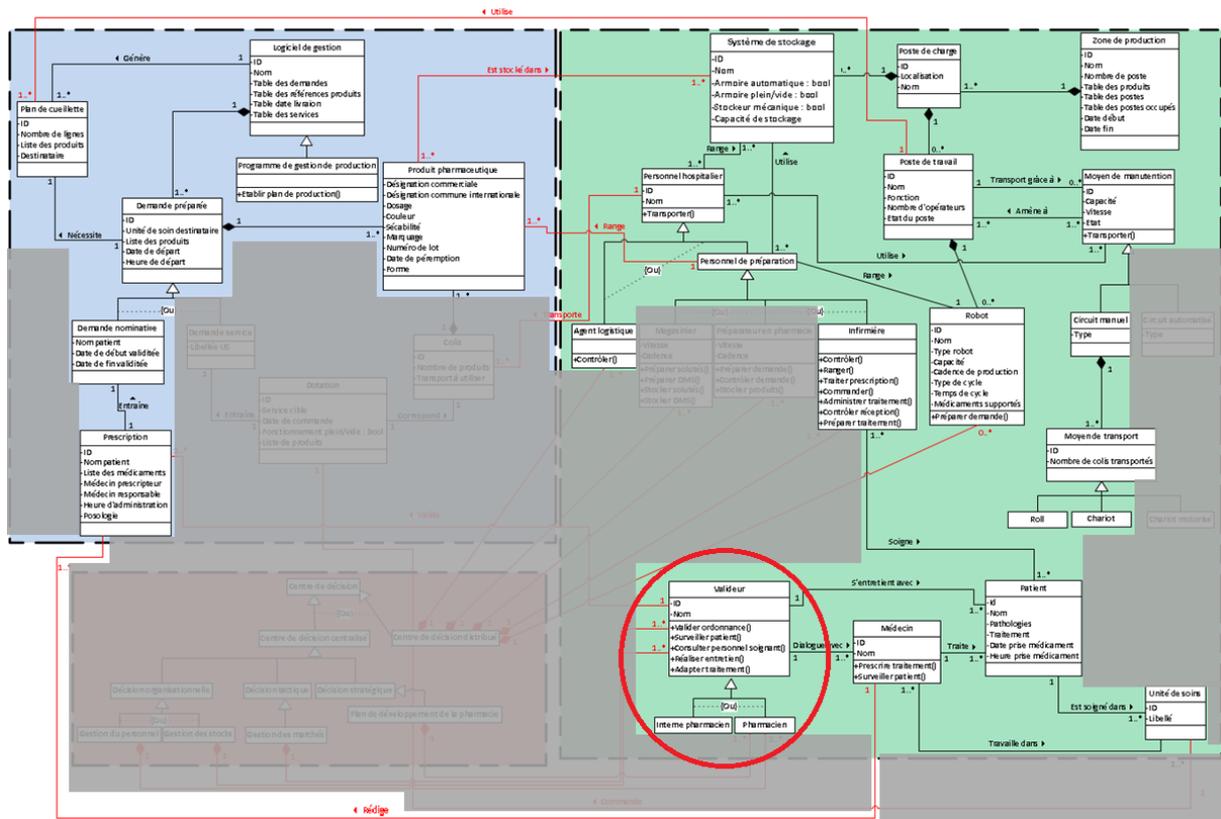


Figure 9.14 : Vue du SSPDL utilisé dans de le cas de la DJINA avec validation (parties non grisées)

Le valideur est donc majoritairement caractérisé par des fonctions et non pas par des caractéristiques comme pour le robot dans le scénario 2.

9.6.2.3 Phase 3 et Paramétrage de la simulation

Ce scénario réutilisera lui aussi l'ensemble des données du second, auxquelles s'ajouteront celles de la validation et des valideurs dont les caractéristiques sont résumées dans les Tableaux 9.15 et 9.16.

Les données présentées dans le Tableau 9.15 au niveau de l'analyse des prescriptions sont issues pour la plupart du rapport national Securimed [DHOS 09], et restent donc des moyennes, ce qui ne représente en rien le fonctionnement d'une structure en particulier. Dans ce rapport, il est précisé que 88% des prescriptions nécessitent une analyse, d'où notre « Niveau 1 » avec 12% de probabilité et un temps d'analyse de zéro. La répartition entre les deux autres niveaux d'analyse a été faite comme suit : toutes les prescriptions ne sont pas nouvelles chaque jour et ne redemandent pas une analyse complète. Nous supposons donc que la part maximum des prescriptions devant être revues complètement – donc « Niveau 3 » – sera de 44%, et que la part du « Niveau 2 » sera également de 44%.

Tableau 9.15 : Données liées à l'analyse et la validation d'une prescription

Caractéristique	Réponse
Part des prescriptions à analyser	Entre 15 et 88% pour un CHU
Part de prescriptions analysées	
	Niveau 1 0,12
	Niveau 2 0,88 - Niveau 3
	Niveau 3 De 0,01 à 0,44
Temps pour valider les prescriptions (en minutes)	
	Niveau 1 0
	Niveau 2 Loi triangulaire : Random.Triangular(4,5,6)
	Niveau 3 Loi triangulaire : Random.Triangular(6,7,8)

Les données des valideurs (Tableau 9.16) seront, quant à elles, dépendantes des différents scénarios que nous établirons. Plusieurs situations seront testées telles que tous les valideurs travaillant en même temps, ou bien étalés sur toute la journée. Les seules informations limitantes sont que nous partirons sur une base de quinze valideurs (sénior ou interne) dont les horaires de validation seront compris entre 8h et 15h avec une période de travail minimum de trente minutes.

Tableau 9.16 : Données liées aux valideurs

Caractéristique	Réponse
Valideur 1 à 6	
	Nombre Entre 1 et 15
	Heure de début de validation Entre 8h et 14h30
	Heure de fin de validation Entre 8h30 et 15h

De nouveau, les résultats seront présentés dans la partie suivante.

9.6.3 Résultats de la simulation

Basé sur le scénario 2, ce troisième y rajoute la validation et porte l'objectif de déterminer quelle configuration il faut adopter pour mettre en adéquation validation et préparation. Rappelons que notre équipe de valideurs ne compte dans ce cas que 15 membres, où nous ne différencierons pas valideur senior et junior, les temps relevés dans le rapport national Securimed [DHOS 09] n'étant pas impactés suivant l'expérience du valideur.

Nous ne simulerons ici que deux parts d'UF en nominatif, la 10% et la 34%. La première permet de fournir une première analyse de l'impact de la validation, tandis que la seconde a pour objectif de voir si la DJINA est compatible avec cette équipe de valideurs.

De même, nous ferons varier deux fois les pourcentages liés aux niveaux d'analyses, le premier cas étant 0,12/0,7/0,18 et le second 0,12/0,63/0,25, la fourchette 0,18/0,25 représentant la part des prescriptions complètement analysées chaque jour. Le Tableau 9.17 récapitule les différentes possibilités.

Tableau 9.17 : Récapitulation des différentes situations

	Part d'UF en nominatif	%Niveau 1	%Niveau 2	%Niveau 3
Situation 1	10%	12	70	18
Situation 2	34%	12	70	18
Situation 3	10%	12	63	25
Situation 4	34%	12	63	25

Enfin, comme déjà énoncé, nous n'aborderons pas ici la partie préparation par les préparateurs.

9.6.3.1 Situation 1

Lors de ce premier cas, nous allons simuler les deux parts d'UF précédemment définies. Bien entendu, la répartition des valideurs sera changée afin de tester différentes configurations. Nous constaterons au fur et à mesure de l'analyse des résultats que nous retrouverons trois configurations.

↳ Configuration 1 : 15 valideurs entre 8h et 10h

Lors de cette première situation, tous les valideurs vont travailler sur la première tranche de deux heures. Le Tableau 9.18 présente les résultats de cette simulation.

Tableau 9.18 : Résultats pour la Situation 1-Configuration 1

	Horaire	8h	10h
Nombre de prescriptions en attente de validation		153	0
Nombre de prescriptions validées		0	153
Nombre de prescriptions préparées		0	78

Cette première configuration permet de traiter toutes les prescriptions en attente de validation. Cependant, le robot de dispensation n'arrive pas à suivre la cadence de validation et par

conséquent, les prescriptions s’accumulent. Il est donc nécessaire de déterminer sur le premier créneau combien de valideurs sont nécessaires sur celui-ci.

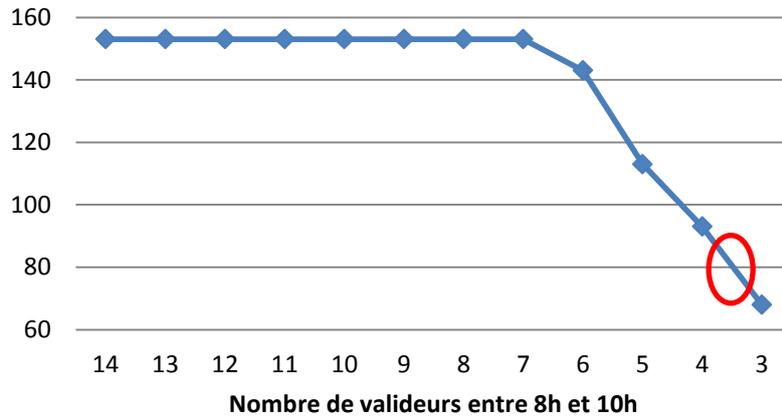


Figure 9.15 : Nombre de prescriptions validées suivant le nombre de valideurs à 10h

La Figure 9.15 montre la courbe représentant le nombre de prescriptions validées à 10h par une équipe de taille variable. Il est alors possible de déterminer la taille adéquate de l’équipe sur le premier créneau horaire. Etant donné que le robot ne peut traiter que 78 prescriptions environ sur ce créneau, l’équipe la plus adéquate sera composée de quatre valideurs. Nous ne choisissons pas une équipe de trois personnes afin d’éviter des périodes d’inactivité pour le robot.

Dès lors, nous pouvons tester la configuration 2.

↳ Configuration 2 : 4 valideurs entre 8h/10h et 11 entre 10h/12h

Le Tableau 9.19 présente les résultats de cette simulation.

Tableau 9.19 : Résultats pour la Situation 1-Configuration 2

Horaire	8h	10h	12h
Nombre de prescriptions en attente de validation	153	60	0
Nombre de prescriptions validées	0	93	153
Nombre de prescriptions préparées	0	77	153

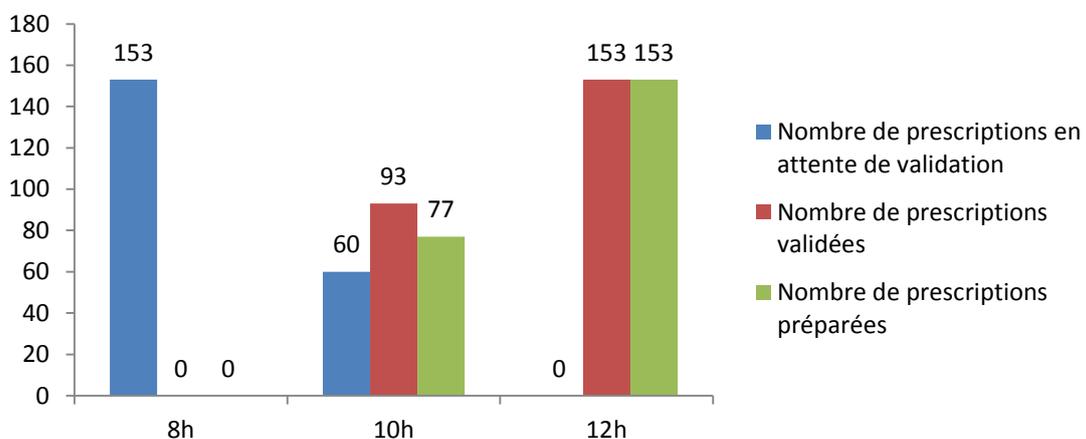


Figure 9.16 : Résultats pour la Situation 1-Configuration 2

Cette seconde configuration permet de constater que toutes les prescriptions sont prêtes à 12h (Figure 9.16). Cependant, une équipe de 11 personnes pour valider entre 10h et 12h est bien entendu trop conséquente. Ainsi, en testant de nouveau avec un nombre décroissant de valideurs, nous observons que pour le créneau 10h/12h, une équipe de trois personnes suffit.

Au final pour la situation 1, seuls sept valideurs seraient requis avec quatre d'entre eux validant entre 8h et 10h, et trois d'entre eux entre 10h et 12h.

9.6.3.2 Situation 2

Cette fois-ci, le nombre de prescriptions générées va être beaucoup plus conséquent. De plus, la Situation 1 nous a permis de voir que grouper tous les préparateurs entre 8h et 10h n'était pas pertinent. Nous partons donc de la configuration 2 pour cette situation.

↳ Configuration 2 : 4 valideurs entre 8h/10h et 11 entre 10h/12h

Le Tableau 9.20 présente les résultats de cette simulation.

Tableau 9.20 : Résultats pour la Situation 2-Configuration 2

Horaire	8h	10h	12h
Nombre de prescriptions en attente de validation	428	329	57
Nombre de prescriptions validées	0	99	371
Nombre de prescriptions préparées	0	75	230

Tout comme pour la Situation 1-Configuration 1, le robot n'arrive pas à suivre la cadence de validation mais cette fois-ci sur le créneau 10h – 12h (Figure 9.17). Une nouvelle fois, en testant un nombre décroissant de valideurs, il se trouve que le nombre adéquat, pour alimenter le robot en prescription validée, est de six. Nous obtenons par conséquent une troisième configuration à tester.

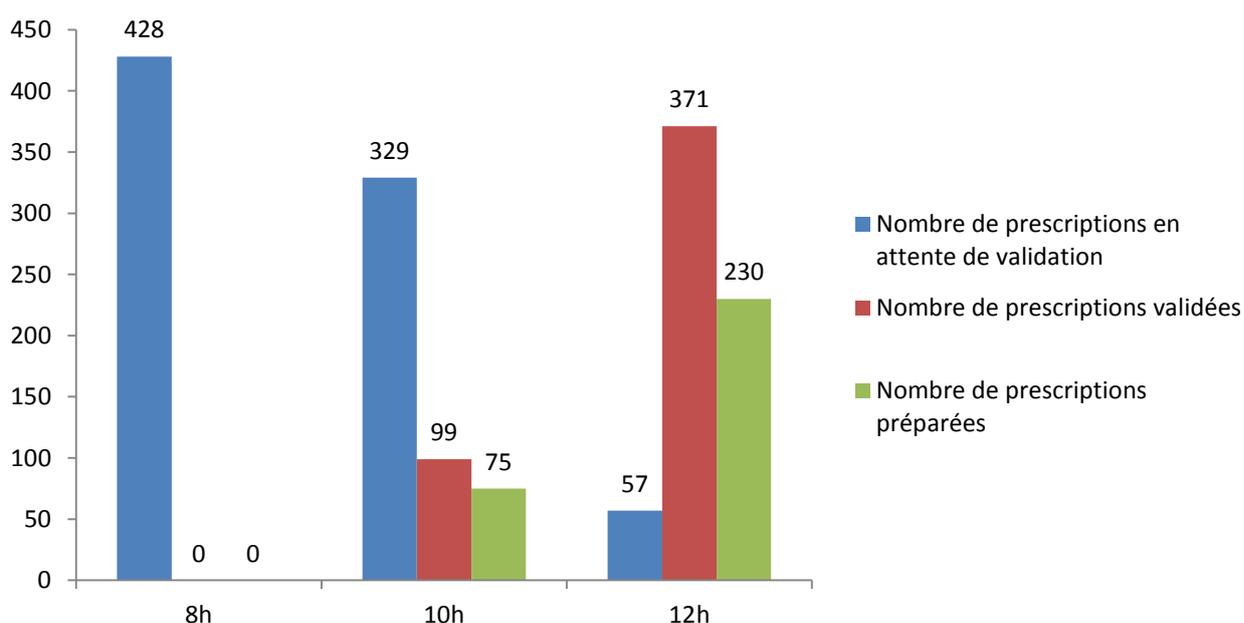


Figure 9.17 : Résultats pour la Situation 2-Configuration 2

↳ Configuration 3 : 4 valideurs entre 8h/10h, 6 entre 10h/12h et 5 entre 12h/14h

Le Tableau 9.21 présente les résultats de cette simulation.

Tableau 9.21 : Résultats pour la Situation 2-Configuration 3

	Horaire	8h	10h	12h	14h
Nombre de prescriptions en attente de validation		428	329	176	50
Nombre de prescriptions validées		0	99	252	378
Nombre de prescriptions préparées		0	75	230	375

Cette dernière configuration permet de mettre en lumière deux résultats. Le premier est que la répartition des valideurs sur les deux premiers créneaux est en adéquation avec la cadence du robot (Figure 9.18), permettant ainsi de ne pas monopoliser un pharmacien pour rien. La seconde constatation est que malheureusement une équipe de 15 valideurs ne suffit pas à traiter toutes les prescriptions, étant donné que 50 prescriptions sont restantes à 14h (Figure 9.18).

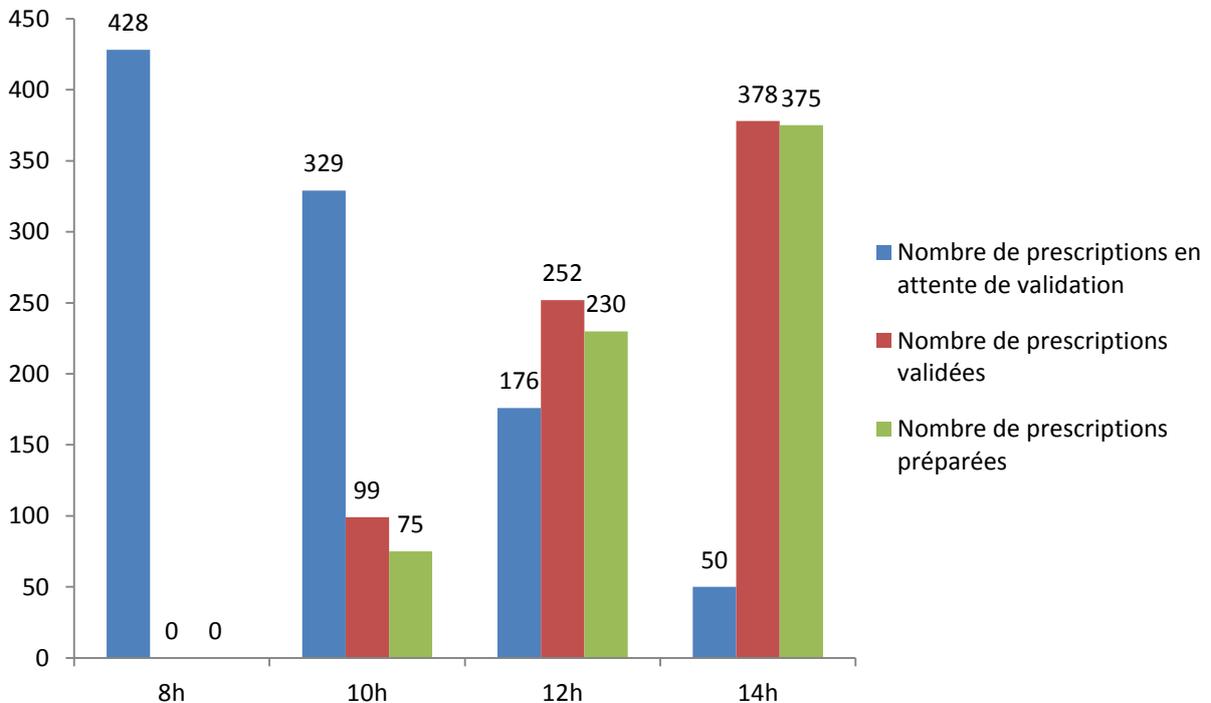


Figure 9.18 : Résultats pour la Situation 2-Configuration 3

Ainsi, avec une équipe de cette taille, seulement 378 prescriptions peuvent être validées soit 34% des lits pouvant être délivrés nominativement.

Passons maintenant aux deux dernières situations. Connaissant déjà les tailles des équipes adéquates, nous allons soumettre celles-ci à la validation de prescriptions dont le niveau 3 a été augmenté de 7%. Cela aura-t-il un impact sur le nombre de prescriptions validées ?

9.6.3.3 Situation 3

Testons d'entrée la configuration avec sept valideurs (Tableau 9.22).

Tableau 9.22 : Résultats pour la Situation 3

Horaire	8h	10h	12h
Nombre de prescriptions en attente de validation	153	66	0
Nombre de prescriptions validées	0	87	153
Nombre de prescriptions préparées	0	76	153

Il y a donc bien une influence sur le nombre de prescriptions validées mais uniquement à 10h (Figure 9.19), et cette influence reste très faible (augmentation de 10% des prescriptions non validées). L'impact pour une part de 10% des UF en nominatif ne sera donc pas pertinent.

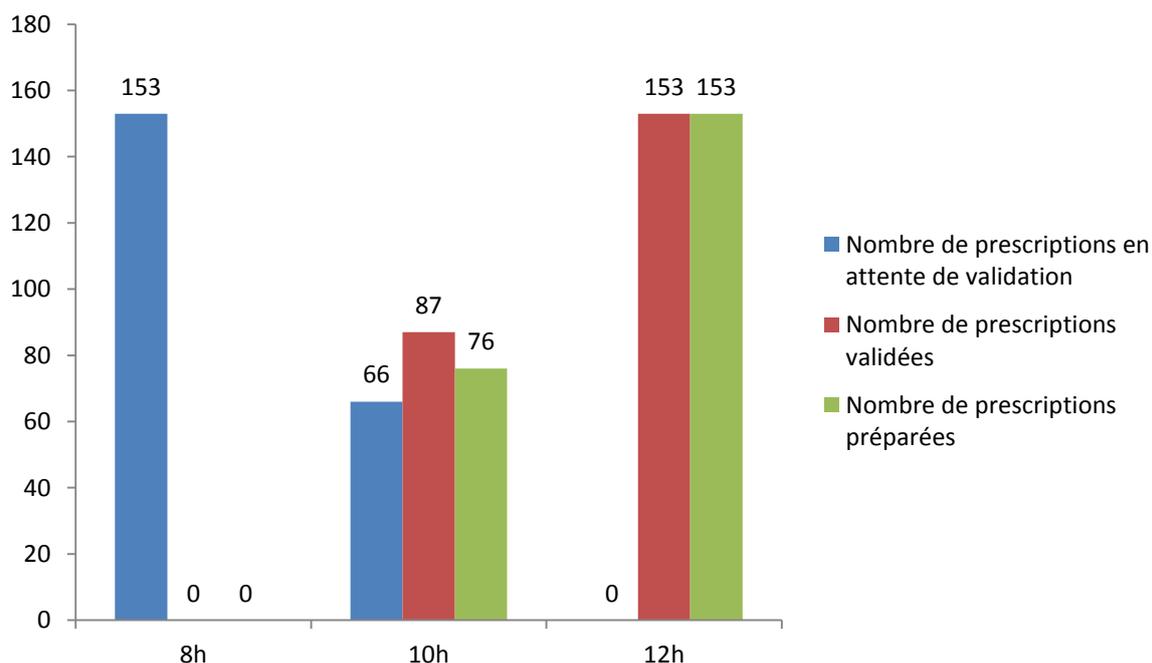


Figure 9.19 : Résultats pour la Situation 3

Voyons s'il en est de même pour la part de 34%.

9.6.3.4 Situation 4

Prenons d'emblée la configuration 3 pour réaliser ce test. Les résultats sont dans le Tableau 9.23.

Tableau 9.23 : Résultats pour la Situation 4 – Configuration 3

Horaire	8h	10h	12h	14h
Nombre de prescriptions en attente de validation	428	332	182	62
Nombre de prescriptions validées	0	96	246	366
Nombre de prescriptions préparées	0	76	230	364

La très faible augmentation de la part des prescriptions de Niveau 3, ne se répercute finalement que très peu sur la capacité de validation des valideurs. En effet, comparé aux données du

Tableau 9.21, à 10h l'écart est de 1%, à 12h de 3,3% et enfin à 14h de 20% (équivalent à 12 prescriptions de retard) (Figure 9.20).

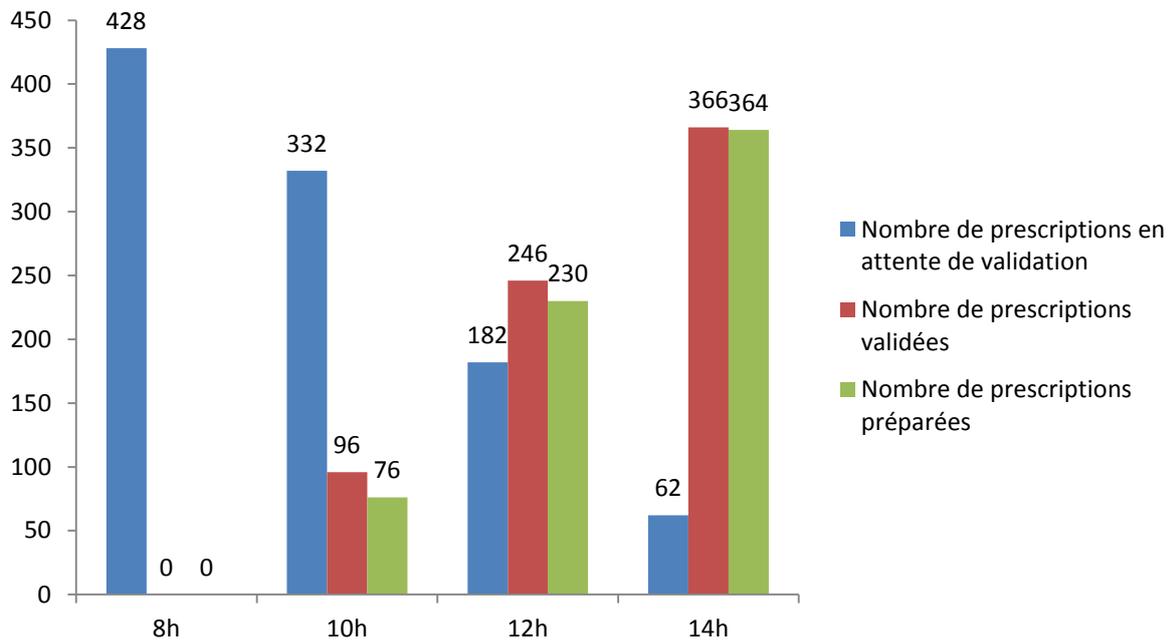


Figure 9.20 : Résultats pour la Situation 4 – Configuration 3

Les conclusions restent cependant les mêmes que pour la Situation 2 – Configuration 3, c'est-à-dire que la taille de l'équipe en place est trop faible mais que la répartition des valideurs permet d'avoir le bon nombre de prescriptions en attente devant le robot.

9.6.4 Conclusion

Le choix de la taille d'équipe et de la répartition des valideurs sur une journée sont donc bien deux facteurs déterminants dans la capacité de validation d'une PUI. La taille de l'équipe sera dépendante du temps que le valideur consacre à cette tâche tandis que la répartition doit considérer des éléments extérieurs tels que les heures de contre-visites et d'arrivée des nouveaux patients.

L'impact du niveau d'analyse des prescriptions se révèle finalement assez faible même si son influence variera beaucoup suivant le type de service et le niveau de connaissance du valideur.

Une solution permettant de contourner la petite taille d'une équipe serait de solliciter un valideur deux fois par jour, ce qui rendrait possible des roulements mais demanderait aux différents valideurs d'interrompre leur travail quotidien plusieurs fois, perturbant ainsi l'efficacité et la sûreté de ces tâches.

9.7 CONCLUSION

Nous avons dans un premier temps évalué la généricité de nos outils par le biais des critères de [FLETCHER 09a]. Bien que couvrant tous les aspects liés à la conception d'un modèle, les réponses apportées peuvent être sujettes à discussion et faire varier quelque peu, suivant son interprétation, les conclusions qui en découlent. Ainsi, même si nos outils s'orientent vers un niveau

3B, la flexibilité dans l'interprétation de certaines réponses peut faire pencher ce niveau vers un 3A, laissant par conséquent la discussion ouverte sur ce sujet.

La création et l'analyse des scénarios a permis dans un second temps de mettre en lumière l'utilisation de notre modèle générique des processus et de notre diagramme générique des classes. Par la suite, l'extraction et la mise en forme des données nécessaires à la simulation a mis en exergue la relative complexité d'obtention de ces données, mais aussi la difficulté liée à leur mise en forme qui est plus que cruciale pour utiliser correctement notre modèle de simulation. C'est pourquoi, nous avons développé des interfaces permettant d'aider à identifier les données essentielles à intégrer dans le modèle.

Une fois les modifications connues et les données traitées, chaque scénario a pu être simulé, testant ainsi des situations inédites pour le CHU de Clermont-Ferrand, scénario grâce auxquelles certaines conclusions structurelles ou organisationnelles ont pu être tirées.

Pour le premier scénario, l'impact majeur de la DJINM a été comme attendu une surcharge de travail pour les préparateurs. Dans notre cas, une part de 10% des UF en DJINM se résumerait à une augmentation de la charge de travail des préparateurs de 10% par jour. Par conséquent, il sera donc difficile pour ce CHU de mettre cette dispensation en place sans revoir la taille de son équipe ou sans ajuster la répartition des UF sur les préparateurs, dont la multidisciplinarité ne pourra se révéler qu'utile pour absorber de manière flexible les demandes à préparer chaque jour.

Le second scénario a surtout été l'occasion de montrer une fois de plus l'impact et les performances d'un robot de dispensation pour préparer des prescriptions (dans le cadre de la DJINA). Même avec un robot possédant une capacité de production plutôt limitée, nous avons pu voir que ce dernier serait en mesure de couvrir entre 34% et 45% des préparations nominatives à réaliser chaque jour suivant sa durée de fonctionnement (respectivement six et sept heures), sans perdre pour autant d'efficacité au niveau du travail des préparateurs. Ces chiffres restent cependant à nuancer, le type de l'UF n'étant pas prise en compte ici, alors que ce dernier peut influencer la taille de la prescription (globalement les prescriptions générées étaient plus proches d'un format MCO que Soins long durée/psychiatrie).

Enfin lors du troisième et dernier scénario, la validation des prescriptions avant préparation a été mise en pratique, afin d'en étudier la faisabilité au CHU. La conclusion majeure qui en découle est que la validation pharmaceutique demande une équipe de pharmaciens conséquente et un temps de disponibilité qui l'est tout autant. Dans le cas du CHU, il a été mis en évidence qu'une équipe composée de 15 pharmaciens (sénior ou junior) ne se révélerait pas suffisante pour prendre en charge les prescriptions arrivant de 34% des UF (pourcentage maximum issu de la DJINA), à moins que ces derniers ne fassent de la validation sur plusieurs créneaux horaires, ce qui aurait malheureusement pour conséquence de non seulement grever le temps de travail dévoué à la routine mais aussi d'en diminuer la sûreté du fait des diverses sollicitations extérieures. Les autres solutions seraient de solliciter d'autres pharmaciens, ou bien de dédier certains d'entre eux à cette activité. Ces solutions ne sont cependant pas sans défaut, la première augmentant encore le nombre d'interlocuteurs pour les services, tandis que la seconde pourrait se révéler rébarbative si seule cette tâche était réalisée durant les journées de travail.

A noter qu'une équipe de 16 valideurs répartie en 4/6/6 sur les tranches horaires 8h – 10h/ 10h – 12h/ 12h – 14h permettrait tout juste de prendre en charge les 34% de prescriptions sur les mêmes horaires – de 8h à 14h.

CONCLUSION GÉNÉRALE

Ayant pour objectif de concevoir un outil d'aide à la décision à destination du CHU de Clermont-Ferrand, nous tentons par le biais de ces travaux de résoudre les problématiques actuelles liées à la prise de décision pour les décisionnaires hospitaliers. En effet, impossibilité physique de mener des tests, restriction économique, de personnel, etc. sont autant de contraintes ne permettant pas à ces acteurs de pouvoir se projeter sereinement dans le futur, et c'est pourquoi nous devons avec cet outil fournir des réponses à ces personnes.

Travaillant tout particulièrement sur le *Circuit Du Médicament (CDM)*, nous avons constaté que peu d'outils et de travaux s'étaient axés sur la résolution de ces problématiques, et en conséquence, nous avons à travers ces travaux cherché à développer non seulement l'outil désiré mais aussi à intégrer celui-ci dans une démarche plus globale et structurée permettant d'accompagner les acteurs tout au long de leur processus de décision.

Lors du **Chapitre 1**, nous présentons le CDM et abordons les nombreuses problématiques qui lui sont liées. Le CDM s'est alors révélé être un flux complexe incorporant de nombreux acteurs et de nombreuses entités. S'articulant autour de trois étapes majeures – prescription, dispensation, administration – nous détaillons dans ce chapitre, les différents types de dispensation existants, qui définissent et régissent la structure organisationnelle du CDM, chaque dispensation utilisant des méthodes diverses et variées (préparation manuelle, robotisée, délivrance individuelle ou nominative, etc.). Par la suite, nous exposons les problèmes auxquels est soumis le CDM, en se penchant tout particulièrement sur les *Erreurs Médicamenteuses (EM)*, fléau moderne, en un sens, du monde hospitalier. En détaillant les nombreuses causes liées à ces erreurs, nous en venons à voir quelles sont les solutions préconisées et reconnues efficaces pour lutter contre cette problématique et ainsi sécuriser le CDM. Dès lors, il apparaît que la *dispensation journalière individuelle nominative (DJIN)*, la robotisation et l'informatisation de ce circuit constituent les trois piliers de la lutte contre les EM, mais à ces dernières peuvent se greffer des solutions connexes telles que les doses unitaires ou l'intégration de pharmaciens dans les services, solutions de plus en plus usitées.

La question est par conséquent de savoir comment à partir d'un système réel – la pharmacie hospitalière – une personne pourrait être en mesure d'identifier et d'évaluer la pertinence de ces améliorations sur son système.

Le **Chapitre 2** a pour objectif d'apporter une réponse à cette question. Dans un premier temps, nous nous intéressons à la problématique de savoir comment caractériser le CDM, et quel « outil » peut permettre d'en discuter et de le manipuler avec aisance. A la lecture de nombreux rapports et textes ministériels, il apparaît que la notion de processus semble la plus usitée et la plus adéquate pour représenter le CDM, mais faut-il encore en comprendre l'essence et savoir comment le construire. Pour cela, nous détaillons cette notion en abordant différents thèmes qui leur sont liés – granularité, définition, classes, etc. – puis nous listons et analysons différents langages, méthodes et méthodologies qui permettent de construire les processus et modéliser un système. Parmi ces langages, méthodes et méthodologies, issus pour la plupart du monde industriel, nos choix se sont portés sur le langage BPMN, et sur la méthodologie ASCI. En effet, tandis que le premier a été sélectionné avec l'aide des pharmaciens – évaluation de divers langages – le second a déjà prouvé à

maintes reprises sa bonne intégration et application au monde hospitalier [RODIER 10], [HUET 11]. Complété par la simulation, apportant la dernière pierre à cet édifice, ces choix forment un trio interdépendant qui se complète, étant donné que BPMN permet de modéliser les processus, qu'ASCI structure la modélisation et l'étude du système et enfin que la simulation rend possible la visualisation du comportement de ce dernier. Mais pourquoi choisir la simulation ?

Ce chapitre apporte des réponses à cette question en mettant en avant les nombreux avantages de la simulation, sans pour autant en oublier les inconvénients. De plus, nous abordons des notions telles que les différentes classes de simulation et les niveaux d'abstractions, éléments importants si la simulation vient à être sollicitée. Enfin, nous nous penchons sur un des piliers de notre travail, à savoir la généricité, en présentant en quoi elle consiste, comment y parvenir et comment l'évaluer, ce dernier point utilisant les nombreux critères définis par [FLETCHER 09a].

Dans le **Chapitre 3**, nous restons sur cette thématique de la simulation, en montrant que celle-ci a déjà trouvée sa place dans le monde hospitalier et qu'elle continue même à y être de plus en plus sollicitée. Notamment utilisée dans les pays anglophones [LENT 12], [BRAILSFORD 11], nous constatons que les travaux dans lesquels elle est intégrée, portent en grande majorité sur l'étude des urgences hospitalières et sur les blocs opératoires, deux services où la gestion des flux est un problème épineux pour les structures hospitalières. De surcroît, il apparaît clairement à la vue des publications dans ce domaine que le service de la pharmacie est malheureusement peu étudié, constat encore plus dur quand on considère le CDM dans son entièreté, où les travaux sont rares. En effet, la plupart des travaux sur les pharmacies hospitalières concerne le domaine de la rétrocession des médicaments aux patients, et ne concerne donc pas la pharmacie. Pour autant, certains travaux récents considèrent le CDM plus globalement, tels que les travaux de [WONG 03] et [HUET 11]. Au final, les travaux réalisés dans le cadre de cette thèse permettent d'apporter des éléments pour étudier le CDM, qui en manque cruellement, et de fournir éventuellement une base de travail pour de futures recherches.

Le **Chapitre 4** a pour objectif de présenter deux des trois outils – BPMN et ASCI – que nous sollicitons dans le cadre de nos travaux. Ainsi, nous commençons par détailler précisément comment la méthodologie ASCI est construite, quel est son raisonnement et comment elle s'utilise. Raisonnant étape par étape – qui sont au nombre de quatre – nous poursuivons ce chapitre en présentant les « arcanes » du langage BPMN. Composé de très nombreux éléments et de trois types de diagramme – seul le diagramme de collaboration nous a intéressés ici – nous nous concentrons sur la présentation des éléments que nous étions le plus à même d'utiliser pour notre modélisation. Evènement, porte et activité sont donc passés au crible afin de mettre en lumière leur fonctionnement et utilisation, mais aussi et surtout afin de souligner le potentiel de ce langage qui se révèle flexible et simple d'appréhension dans le cadre d'une utilisation courante. La fin de ce chapitre a pour but de vérifier cette affirmation. Ainsi, en utilisant les travaux de [RECKER 08] et [ZUR MUEHLEN 08], la conclusion principale concernant ce langage est bien que celui-ci est simple dans son utilisation mais qu'il reste malheureusement sous-utilisé et donc que son potentiel reste inexploité dans le monde industriel. Ce bilan est bien entendu le même dans le monde hospitalier, où les quelques rares cas de sollicitation de BPMN sont limités, les travaux de [ROJO 08] faisant exception, les auteurs ayant ici travaillé à la fois à une modélisation macroscopique et microscopique du traitement chirurgical d'une pathologie.

Avant de commencer à concevoir nos processus, un constat nous a frappés. La méthodologie ASCI est parfaite pour le monde de l'industrie, ou tout du moins pour les ingénieurs, mais il se trouve que le public ici visé n'est ni familier, ni spécialiste du domaine. Il est donc nécessaire de « repenser » la méthodologie ASCI, non pas dans son contenu mais dans sa forme, ce qui est l'objet de ce **Chapitre 5**. Ici, nous nous attelons à « repenser », « reconstruire » cette méthodologie, afin d'en adapter le contenu et la forme aux praticiens hospitaliers et plus particulièrement aux pharmaciens. Passant par une simplification et une recomposition des étapes de base, notre nouvelle démarche s'appuie finalement elle aussi sur quatre phases, qu'il a été possible de mettre en miroir avec ASCI, pour en souligner les similitudes. La méthodologie étant désormais utilisable par un pharmacien, il est temps de concevoir les outils jalonnant l'utilisation de notre démarche.

Le **Chapitre 6** est dédié à décrire comment nous parvenons à construire le modèle générique de connaissance du domaine, le CDM, modèle composé du modèle générique des processus sous BPMN et d'un diagramme générique des classes UML venant le compléter. Pour le modèle BPMN, il est nécessaire d'identifier avec précision les acteurs, les événements et les choix faits tout du long des processus du CDM, afin de les convertir dans le langage choisi. Ces éléments identifiés, nous sommes alors capables de concevoir une ébauche du modèle final, que nous complétons par la suite avec les nombreuses actions et activités réalisées par les acteurs dans le CDM, pour ainsi obtenir le modèle générique des processus. La dernière partie de ce chapitre est dédiée à la conception du diagramme générique des classes UML, pour lequel nous avons grandement réutilisé les travaux de [\[HUET 11\]](#), diagramme bien entendu remis à jour et complété suivant nos besoins.

A ce niveau de nos travaux, nous sommes alors en possession des deux premiers outils nécessaires à l'application de notre démarche. Par conséquent, il ne reste que l'un d'entre eux, et pas des moindres, à concevoir : le modèle de simulation.

Le **Chapitre 7** amorce cette démarche, cependant avant de pouvoir détailler la construction du modèle, il est nécessaire de présenter le logiciel utilisé pour cette conception. Ce logiciel, SIMIO, choisi en grande partie pour ses capacités graphiques et sa ressemblance avec ARENA, est un outil flexible et puissant mais dont la prise en main peut être complexe afin d'en tirer tout son potentiel. En commençant par une brève discussion sur les leviers et critères de performance liés à notre modèle, nous poursuivons ce chapitre par une présentation des interfaces du logiciel et de leur rôle. Par la suite, nous détaillons les objets composant SIMIO et cela par classe. Ainsi les objets « Fixe » (Serveur, Source, etc.), les « Agent » (Travailleur, Véhicule, etc.), les « Lien » et les « Nœud » sont présentés et détaillés tour à tour, avant de donner un court exemple de création d'un modèle sous SIMIO.

Ce chapitre conclut sur la création d'un certain nombre de règles facilitant le passage de BPMN à SIMIO. En effet, ces deux outils étant en lien direct, le modèle BPMN ayant été utilisé pour la construction du modèle SIMIO, nous avons proposé des règles permettant de s'attaquer au problème de traduction des éléments du premier vers le second. Touchant aux activités, aux acteurs, et aux événements, ces règles ne restent cependant qu'une interprétation personnelle de la traduction du langage BPMN en « langage » SIMIO et pourrait donc être approfondi en vue d'automatiser et de faciliter encore plus cette conversion.

Nous pouvons désormais présenter comment le modèle de simulation a été construit, ce qui est l'objet du **Chapitre 8**. En partant d'une vue aérienne du modèle, nous avons choisi de

décomposer la présentation en deux parties. En effet, après avoir détaillé les éléments sources de notre modèle, nous suivons dans un premier temps les éléments liés à la préparation des demandes nominatives avant de faire de même pour les éléments liés à la préparation des demandes globales. Enfin, nous terminons par le détail du transport et de l'envoi au service de ces demandes. Ainsi, pour chaque élément du modèle, nous présentons les propriétés de celui-ci, les données utiles et nécessaires à son exécution et enfin les « processus » SIMIO régissant son comportement.

Finalement, nous concluons ce chapitre par la présentation du mécanisme de tables de données que nous proposons, mécanisme permettant de résoudre le problème de la généricité, mais aussi de faire en sorte que les pharmaciens n'aient pas à entrer dans la complexité du code du modèle SIMIO afin de tester leurs différentes hypothèses.

Tous les éléments nécessaires à l'utilisation de notre démarche par un pharmacien étant opérationnels, il est temps de mettre en application celle-ci sur des problèmes concrets. Dans cet ultime **Chapitre 9**, nous détaillons donc trois scénarios pensés autour de problématiques actuellement soulevées au CHU de Clermont-Ferrand, et qui sont toutes en lien direct avec la dispensation journalière individuelle. Ainsi, tandis que dans le premier nous regardons la faisabilité de cette DJIN de manière manuelle, dans le second, nous observons la même problématique mais cette fois-ci en utilisant une préparation automatisée des demandes nominatives. Enfin, le troisième et dernier scénario concerne un aspect spécifique de cette préparation nominative à savoir la validation pharmaceutique des prescriptions.

Chacun de ces scénarios a donc permis de tirer des conclusions pour le CHU, mais aussi d'observer les divers outils en application. Pour chacun d'entre eux nous présentons comment le modèle générique des processus et le diagramme générique des classes sont sollicités avant de détailler les nombreuses données qu'il est nécessaire de fournir au modèle de simulation afin d'en tirer des résultats. Le premier scénario ici présenté, nous apprend, sans trop de surprise, que la DJINM sera compliquée à mettre en œuvre au CHU sans réajuster la taille de l'équipe préparateur. Le second nous montre que la DJINA est au contraire possible au CHU, ne demandant que peu de modifications sur les habitudes de travail, mais nécessitant un fort investissement économique. Enfin, le dernier met en évidence le fait que la validation des prescriptions pour un établissement de la taille du CHU de Clermont-Ferrand, demandera énormément de temps pharmacien et/ou des équipes dédiées à cette activité.

BILAN

Cette thèse a de par son application permis de constater, sans trop de surprise, que le problème majeur rencontré par les praticiens hospitaliers, et notamment les pharmaciens dans notre cas, était le manque de lisibilité quant aux décisions qu'ils doivent prendre et aux conséquences (organisationnelles, humaines, économiques, etc.) que celles-ci impliqueraient. En se concentrant tout particulièrement sur le *Circuit Du Médicament* dans ces travaux, nous en venons à la conclusion que faire évoluer celui-ci afin qu'il se retrouve en adéquation avec son environnement, en perpétuel changement, ne pourra se faire sans contraintes et imposera aux praticiens et personnels

hospitaliers de **disposer de méthodes leur permettant d'anticiper et de mettre en place de telles évolutions.**

Ainsi, de par l'analyse des solutions existantes et de par notre expérience, la *méthode d'anticipation* permettant de prendre en compte un maximum de situations, tout en minimisant les coûts annexes, se trouve être **la simulation informatique**, outil désormais largement utilisé dans le monde hospitalier. L'objectif étant à terme d'obtenir un modèle générique de connaissances dans lequel chaque hôpital pourra se retrouver, ce modèle devait de plus aider à la sécurisation du CDM, en identifiant plus précisément les points sensibles et en permettant de mettre en lumière quelles seront les modifications nécessaires au niveau organisationnel et structurel.

L'obtention de ce modèle de simulation a néanmoins nécessité en amont d'analyser le système et d'en comprendre le fonctionnement. L'utilisation de la **methodologie ASCI**, et son adaptation, a donc permis d'encadrer cette démarche en structurant notre étude et en mettant en lumière les éléments nécessaires à l'aboutissement de nos travaux. De ce fait, la sollicitation du **langage BPMN** pour présenter et modéliser les processus du CDM, donnera, nous l'espérons, envie aux pharmaciens de continuer à utiliser ce langage dans leur démarche d'amélioration, mais aussi, pourquoi pas, dans leur discussion quotidienne.

A ce niveau, nous tenons à souligner que les conclusions de [**FLETCHER 09a**] sont en accord avec nos méthodes de travail, à savoir que la conception (design) d'un modèle de simulation se fait souvent par un mapping des processus en amont (modèle BPMN pour nous), que l'acquisition des données est réalisée par récupération des données des systèmes informatiques (ici Pharma©), des relevés terrains (CHU de Clermont-Ferrand, données nationales), et issues de discussions avec les personnes compétentes (Pharmacien, Interne).

Le modèle de simulation a ainsi été développé à partir du modèle générique de connaissance et se veut à la fois utilisable à un niveau local tout en étant capable d'apporter des conclusions générales. En effet, les nombreuses options de paramétrages existantes dans ce modèle, permettent d'intégrer en tant que données d'entrées, des spécificités locales et donc de les simuler. Par conséquent, le degré de précision des données d'entrées déterminera le type de problème étudié (local ou global). Cette flexibilité dans le format des données d'entrées, nous a permis de contourner des difficultés liées à certains problèmes locaux trop détaillés pour être exploités par un modèle générique.

Au final, ce modèle étant testé au niveau local avec les données du CHU de Clermont-Ferrand, il a été possible de simuler des situations, configurations inexistantes dans cet hôpital grâce à des données nationales obtenues par le biais de rapports et de sites web officiels, et ainsi apporter certains éclaircissements au CHU.

Pour terminer, si une structure hospitalière quelconque venait à solliciter nos outils et notre démarche afin de l'appliquer sur sa propre pharmacie, nous pouvons estimer qu'une mise en œuvre de nos travaux serait extrêmement rapide. En effet, en considérant le temps de récolte de données, quatre jours seraient amplement suffisants pour appliquer notre démarche sur ladite structure. Ainsi, la première journée étant consacrée à la définition du cadre des évolutions et du bilan de la situation actuelle, l'utilisation de nos deux premiers outils ne prendrait pas plus d'une matinée lors de la seconde journée, laissant deux journées et demie pour récolter les données et faire les simulations. Tout en sachant que la récolte des données sera ce qui nécessitera le plus temps, les simulations pouvant prendre un après-midi tout au plus suivant le degré de simulation souhaité et la période à simuler.

PERSPECTIVES

Ce travail de longue haleine n'en est pas pour autant terminé et de nombreuses perspectives autant d'un point de vue technique qu'applicatif existent.

D'un point de vue général

Tout d'abord, et comme nous l'avons déjà énoncé, l'application de notre démarche sur de nouvelles structures hospitalières serait plus que recommandée afin de rendre nos différents outils encore plus robustes et de construire une bibliothèque de résultats permettant à des hôpitaux de voir ce qui se fait et ce qui pourrait être fait.

Par conséquent, une large diffusion de ce qui peut se faire, permettrait à de nouvelles structures de s'intéresser et de justifier des investissements dans de nouvelles organisations et solutions techniques, participant ainsi à une mutualisation des moyens mis en œuvre dans le monde hospitalier français pour s'améliorer et se sécuriser.

Au niveau du modèle générique de connaissance du CDM

En plus des évolutions techniques liées au langage BPMN (nouvelles normes, améliorations), les processus représentés pourraient être amenés à évoluer ou à s'enrichir. Cependant toute démarche dans ce sens devrait prendre en compte et vérifier la cohérence de nos trois outils, tout nouvel élément devant apparaître, ou être prise en compte, sur les trois.

Parmi les grosses évolutions que BPMN pourrait subir, il y aurait l'intégration des processus liés au transport et à la logistique accompagnant le CDM. Bien qu'il serait possible de pousser assez loin le détail de ces processus, il ne faudra pas oublier que cet élément est représenté de manière assez simple sous SIMIO.

Au niveau du modèle de simulation et des interfaces

Il s'agit là certainement du point qui subira prochainement, ou dans un futur plus lointain, le plus de changements. Pour le modèle de simulation, il est bien sûr toujours possible d'intégrer de nouveaux éléments, mais nous ne recommanderions que peu cette démarche, la cohérence globale du modèle pouvant alors être mise à mal. Cependant, un gros travail graphique reste à faire sur celui-ci afin de le rendre plus lisible pour les pharmaciens et décisionnaires. On pensera notamment à l'intégration de murs, de salles et l'ajout d'animations pour certains acteurs.

Pour les interfaces, quelques améliorations sont à prévoir. Tout d'abord, comme pour le modèle de simulation, un travail graphique serait à mener afin de rendre plus agréable la navigation dans les différents menus et dans l'exploitation des résultats obtenus par la simulation. Il serait aussi possible de retravailler la saisie des informations, mais nous tenons à souligner que les tables Excel étant liées à celles de SIMIO, il serait impératif dans ce cas de retravailler le code VBA qu'utilisent nos interfaces.

Pour le CHU

En prolongation de ces travaux de recherche, durant les quelques semaines restantes sur le contrat avec le CHU, nous allons nous atteler à la rédaction d'un manuel d'utilisation détaillé du logiciel que nous avons réalisé. Ce manuel permettra ainsi aux futurs utilisateurs de ne pas prendre peur à la vue des nombreuses données à renseigner dans l'interface et leur permettra d'utiliser intuitivement notre modèle.

Dans le même temps, nous allons utiliser nos travaux pour étudier une pharmacie entièrement nouvelle pour le CHU. Le but est d'intégrer dans une structure fictive des solutions inexistantes ou inenvisageables pour le moment dans le CHU. Ce véritable laboratoire d'idées pourra alors fournir aux pharmaciens des pistes pour leurs choix futurs.

BIBLIOGRAPHIE

- [ANAES 03]** Agence Nationale d'Accréditation et d'Evaluation en Santé (ANAES). *Principes méthodologiques pour la gestion des risques en établissement de santé*. Rapport, Janvier 2003.
- [ANAP 12]** Majid Talla & Xavier Liu. *Sécuriser la prise en charge médicamenteuse du patient – La délivrance nominative des médicaments dans les établissements de santé*. Rapport ANAP, Décembre 2012.
- [AUGUSTO 09]** Vincent Augusto & Xiaolan Xie. *Redesigning pharmacy delivery processes of a health care complex*. Health Care Management Science, vol. 12, pages 166 – 178, 2009. Numéro DOI : 10.1007/s10729-008-9086-3.
- [BARNES 97]** C.D. Barnes & J.L. Quason. *Success stories in simulation health care*. Proceedings of the Winter Simulation Conference, pages 1280 – 1285, 1997.
- [BATES 95]** DW Bates, DJ Cullen, N Laird, LA Petersen et al. *Incidence of adverse drug events and potential adverse drug events, Implications for prevention, ADE Prevention Study Group*. Journal of the American Medical Association, vol. 5, no. 274, pages 29 – 34, Juillet 1995.
- [BATES 00]** DW Bates. *Using information technology to reduce rates of medication errors in hospitals*. British Medical Journal, no. 320, pages 788 – 791, Mars 2000.
- [BERNHEIM 05]** C. Bernheim, E. Schmitt, E. Dufay. *Iatrogénie médicamenteuse nosocomiale et gestion des risques d'erreur médicamenteuse : à propos de l'analyse des notifications du réseau REEM*. Oncologie, no. 7 (2), Springer, pages 104 – 119, Avril 2005.
- [BRAILSFORD 04]** S.C. Brailsford, V.A. Lattimer, P. Tarnaras é J.C. Turnbull. *Emergency and on-demand health care: modeling a large complex system*. Journal of the Operational research Society, no. 55, pages 34 – 42, 2004.
- [BRAILSFORD 09]** S.C. Brailsford, P.R. Harper, B. Patel & M. Pitt. *An analysis of the academic littérature on simulation and modelling in health care*. Journal of Simulation, no. 3, pages 130 – 140, 2009.
- [BRAILSFORD 11]** Sally Brailsford & Jan Vissers. *OR in healthcare: A European perspective*. European Journal of Operational Research, no. 212, pages 223 – 234, 2011.

- [BUJANOWSKA 11] Anna Bujanowska & Witold Bialy. *The identification and the modelling of processes of the use of the technical infrastructure in the hospital organization*. Management systems in Production Engineering, no.4 (4), pages 7 – 12, 2011.
- [CALVEZ 90] J.P. Calvez. *Spécification et conception des systèmes : une méthodologie mcse*. Numéro ISBN : 2-225-82107-0. Normandie Impr., 1990.
- [CASTOT 09] Anne Castot, Angélique Arnoux & Dorothee Durand. *Guichet erreurs médicamenteuses, Présentation et bilan depuis sa mise en place*. Agence française de sécurité sanitaire des produits de santé (AFSSAPS), Juin 2009.
- [CASTOT 10] Anne Castot, Angélique Arnoux & Dorothee Durand. *Le guichet erreur médicamenteuse, Bilan de l'activité 2009*. Agence française de sécurité sanitaire des produits de santé (AFSSAPS), Juillet 2010.
- [CAUTERMAN 07] Maxime Cauterman, Hervé Laborie & Sébastien Woynar. *Circuit du médicament, Approfondissement*. Rapport Intermédiaire, Mission national d'Expertise et d'Audit Hospitalier (MeaH), 2007.
- [CHAUVET 09] Julie Chauvet. *Une méthodologie de modélisation pour les systèmes hospitaliers : application sur le Nouvel Hôpital Estaing*. Thèse de doctorat, Université Blaise Pascal – Clermont II, 2009.
- [COCHRAN 06] Jeffrey K. Cochran & Aseem Bharti. *A multi-stage stochastic methodology for whole hospital bed planning under peak loading*. International Journal of Industrial and Systems Engineering, no. 1, pages 8 – 36, 2006.
- [COMBES 94] C. Combes. *Méthodologie de modélisation orientée objets des systèmes hospitaliers*. Thèse de doctorat, Université Blaise Pascal – Clermont II, 1994.
- [COOPER 02] K. cooper, R. Davies, P. Roderick, D. Chase, J. Raftery. *The Development of a Simulation Model of the Treatment of Coronary Heart Disease*. Health Care Management Science, no. 5, pages 259 – 267, 2002.
- [CUBAYNES 11] Marie-Hélène Cubaynes, Didier Noury, Muriel Dahan & Evelyne Falip. *Le circuit du médicament à l'hôpital*. Rapport, Inspection générale des Affaires Sociales (IGAS), Mai 2011.
- [DAHAN 10] Muriel Dahan & Jacques Sauret. *Sécurisation du circuit du médicament à l'Assistance Publique-Hôpitaux de Paris (AP-HP)*. Rapport, Inspection générale des Affaires Sociales (IGAS), Juillet 2010.
- [DEAN 97] Bryony Dean, Steve Gallivan, Nick Barber & Ann Van Ackere. *Mathematical modeling of pharmacy systems*. American Journal of Health-System Pharmacy, vol. 54 no. 21 pages 2491 – 2499, Novembre 1997.
- [DEAN 99] Bryony Dean, Ann van Ackere, Steve Gallivan & Nick Barber. *When should pharmacists visit their wards ? An application of simulation to planning hospital pharmacy services*. Health Care Management Science, vol. 2, pages 35 – 42, 1999.

- [DGOS 12] Direction Générale de l'Offre de Soins (DGOS). *Qualité de la prise en charge médicamenteuse, Outils pour les établissements de santé*. Février 2012.
- [DHOS 09] Direction de l'Hospitalisation et de l'Organisation des Soins. *Etude d'impact organisationnel et économique de la sécurisation du circuit du médicament dans les établissements de santé*. Rapport de l'étude, Octobre 2009.
- [DOUMEINGTS 84] G. Doumeingts. *Méthode GRAI : méthode de conception des systèmes productives*. Thèse de doctorat, Université de Bordeaux 1, 1984.
- [EL HAOUZI 08] H. El Haouzi, A. Thomas & J.F. Pétin. *Contribution to reusability and modularity of manufacturing systems simulation models: Application to distributed control simulation within DFT context*. International Journal of Production Economics, no. 112, pages 48 – 61, 2008. Numéro DOI : 10.1016/j.ijpe.2006.12.067
- [ENEIS 11] Direction de la Recherche, des Etudes, de l'Evaluation et des Statistiques. *Comparaison des deux études ENEIS 2004 et 2009*. Rapport final, Enquête National sur les Evènements Indésirables graves associés aux Soins (ENEIS), Septembre 2011.
- [FENIES 06] Pierre Fénies. *Une méthodologie de modélisation par processus multiples et incrémentiels: application pour l'évaluation des performances de la Supply Chain*. Thèse de doctorat, Université Blaise Pascal – Clermont II, 2006.
- [FETTER 65] R.B. Fetter, J.D. Thompson. *The simulation of hospital systems*. Operations Research, no. 13, pages 689 – 711, 1965.
- [FLETCHER 09a] Adrian Fletcher & Dave Worthington. *What is a "generic" hospital model? A comparison of "generic" and "specific" hospital models of emergency patient flows*. HealthCare Management Science, no. 12, pages 374 – 391, 2009. Numéro DOI: 10.1007/s10729-009-9108-9.
- [FLETCHER 09b] Adrian Fletcher & Dave Worthington. *Generic simulation modelling of A&E patient flows through the whole hospital*. Operations Research Applies to Health Services (ORAHs), Juillet 2009.
- [FONE 03] D. Fones, S. Hollinghurst, M. Temple, A. Round, N. Lester, A. Weightman et al. *Systematic review of the use and value of computer simulation modeling in population health and health care delivery*. J Public Health Med, no. 25, pages 325 – 335, 2003.
- [GLOUBERMAN 01] Sholom Glouberman & Henry Mintzberg. *Managing the care of health and the cure of disease, Part 1: Differentiation*. Janvier 1996.
- [GOUJON 97] J.Y. Goujon. *Un environnement de modélisation multi-agents pour la spécification et l'évaluation des performances des systèmes industriels de production*. Thèse de doctorat, Université Blaise Pascal – Clermont II, 1997.
- [GOURGAND 91] Michel Gourgand & Patric Kellert. *Conception d'un environnement de modélisation des systèmes de production*. In 3ème congrès international de génie industriel, Tour, France, 1991.

- [GOURGAND 92] Michel Gourgand & Patric Kellert. *An object-oriented methodology for manufacturing system modeling*. Summer Computer Simulation Conference, pages 1123 – 1128, Reno, Nevada, USA, Juillet 1992.
- [GOURGAND 07] Michel Gourgand, Fateh Mebrek & Alain Tanguy. *Modelling and Simulation of the pharmacy of a new hospital*. European Conference of Modelling and Simulation, 2007.
- [GUNAL 10] MM. Günal & M. Pidd. *Discrete event simulation for performance modelling in health care: a review of the littérature*. Journal of Simulation, no. 4, pages 42 – 51, 2010.
- [HAS 11] Haute Autorité de Santé (HAS). *Outils de sécurisation et d'auto-évaluation de l'administration des médicaments*. Juillet 2011.
- [HENDERSHOTT 96] Th. Hendershott. *Evaluating process change proposals in an outpatient pharmacy using simulation*. Proceedings Health Information and Management Systems Society (HIMSS), pages 165 – 174, 1996.
- [HLUPIC 05] V. Hlupic & G. de Vreede. *Business process modelling using discrete-event simulation: current opportunities and future challenges*. International Journal of Simulation and Process Modelling, no. 1, pages 72 – 81, 2005.
- [HUET 10] Jean-Charles Huet, Jean-Luc Paris, Michel Gourgand & Khalid Kouiss. *Modèle de connaissance générique du circuit du médicament dans un hôpital*. In MOSIM'10 : 8^{ième} ENIM IFAC Conférence Internationale de Modélisation et Simulation, Hammamet, Tunisie, Mai 2010.
- [HUET 11] Jean-Charles Huet. *Proposition d'une méthodologie de réingénierie pour le contrôle par le produit de systèmes manufacturiers : Application au circuit du médicament d'un hôpital*. Thèse de doctorat, Université Blaise Pascal – Clermont II, 2011.
- [JUN 99] J.B. Jun, S.H. Jacobson, J.R. Swisher. *Application of discrete event simulation in health care clinics: a survey*. Journal of Operational Research Society, no.50, pages 109 – 123, 1999.
- [LABORIE 08] Hervé Laborie & Sébastien Woynar. *Organisation et sécurisation du circuit du médicament, Approfondissement*. Rapport final, Mission national d'Expertise et d'Audit Hospitalier (MeaH), Juillet 2008.
- [LAGERGREN 98] M. Lagergren. *What is the role and contribution of models to management and research in health services? A view from Europe*. European Journal of Operational Research, no. 105, pages 257 – 266, 1998.
- [LEAPE 95] L.L. Leape, D.W. Bates, D.J. Cullen et al. *Systems analysis of adverse drug events*. ADE Prevention Study Group. Journal of the American Medical Association, no. 274, pages 35 – 43, Juillet 1995.

- [LENT 12] Wineke Am van Lent, Peter Vanberkel & Wim H van Harten. *A review on the relation between simulation and improvement in hospitals*. BMC Medical Informatics & Decision making, pages 12 – 18, 2012. Numéro DOI: 10.1186/1472-6947-12-18.
- [LE MOIGNE 77] Jean-Louis Le Moigne. *La théorie du système général*. Volume 2130433235, 1977.
- [LORINO 95] Philippe Lorino. *Le déploiement de la valeur par les processus*. Revue Française de Gestion, no. 104, pages. 55 – 71, Juin-Juillet-Août 1995.
- [LORINO 98] Philippe Lorino & Jean-Claude Tarondeau. *De la stratégie aux processus stratégiques*. Revue Française de Gestion, no. 117, pages. 5 – 17, Janvier-Février 1998.
- [LOWERY 98] J.C. Lowery. *Getting started in simulation in health care*. Proceedings of Winter Simulation Conference. 1998.
- [MAYER 92] R.J. Mayer, M.K. Painter & P. S. deWitte. *IDEF Family of Methods for Concurrent Engineering and Business Re-engineering Applications*. Knowledge-Based Systems, Inc., 1992.
- [MARCOTTE 94] François Marcotte. *Contribution à la modélisation des systèmes de productions : extension du modèle GRAI*. Thèse de doctorat, Université de Bordeaux 1, 1994.
- [MUKHERJEE 91] Arup K. Mukherjee. *A simulation model for management of operations in the pharmacy of a hospital*. Simulation, vol. 56, no. 2, pages 91 – 103, Février 1991.
- [MULLER 11] Richard Müller & Andreas Rogge-Solti. *BPMN for Healthcare Processes*. Proceedings of the 3rd Central-European Workshop on Services and their Composition, pages 65 – 72, Karlsruhe, Allemagne, 2011.
- [OUZAYD 12] F. Ouzayd, J. Saadi, J. Benhra & A. Zakari. *Medecine Drugs Circuit: Analyze Performance with Simulation Models*. International Journal Of Computer Applications, Vol. 53, no. 14, pages 44 – 50, Septembre 2012.
- [PAOLUCCI 97] E. Paolucci, F. Bonci & V. Rucci. *Redesigning organisations through business process re-engineering and object-orientation*. Proceedings of the European Conference on Information Systems, Cork, Irlande, pages 587 – 611, 1997.
- [PIDD 09] Michael Pidd. *Tools for Thinking: Modelling in Management Science, 3rd Edition*. John Wiley & Sons Ltd, Numéro ISBN : 978-0-470-72142-1, Octobre 2009.
- [PIERREVAL 90] Henri Pierreval. *Les méthodes d'analyse et de conception des systèmes de production*. Numéro ISBN : 2-86601-201-1. 1990.
- [RECKER 08] Jan Recker. *BPMN Modeling – Who, Where, How and Why*.

- [RECKER 09] Jan Recker, Martha Indulska, Michael Rosemann & Peter Green. *Business Process Modeling – A Comparative Analysis*. Journal of the Association for Information Systems, no. 10, Issue 4, pages 333 – 363, Avril 2009.
- [RECKER 10] Jan Recker. Opportunities and constraints: the current struggle with BPMN. *Business Process Management Journal*, no. 16, Issue 1, pages 181 – 201, 2009.
- [REYNOLDS 11] Matthew Reynolds, Christos Vasikalis, Monsey McLeaod, Nicholas Barber, Ann Mounsey, Sue Newton, Ann Jacklin & Bryony Dean Franklin. *Using discrete event simulation to design a more efficient hospital pharmacy for outpatients*. *Health Care Management Science*, no. 14, pages 223 – 236, 2011. Numéro DOI : 10.1007/s10729-011-9151-1.
- [ROBINSON 04] Stewart Robinson, Richard E. Nance, Ray J. Paul, Michael Pidd & Simon J.E. Taylor. *Simulation model reuse: definitions, benefits and obstacles*. *Simulation Modelling Practice and Theory*, no. 12, pages 479 – 494, 2004.
- [RODIER 10] Sophie Rodier. *Une tentative d'unification et de résolution des problèmes de modélisation et d'optimisation dans les systèmes hospitaliers. Application au Nouvel Hôpital Estaing*. Thèse de doctorat, Université Blaise Pascal – Clermont II, 2010.
- [ROHLEDER 07] T.R. Rohleder, D.P. Bischak, L.B. Baskin. *Modeling patient service centers with simulation and system dynamics*. *Health Care Management Science*, no. 10, pages 1 – 12, 2007.
- [ROJO 08] Marcial Garcia Rojo, Elvira Rolon, Luis Calahorra, Felix Oscar Garcia, Rosario Paloma Sanchez, Francisco Ruiz, Nieves Ballester, Maria Armenteros, Teresa Rodriguez & Rafael Martin Espartero. *Implementation of the Business Modelling Notation (BPMN) in the modelling of anatomic pathology processes*. *Diagnostic Pathology*, no. 3, supplement 1, 2008. Numéro DOI: 10.1186/1746-1596-3-SI-S22.
- [ROSS 77] D.T. Ross & K.E. Schoman. *Structured Analysis for Requirements Definition*. *IEEE Transactions on Software Engineering*, no. 3, pages 6 – 15, 1977.
- [SARRAMIA 02] David Sarramia. *ASCI-mi: une méthodologie de modélisation multiple et incrémentielle. Application aux systèmes de trafic urbain*. Thèse de doctorat, Université Blaise Pascal – Clermont-Ferrand II, 2002.
- [SCHMITT 06] Etienne Schmitt, Daniel Antier, Chantal Bernheim, Edith Dufay, Marie Caroline Husson & Edgar Tissot. *Dictionnaire français de l'erreur médicamenteuse*. Numéro ISBN : 2-9526010-0-3, Janvier 2006.
- [SINREICH 04] David Sinreich & Yariv N. Marmor. *A simple and intuitive tool for analysing emergency department operations*. *Winter Simulation Conference*, pages 1994 – 2002, 2004.
- [SPRY 05] Charles W. Spry & Mark A. Lawley. *Evaluating hospital pharmacy staffing and work scheduling using simulation*. *Winter Simulation Conference, Proceedings of the 37th conference*, pages 2256 – 2263, 2005.

- [STRASSER 11] Mélanie Strasser, Franz Pfeifer, Emmanuel Helm, Andreas Schuler & Josef Altmann. *Defining and Reconstructing Clinical Processes Based on IHE and BPMN 2.0*. User Centred Networked Health Care, IOS Press, 2011. Numéro ISBN: 978-1-60750-805-2 (print) | 978-1-60750-806-9 (online)
- [TARDIEU 83] H. Tardieu, A. Rochfeld & R. Colletti. *La méthode merise – tome 1 principes et outils*. Numéro ISBN : 2-7081-1106-X, 1983.
- [THORWARTH 09] Michael thorwarth & Amr Arisha. *Application of Discrete-Event Simulation in Health Care : a Review*. Dublin Institute of Technology, Report, 2009.
- [WESKE 04] M. Weske, W.M.P. van der Aalst & H.M.W. Verbeek. Advances in business process management. *Data & Knowledge Engineering*, no. 1, pages 1 – 8, 2004.
- [WONG 03] Cathy Wong, Glen Geiger, Yaron D. Derman, Carolyn R. Busby & Michael W. Carter. *The use of simulation for a hospital's pharmacy for redesigning the medication ordering, dispensing and administration process*. Proceedings of the Winter Simulation Conference, 2003.
- [YURTKURAN 08] Alkin Yurtkuran & Erdal Emel. *Simulation based decision-making for hospital pharmacy management*. Proceedings of the Winter Simulation Conference, 2008.
- [ZUR MUEHLEN 08] Michael Zur Muehlen & Jan Recker. How much language is enough? Theoretical and Practical Use of the Business Process Modeling Notation. In Proceedings 20th International Conference on Advanced Information Systems Engineering, Montpellier, France, 2008.

A

ANNEXES

SOMMAIRE

A.1	ANNEXE 1 : Données du rapport GEM	338
A.2	ANNEXE 2 : Données de l'étude MERVEIL.....	340
A.3	ANNEXE 3 : Critères de généralité de [Fletcher 09a].....	342
A.4	ANNEXE 4 : Compléments du langage BPMN	344
A.5	ANNEXE 5 : Propriétés des éléments de SIMIO.....	345
A.6	ANNEXE 6 : Guide mathématique de SIMIO	348

Tableau A.2 : Origine des erreurs entre 2005 et 2008

	2005	2006	2007	2008	TOTAL
Similitude des conditionnements	47	118	52	128	345
Défaut d'information	15	37	23	35	110
Défaut de conditionnement	12	24	12	24	72
Similitude des noms	5	9	9	13	36
Problème de qualité pharmaceutique	1	10	10	7	28
Mésusage	1	0	4	2	7
Erreur d'utilisation	2	21	31	78	132
Erreur de pratique	14	54	87	134	289
Autres	5	14	17	13	49
Non spécifié	0	0	0	14	14
TOTAL	102	287	245	448	1082

Causes d'erreurs

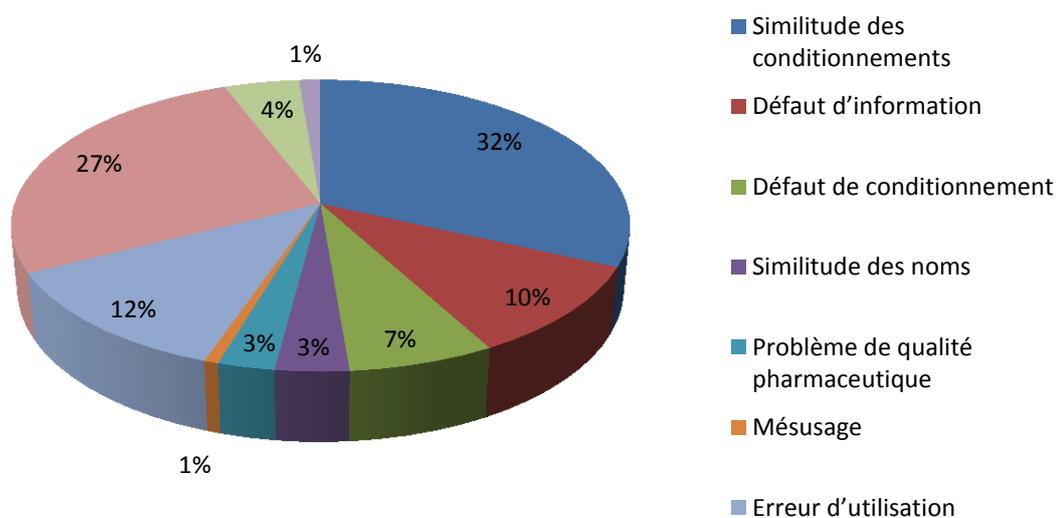


Figure A.2 : Répartition des causes d'erreurs entre 2005 et 2008

A.2 ANNEXE 2 : Données de l'étude MERVEIL

Les causes d'une erreur médicamenteuse sont multifactorielles, cette étude identifiant 2648 causes aux 295 EM soit 9 causes par EM en moyenne. MERVEIL répertorie de plus ces causes dans différentes catégories dont sept majeures :

Tableau A.3 : Nature des causes des EM

Nature de la cause	Nombre d'erreurs associées	Pourcentage
Moyens humains	283	96
Méthode	258	87
Management	183	62
Médicament et DM associé	173	59
Milieu	154	52
Matériel	127	43
Malade	71	24

Cependant seul 59.6% des causes ont pu être identifiées pour les EM, ce qui représente un total de 1577 sur 2648 dont 1249 parmi les sept catégories sus citées.

En détails, nous obtenons :

Tableau A.4 : Vue détaillée des causes des EM

Classement	Libellé de la cause	Fréquence de citation
	283 erreurs impliquant la branche « Moyens humains »	
1	Défauts de connaissance	200
4	Non-respect des bonnes pratiques	96
7	Charge de travail trop élevée	72
11	Distraction, interruption de tâches	54
12	Stress, situation d'urgence	51
	258 erreurs impliquant la branche « Méthode »	
2	Absence de contrôle avant administration	118
5	Délivrance erronée, absente, retardée	89
8	Protocole absent, obsolète, inconnu, non expliqué	67
	183 erreurs impliquant la branche « Management »	
6	Absence ou défaillance dans la gestion du personnel	74
9	Problème de coordination	69
10	Absence ou défaillance de gestion des ressources humaines	66
	173 erreurs impliquant la branche « Médicament »	
3	Confusion (conditionnement, dosage, étiquetage)	110
	154 erreurs impliquant la branche « Milieu »	
3	Qualité des échanges professionnels	113
	127 erreurs impliquant la branche « Matériel »	
11	Informatique (complexité logiciel, défaut d'alertes)	56

L'étude MERVEIL mettait aussi en avant le taux d'interception des erreurs à chaque étape.

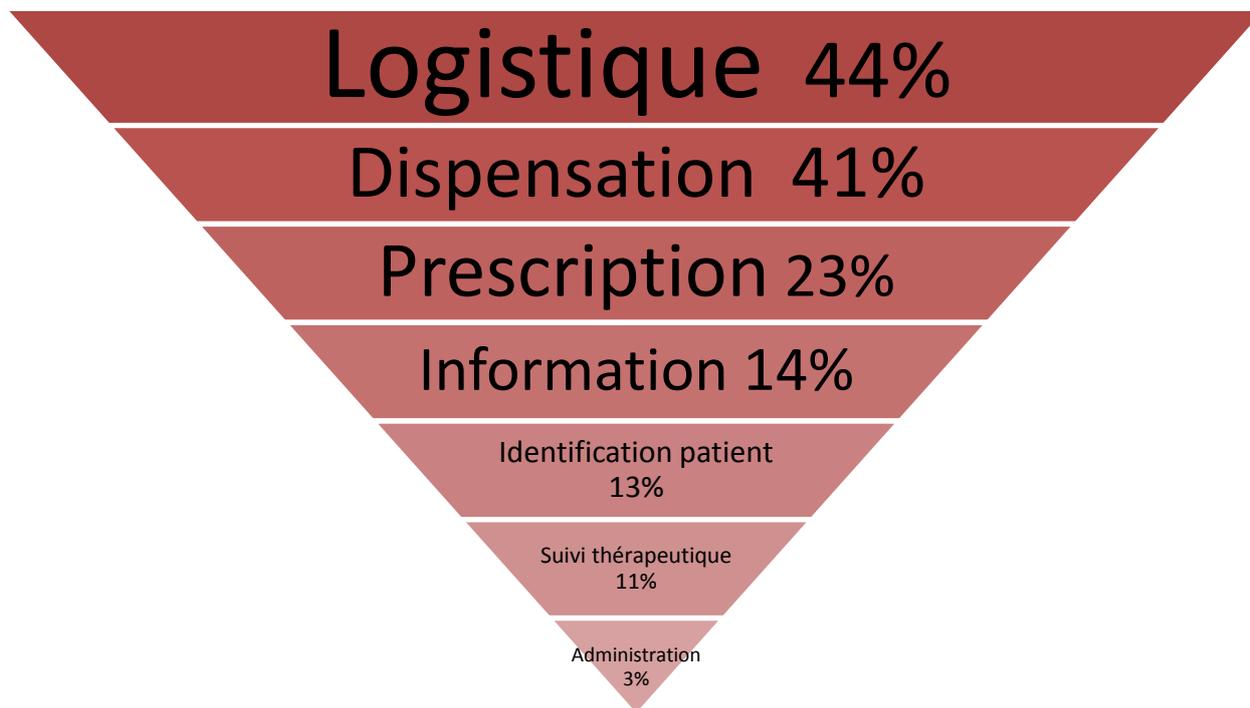


Figure A.230 : Vue hiérarchique du taux d'interception des erreurs à chaque étape du CDM

A.3 ANNEXE 3 : Critères de généricité de [Fletcher 09a]

Issue	Level 3A: generic model—central	Level 3B: generic model—multiple applicability	Level 4: specific model	Similarity
Scope	Depends on problem	Depends on problem	Depends on problem	***
Purpose	Some or all of the following: 1. To make general observations about the design and performance of a service. 2. To identify high impact interventions to improve the service 3. To build common understanding of the system amongst key stakeholders 4. To build greater understanding of the theoretical capability of the system	To be transportable enough to achieve some or all of purposes 1-4 in a number of specific local services. 5. To model the local system, and proposed changes, in detail	Some or all of the following: 1. To make observations about the design and performance of a local service. 2. To identify high impact interventions to improve the local service 3. To build common understanding of the system amongst key local stakeholders 4. To build greater understanding of the theoretical capability of the system 5. To model the current system, and proposed changes, in detail	**
Level of insight	Broad discussion of issues and potential high impact interventions	From broad discussion of issues to accurate and detailed local improvement strategies	From broad discussion of issues to accurate and detailed local improvement strategies	**
Level of accuracy	Possibly lower	Possibly higher	Possibly highest	**
Conflict with existing models	Addresses perceived/actual gaps in existing central models	May conflict with existing local models, or may have been designed to fill a perceived local gap.	Addresses perceived/actual gaps in existing local models	**
Target level of use	Centrally	In multiple providers	In a single provider	*

Figure A.4 : Critères clés de la généricité durant l'initiation du projet

Issue	Level 3A: generic model—central	Level 3B: generic model—multiple applicability	Level 4: Specific model	Similarity
Levels of code/model reuse	Depends on previous work by the modeller or organisation, degree of knowledge sharing or availability of generic software code/modules	Depends on previous work by the modeller or organisation, degree of knowledge sharing or availability of generic software code/modules	Depends on previous work by the modeller or organisation, degree of knowledge sharing or availability of generic software code/modules	***
Design process	Working with central experts, plus local service(s)	Working with central experts, plus local service(s)	Working with local expert(s)	**
Representation of local issues	Not necessarily required in detail.	By changing data inputs, not model structure.	Potentially better representation—through model structure and design as well as input data	**
Appropriateness of use	Clearly defined systems and problems for which the model is applicable. Model may be adjustable to fit different problems	Need to clearly define when model can be appropriately used—i.e. when modification of input data is appropriate to model the problem.	Greater likelihood of local adjustment of model structure to fit the problem	**
Level of detail	Inclusive of common significant processes in multiple providers	Inclusive of common significant processes in multiple providers	Inclusive of significant (locally agreed) processes in the particular provider	**
Design objective	To model common processes in a range of providers.	To model common processes but ideally flexible enough to model some local process differences through input data	To model one local system. The local process may be similar to other providers, but not a design objective	**
User knowledge of structure and inner workings of model	If users have been involved in developing model, possibly high.	Initially low (local users weren't involved in design),	Local user(s) likely to have been involved in model build, so probably higher?	**

Figure A.5 : Critères clés de la généricité durant la conception et le développement

Issue	Level 3A: generic model—central	Level 3B: generic model—multiple applicability	Level 4: Specific model	Similarity
Data quality	Depends on type of model/required accuracy	Depends on type of model/required accuracy	Depends on type of model/required accuracy	***
Data accessibility	Ideally easily accessible from standard operational systems	Ideally easily accessible from standard operational systems	Ideally easily accessible from standard operational systems. Extra local collections possible.	**
Test “starting” data	Possibly “national average”, plus local examples and national estimates	Possibly “national average”, plus local examples	Local historical data	**

Figure A.6: Critères clés de la généralité pour les données

Issue	Level 3A: generic model—central	Level 3B: generic model—multiple applicability	Level 4: Specific model	Similarity
Technique	“Open box” with national and local experts, “black box” with national/local data	“Open box” with national and local experts, “black box” with local data	“Open box” validation with local experts, plus “black box” with local data	***
Representation	“General” level only	A range of individual providers	One provider	**
Required accuracy	Depends on purpose of model Broad accuracy against average data may be enough	Depends on purpose of model Medium to high levels of accuracy against local data.	Depends on purpose of model High levels of accuracy probably required.	**

Figure A.7 : Critères clés de la généralité durant la validation

Issue	Level 3A: generic model—central	Level 3B: generic model—multiple applicability	Level 4: Specific model	Similarity
Capability to use it	May not be handed over. If handed over, ideally low required user capability, clean and transparent input/output interfaces. If user was involved in development, may be less need for this.	May not be handed over. If handed over, ideally low required user capability, clean and transparent input/output interfaces.	May not be handed over. If handed over, ideally low required user capability, clean and transparent input/output interfaces. If user was involved in development, may be less need for this.	***
Establishing model as operational tool	Demo and workshops of validated model with key stakeholders. Use on currently important issues/scenarios. Demonstration that model works on “real” important issues.	Depending on model, local site visits, demo/workshops, validation and supported model runs may be required. Use on currently important issues/scenarios. Demonstration that model works on “real” important issues.	Demo and workshops of validated model with key stakeholders. Use on currently important issues/scenarios. Demonstration that model works on “real” important issues.	***
Post development user support or use of model	If not handed over, modelling team must be geared up to quickly perform runs and communicate results effectively. If handed over, geared towards handover to user(s) at the particular department. If the users are technically proficient or involved in model development, less need for “clean” user interfaces, documentation etc.	If not handed over, modelling team must be geared up to quickly perform runs and communicate results effectively. This may be “remote” support. If “handed over”, geared towards multiple (geographically spread) users and/or regional/national teams with clean user interface and user guide/documentation.	If not handed over, modelling team must be geared up to quickly perform runs and communicate results effectively. If handed over, geared towards particular central user(s). If the users are technically proficient or involved in model development, less need for “clean” user interfaces, documentation etc.	***
Desire to use it	Should be high, as model was commissioned, but central politics still a factor. Depends on perception of quality, coverage, plus existence of other models	Depends on perception of quality, coverage, plus existence of other models and local politics.	Should be high, as model was locally commissioned, local politics still a factor. Depends on perception of quality, coverage, plus existence of other models	**
Hardware/software requirements	Ideally inexpensive or free, but some investment in software may be needed.	Probably more important that no investment in software is required.	Ideally inexpensive or free, but some investment in software may be needed	**

Figure A.8 : Critères clés de la généralité durant l’implémentation

A.4 ANNEXE 4 : Compléments du langage BPMN

Types	Start			Intermediate				End
	Top-Level	Event Sub-Process Interrupting	Event Sub-Process Non-Interrupting	Catching	Boundary Interrupting	Boundary Non-Interrupting	Throwing	
None								
Message								
Timer								
Error								
Escalation								
Cancel								
Compensation								
Conditional								
Link								
Signal								
Terminate								
Multiple								
Parallel Multiple								

Figure A.9 : Ensemble des évènements existants dans BPMN 2.0

A.5 ANNEXE 5 : Propriétés des éléments de SIMIO

Tableau A.5 : Ensemble des propriétés de l'élément « Prescriptions »/ « Demande »

Nom de la propriété	Type	Type de variable	Rôle pour la demande/prescription
Indicedemande	States	Entier	Indice du type de demande
Typedemande	States	Chaînes de caractères	Type de demande
Nbrproduitdemande	States	Réel	Nombre de produits
Tempsprepa	States	Réel	Temps de préparation
Validationdemande	States	Entier	Si prescription validée égal à 1 Sinon égal à 0
Modetraitementdemande	States	Entier	Suivant la valeur définit le mode de préparation
Preparateurassocie	States	Référence Objet	Enregistre le préparateur ayant commencé à préparer cette demande/prescription (niveau modèle)
Service_cible	States	Entier	Le numéro du service ayant fait la commande
Heure_affectation_valideur	States	Réel	Enregistre l'heure de début de validation
Entite_hybride	States	Entier	Si égal à 1 la demande est un élément d'une demande mère
Niveau_analyse_valid	States	Entier	Détermine le niveau d'analyse de la prescription
Numero_demande	States	Entier	Numéro d'identification de la demande dans SIMIO
Nombre_elmt_par_produit	States	Entier	Nombre d'éléments de chaque produit dans une demande
Hybridite	States	Entier	Détermine la nature hybride (=1) ou non (=0) d'une demande/prescription
Nombre_ss_entite_hybride	States	Entier	Enregistre le nombre de sous-demandes issu d'une demande mère
Preparateur_pour_hybride	States	Référence Objet	Enregistre le numéro du préparateur s'occupant de la demande mère
Demande_Parent	States	Booléenne	Si égal à 1 il s'agit de la demande mère sinon égale à 0
Numero_prepa_necessaire	States	Entier	Récupère le numéro du préparateur devant préparer cette demande/prescription (si existant)
Retard_demande	States	Réel	Impose un retard minimum à toute préparation

Nbr_pdt_en_stockeur	States	Réel	Défini le nombre de produits de la demande/prescription en stockeur
Nbr_elmts_pdt_stockeur	States	Réel	Défini le nombre d'éléments par produit de la demande/prescription en stockeur
Temps_prepa_pdt_stockeur	States	Réel	Défini le temps de préparation des produits de la demande/prescription en stockeur
Nbr_pdt_solute	States	Réel	Défini le nombre de produits de la demande/prescription qui sont des solutés
Nbr_elmts_pdt_solute	States	Réel	Défini le nombre d'élément par soluté de la demande/prescription
Temps_prepa_solute	States	Réel	Défini le temps de préparation des solutés de la demande/prescription
Nbr_pdt_au_frigo	States	Réel	Défini le nombre de produits de la demande/prescription en frigo
Nbr_elmts_pdt_solutes	States	Réel	Défini le nombre d'éléments par produit frigorifique de la demande/prescription
Temps_prepa_pdt_frigo	States	Réel	Défini le temps de préparation des produits frigorifiques de la demande/prescription
Nbr_demande_nominative	States	Réel	Défini le nombre de prescription nominative comprise dans la demande
Nbrproduitdemande_initial	States	Entier	Défini le nombre de produits contenus dans la demande avant d'en soustraire les solutés, les produits frigorifiques et ceux en stockeur

Tableau A.6 : Assignment des états des demandes/prescriptions par la source « Service »

Caractéristique	Variable de la table/Formule/Variable SIMIO
Prescriptions.Indicedemande	Donneesdemande.Indicedemande
Prescriptions.Typedemande	Donneesdemande.Typedemande
Prescriptions.Nbrproduitdemande	Math.If((DateTime.Hour(TimeNow)+DateTime.Minute(TimeNow))/60+DateTime.Second(TimeNow)/3503.65)<12, Donneesdemande.Nbrproduit_matin, Donneesdemande.Nbrproduit_apmidi)
Prescriptions.Tempsprepa	Donneesdemande.Tempsprepa
Prescriptions.Validationdemande	Donneesdemande.Validation
Prescriptions.Modetraitementdemande	Donneesdemande.Modetraitement
Prescriptions.Entite_hybride	0
Prescriptions.Numero_demande	Prescriptions.ID
Prescriptions.Nombre_elmt_par_produit	Donneesdemande.Nbrelmts
Prescriptions.Hybridite	Donneesdemande.Possible_hybride
Prescriptions.Retard_demande	Random.Exponential(1)
Prescriptions.Nbr_demande_nominative	Donnees_demandes_hybrides[Prescriptions.Indicedemande].Nbr_nominative
Prescriptions.Nbrproduitdemande_initial	Prescription.Nbrpdtdemande
Prescriptions.Tempsprepa	Donneesdemande.Tempsprepa+Prescription.Retard_demande
Prescriptions.Priority	1
Moyenne_produit	(Générateur_demandes.OutputBuffer.NumberExited * Moyenne_produit + Prescriptions.Nbrproduitdemande) / (Générateur_demandes.OutputBuffer.NumberExited + 1)
Moyenne_retard	(Générateur_demandes.OutputBuffer.NumberExited * Moyenne_retard + Prescriptions.Retard_demande) / (Générateur_demandes.OutputBuffer.NumberExited + 1)
Moyenne_temps	(Générateur_demandes.OutputBuffer.NumberExited * Moyenne_temps + Prescriptions.Tempsprepa) / (Générateur_demandes.OutputBuffer.NumberExited + 1)

A.6 ANNEXE 6 : Guide mathématique de SIMIO

❖ Fonctions mathématiques utiles :

- `Math.Abs(valeur)` : retourne la valeur absolue de la valeur associée ;
- `Math.Floor(value)` : retourne la plus grande valeur entière non supérieur à la valeur testée ;
- `Math.If(test1, valeur1, test2, valeur2, ..., sinon autre valeur)` : retourne la « valeurN » si le « testN » est vrai. Sinon retourne « autre valeur » ;
- `Math.Round(valeur)` : retourne la valeur entière la plus proche. Par exemple pour 0,5 retourne la valeur « 0 » et pour 0,51 retourne la valeur « 1 » .

❖ Distributions mathématiques utiles :

- Distribution discrète « `Random.Discrete(valeur1, proba1, valeur2, proba2, ..., valeurN, probaN)` » : Distribution empirique définissant un ensemble de valeur avec des probabilités cumulatives. Ainsi « valeur1 » a une probabilité « proba1 » d'être tirée, « valeur2 » une probabilité « proba2 - proba1 », etc ;
- Distribution exponentielle « `Random.Exponential(moyenne)` » ;
- Distribution normale « `Random.Normal(moyenne, deviation standard)` » ;
- Distribution triangulaire « `Random.Triangular(minimum, mode, maximum)` » : La valeur mode représente la valeur la plus probable d'être tirée ;
- Distribution uniforme « `Random.Uniform(minimum, maximum)` » : Tire uniformément une valeur entre le minimum et le maximum.

A noter que SIMIO propose aussi des distributions beta, binomial, de Weibull, de Poisson, gamma, etc.

Titre Proposition d'une méthodologie de modélisation et de réorganisation du circuit du médicament dans les pharmacies hospitalières.

Résumé Les problématiques d'amélioration et de sécurisation du circuit du médicament, restent à ce jour deux thématiques majeures auxquelles les hôpitaux tentent d'apporter les meilleures réponses possibles. DJIN (Dispensation Journalière Individuelle Nominative), informatisation, robotisation ou dose unitaire font parties des nombreuses solutions existantes permettant de fournir aux patients des soins de meilleures qualités, mais aussi et surtout plus sûrs. Les hôpitaux ont par conséquent le besoin et la nécessité de se réorganiser. Malheureusement, la mise en application de cette réorganisation n'est pas toujours évidente et couronnée de succès, du fait du manque de méthodes et d'outils pouvant accompagner les décisionnaires. Ainsi, dans le but de faciliter cette réorganisation, nous proposons par le biais de cette étude de fournir une nouvelle démarche et des outils permettant à ces établissements et notamment aux pharmaciens de réaliser des tests de scénarios eux-mêmes.

Focalisé sur le circuit du médicament, nous utilisons la méthodologie ASCI afin de construire cette nouvelle démarche, générique et adaptée aux pharmaciens. Complétée par l'utilisation des langages BPMN et UML, l'utilisation d'ASCI mène à l'obtention d'un modèle de simulation générique, ayant nécessité la création d'un guide des bonnes pratiques afin d'encadrer le passage du modèle théorique sous BPMN vers ce modèle de simulation sous SIMIO. Aboutissant à une démarche structurée et adaptée à un corps professionnel non coutumier de ces concepts, nous appliquons dans un dernier temps celle-ci au CHU de Clermont-Ferrand, et simulons divers scénarios d'évolutions, dans le but de proposer une aide à la décision à cet établissement.

Mots-clés ASCI, Circuit du médicament, BPMN, simulation.

Title Proposition of modelling and reorganization methodology for the medication-use process in hospital pharmacies.

Abstract To secure and improve medication-use process are nowadays two important challenges for hospitals. Nominative dispensing, computerization, robots, and unit doses are some possible solutions enabling to give better and more secured treatment to patients. Consequently, hospitals need to reorganize and redesign their working. Unfortunately, such actions are not easy and successful due to lack of methods and tools to help decision-makers. So, in order to fill this gap, we develop in this work a new approach and three generic tools enabling health structures, and particularly pharmacists, to test some scenarios and realize themselves this reorganization.

Focused on medication-use process, we use ASDI methodology to build a new generic approach, dedicated to pharmacists. Completed with the use of BPMN and UML languages, ASDI leads to a generic simulation model, which needed the conception of a "good practice guide" to structure the translation of the BPMN theoretical model to this simulation model, developed with SIMIO®. Leading to a structured approach adapted to a profession not used to work with these concepts, we apply our work on the University Hospital of Clermont-Ferrand, and simulate various scenarios of evolution in order to help this hospital to improve its working.

Keywords ASDI, Medication-use process, BPMN, Simulation.