



HAL
open science

Proposition d'une méthodologie pour la conception des systèmes de production reconfigurables et d'un outil associé d'aide à la décision par simulation de flux

Khaled Lameche

► To cite this version:

Khaled Lameche. Proposition d'une méthodologie pour la conception des systèmes de production reconfigurables et d'un outil associé d'aide à la décision par simulation de flux. Automatique. Université Bretagne Loire, 2018. Français. NNT : 2018NANT4008 . tel-02481924

HAL Id: tel-02481924

<https://hal.science/tel-02481924>

Submitted on 17 Feb 2020

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

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

Thèse de Doctorat

Khaled LAMECHE

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université de Nantes
sous le sceau de l'Université Bretagne Loire*

École doctorale : *Sciences & Technologies de l'Information et Mathématiques STIM*

Discipline : *Automatique et Informatique Appliquée*

Spécialité : *Automatique et Productique*

Unité de recherche : *LS2N*

Soutenu le *12/02/2018*

Thèse N° :

Proposition d'une méthodologie pour la conception des systèmes de production reconfigurables et d'un outil associé d'aide à la décision par simulation de flux

JURY

Président du jury	Alexandre DOLGUI, Professeur, IMT Atlantique
Rapporteurs :	Samir LAMOURI, Professeur, ENSAM PARIS Nikolay TCHERNEV, Professeur, Université d'Auvergne
Examineurs :	Muriel LOMBARD, Maitre de conférences HDR, IUT Nancy-Brabois Alexandre DOLGUI, Professeur, IMT Atlantique Pascal BERRUET, Professeur, Université Bretagne Sud Khalid KOUISS, Maitre de conférences, SIGMA Clermont-Ferrand
Invité(s) :	Alexis GIRIN, Docteur, IRT Jules Verne Francisco GAMBOA, Docteur, IRT Jules Verne
Directeur de Thèse :	Pierre, CASTAGNA, Professeur, Université de Nantes
Co-directeur de Thèse :	Najib Mohamed NAJID, Maitre de conférences HDR, Université de Nantes

Remerciements

Je remercie chaleureusement toutes les personnes qui m'ont aidé pendant l'élaboration de ma thèse et notamment mon directeur de thèse, monsieur le professeur Pierre CASTAGNA, pour son intérêt et son soutien, sa grande disponibilité et ses nombreux conseils pendant la préparation de ma thèse. Je le remercie plus particulièrement pour son précieux support concernant la simulation des systèmes de production. Mes remerciements à mon encadrant-monsieur Najib M. NAJID, je le remercie pour son encadrement, son soutien et ses conseils durant la réalisation de cette thèse.

Cette thèse fait partie du projet STAR (Système Transitaire Agile et Robotisé). La réalisation de ce projet n'aurait pas été possible sans le soutien de l'Institut de Recherche Technologique (IRT) Jules Verne, du Laboratoire des Sciences du Numérique de Nantes (LS2N) et des sociétés suivantes : Faurecia, DAHER et BA-systèmes. Grâce à leur financement, j'ai pu me consacrer sereinement à l'élaboration de ma thèse.

Je tiens à remercier l'IRT Jules Verne dont je faisais partie du personnel. Je remercie tous les collaborateurs de cet institut et plus particulièrement madame Linda ALLAL-GAZIO qui m'a constamment aidé pour les différentes démarches administratives, je remercie également mon supérieur hiérarchique monsieur Alexis Gerin et toute l'équipe robotique. Je remercie aussi : monsieur Arnaud FILEAUX, l'informaticien qui m'a toujours aidé avec joie, mes collaborateurs sur le projet STAR messieurs David Garnier et Francisco GAMBOA ainsi que le chef du projet monsieur Cédric TANO sans oublier monsieur David MARTINEZ et madame Valérie DONALD. Je remercie aussi monsieur Olivier Cardan, monsieur Claude MARTINEZ, madame Rosa ABBOU et plus particulièrement madame CHRISTINE BOUTHET ainsi que tout le personnel de laboratoire LS2N de m'avoir accueilli au sein de leur équipe à l'IUT de Carquefou.

Mes remerciements vont particulièrement au représentant de Faurecia monsieur Khalid KOUISS de m'avoir invité à faire partie du projet STAR et d'avoir appuyé ma candidature. Je remercie aussi monsieur Alain Chung, chef d'usine Faurecia à Nogent-Sur-Vernisson, de m'avoir accueilli au sein de son usine et de m'avoir fourni toutes les données nécessaires pour la réalisation de ma thèse. Je remercie tous les collaborateurs de Faurecia qui ont participé de près ou de loin à la réalisation du projet STAR. Je tiens à remercier les représentants de DAHER messieurs Edouard Rousseau et Nicolas LESTIEN de m'avoir accueilli au sein de leur plateforme logistique à Marignane et de m'avoir fourni toutes les données nécessaires à la réalisation de ma thèse. Je les remercie ainsi que tous les collaborateurs de DAHER qui ont participé au projet STAR. Je remercie aussi le personnel de BA-Systèmes et plus particulièrement messieurs Bertrand JACQ et Guy CAVEROT.

Ce travail n'aurait pu être mené à bien sans la disponibilité de mes amis : Kim Phuc TRAN, Meriam KOUKI, Dorra RAHALI, Georges El PANA, Reza MALIKI, Nasser FEKIRI, Fatima Sehar ZAIDI. Je les remercie pour les bons moments qu'on a passés ensemble.

Enfin, je remercie fortement la personne qui m'est très chère, ma mère. Je la remercie d'être toujours à mes côtés et de m'avoir toujours encouragé. Je remercie mes frères et sœurs et mon père décédé. Mes remerciements vont à ma future femme, je la remercie de m'avoir supporté et d'être à mes côtés.

À la mémoire de mon Père

Table des matières

TABLE DES MATIERES	1
LISTE DES FIGURES	8
LISTE DES TABLEAUX	10
INTRODUCTION GENERALE	13
CHAPITRE 01 : ÉTAT DE L'ART SUR LES SYSTEMES MANUFACTURIERS RECONFIGURABLES	15
1. INTRODUCTION	15
2. L'ÉVOLUTION DES SYSTEMES MANUFACTURIERS	15
2.1. LA PRODUCTION ARTISANALE	16
2.2. LA PRODUCTION DE MASSE	16
2.3. CUSTOMISATION DE MASSE	17
2.4. LA GLOBALISATION OU LA PRODUCTION PERSONNALISEE	17
2.5. CHANGEMENT DE PARADIGME.....	17
2.6. LE PARADIGME ET LE TYPE DU SYSTEME DE PRODUCTION.....	19
2.6.1. <i>Systèmes dédiés et les systèmes flexibles</i>	19
3. LE CONCEPT RMS	21
3.1. DEFINITION D'UN RMS.....	21
3.2. LES CARACTERISTIQUES D'UN RMS	22
3.3. LA CONFIGURATION ET LA RECONFIGURATION DU RMS.....	23
3.3.1. <i>La configuration du système</i>	23
3.3.2. <i>Les déclencheurs de la reconfiguration</i>	24
3.3.3. <i>La reconfiguration du système</i>	24
4. LA CONCEPTION DES RMSS	27
4.1. PROBLEMATIQUE DE CONCEPTION DES RMSS	29
4.2. PRINCIPES DE CONCEPTION DES RMSS.....	30
4.3. CONCEPTION DES PRODUITS ET FORMATIONS DES FAMILLES DE PRODUITS	30
4.4. PLANIFICATION/CONCEPTION DU PROCESSUS DE PRODUCTION	33
4.5. CONCEPTION AU NIVEAU SYSTEME : APPROCHES DE CONCEPTION GENERALE.....	34
4.5.1. <i>La conception/sélection de la configuration du système</i>	40
4.6. CONCEPTION DES RMSS : NIVEAU MACHINE	41
4.7. CONCEPTION DES RMSS : SYSTEME DE CONTROLE/PILOTAGE.	43
4.7.1. <i>Modélisation des RMSS</i>	45
4.8. CONCEPTION DES RMSS : GESTION DU SYSTEME.....	46
4.8.1. <i>Réduction du temps de montée en puissance</i>	46

4.8.2.	<i>Indicateurs de performances des RMSs</i>	47
4.8.3.	<i>Planification de la production</i>	47
5.	CONCLUSION	48
CHAPITRE 02 : INGENIERIE SYSTEMES ET LES METHODES DE CONCEPTION DES SYSTEMES COMPLEXES		51
1	INTRODUCTION	51
2	DEFINITIONS	52
3	INGENIERIE SYSTEME	53
3.1	DEFINITION	53
3.2	HISTOIRE	53
3.3	ORGANISMES TRAVAILLANT SUR L'IS.....	54
3.3.1	<i>INCOSE</i>	54
3.3.2	<i>AFIS</i>	54
4	CYCLE DE VIE D'UN SYSTEME/PRODUIT	54
4.1	LES PHASES DU CYCLE DE VIE D'UN SYSTEME	54
4.1.1	<i>Phase de concept</i>	56
4.1.2	<i>Phase de développement</i>	57
4.1.3	<i>Phase de production</i>	57
4.1.4	<i>Phase d'exploitation</i>	57
4.1.5	<i>Phase de maintenance</i>	57
4.1.6	<i>Phase de retrait</i>	57
5	LES PROCESSUS DE L'INGENIERIE SYSTEME	58
5.1	LES PROCESSUS TECHNIQUES	59
5.2	LES PROCESSUS DE MANAGEMENT DE PROJET	61
5.2.1.	<i>Le processus générique de management de projet</i>	61
5.2.2.	<i>Les processus de management plus spécifiques à l'IS</i>	62
5.3.	LES PROCESSUS CONTRACTUELS	62
5.4.	LES PROCESSUS DE L'ENTREPRISE.....	63
6.	LES NORMES DE L'INGENIERIE SYSTEME	63
6.1.	OBJECTIFS DE LA NORMALISATION	63
6.2.	CADRE D'APPLICATION DE LA NORMALISATION	64
6.3.	LIMITATIONS D'UNE NORMALISATION	64
6.4.	LES PRINCIPALES NORMES DE L'INGENIERIE SYSTEME	65
6.5.	LA NORME IEEE-1220.....	67
6.6.	LA NORME EIA/ANSI-632	68

6.7.	LA NORME ISO/IEC/IEEE-15288	69
7.	LES APPROCHES DE DEVELOPPEMENT D'UN SYSTEME	70
7.1.	LE MODELE EN CASCADE	70
7.2.	LE MODELE EN SPIRALE	72
7.3.	LE MODELE EN 'V'	74
7.4.	LE MODELE EN DOUBLE 'V'	77
8.	L'INGENIERIE SYSTEME DIRIGEE PAR LES MODELES MBSE	78
8.1.	MBSE DEFINITION	78
8.2.	LES AVANTAGES DES APPROCHES MBSE	79
8.3.	LES METHODOLOGIES MBSE	80
9.	CONCLUSION	91
CHAPITRE 03 : MODULARITE DANS LA CONCEPTION DES SYSTEMES MANUFACTURIERS RECONFIGURABLES 95		
1.	INTRODUCTION.....	95
2.	LA MODULARITE	96
2.1.	DEFINITION	96
2.2.	LES TYPES DE MODULARITE	96
2.3.	AVANTAGES DE LA MODULARITE	97
2.3.1.	<i>Perspective technique</i>	<i>97</i>
2.3.2.	<i>Perspective économique</i>	<i>98</i>
2.3.3.	<i>Perspective organisationnelle</i>	<i>98</i>
2.4.	LES INCONVENIENTS DE LA MODULARITE.....	98
3.	PRINCIPES DE CONCEPTION MODULAIRE.....	98
4.	EXEMPLES DE SYSTEMES MANUFACTURIERS MODULAIRES.....	99
4.1.	EXEMPLE 01 : MACHINE MODULAIRE.....	99
4.2.	EXEMPLE 02 : SYSTEME LOGISTIQUE MODULAIRE	100
4.3.	EXEMPLE 03 : SYSTEME DE PRODUCTION MODULAIRE.....	101
5.	TECHNIQUES ET METHODES DE CONCEPTION D'UN SYSTEME MODULAIRE	101
5.1.	MATRICE DE CONCEPTION STRUCTURELLE (DESIGN STRUCTURE MATRIX DSM)	102
5.1.1.	<i>Exemple d'application de la méthode DSM</i>	<i>103</i>
5.1.2.	<i>Application de la modularité pour la conception d'un système logistique reconfigurable</i>	<i>107</i>
6.	CONCLUSION	115
CHAPITRE 4 : UNE METHODOLOGIE GENERIQUE POUR LA CONCEPTION DES SYSTEMES MANUFACTURIERS RECONFIGURABLES		
		117

1. INTRODUCTION.....	117
2. CANEVAS DE LA METHODOLOGIE DE CONCEPTION DES RMSS.....	118
2.1. RECONFIGURATION DU SYSTEME SANS EVOLUTION	121
2.2. RECONFIGURATION DU SYSTEME AVEC EVOLUTION	122
2.3. L'ÉCHANGE DES DONNEES.....	123
2.4. PRINCIPE DE DECOMPOSITION	124
3. LES PROCESSUS ET LES ACTIVITES DE LA METHODOLOGIE	125
3.1. DEFINITION DES EXIGENCES DES PARTIES PRENANTES.....	127
3.1.1. <i>Conception autour d'une famille de produits et changement de capacité</i>	127
3.1.2. <i>Exigences et contraintes de reconfiguration</i>	127
3.1.3. <i>Contraintes de précédence de conception</i>	128
3.1.4. <i>Les activités et les tâches à réaliser</i>	128
3.1.5. <i>Les liens de traçabilité</i>	131
3.2. ANALYSE DU SYSTEME.....	133
3.2.1. <i>Les activités et les tâches à réaliser</i>	134
3.2.2. <i>Les liens de traçabilités</i>	135
3.3. LA CONCEPTION DE L'ARCHITECTURE DU SYSTEME	135
3.3.1. <i>La conception autour d'une famille de produits</i>	135
3.3.2. <i>La modularité</i>	136
3.3.3. <i>La simulation</i>	137
3.3.4. <i>Les activités et les tâches à réaliser</i>	137
3.3.5. <i>Liens de traçabilité</i>	138
3.4. LA VERIFICATION DU SYSTEME	140
3.4.1. <i>Processus de vérification</i>	140
3.4.2. <i>Les tâches et les activités à réaliser</i>	141
3.4.3. <i>Liens de traçabilité dans le processus « Vérification du système »</i>	141
3.4.4. <i>Processus de validation</i>	142
3.4.5. <i>Les tâches et activités à réaliser</i>	142
3.4.6. <i>Liens de traçabilité dans le processus « validation du système »</i>	143
4. CONCLUSION	144
CHAPITRE 05 : LA SIMULATION POUR LA CONCEPTION ET LA GESTION DES SYSTEMES MANUFACTURIERS RECONFIGURABLES	145
1. INTRODUCTION.....	145
2. LA SIMULATION DES FLUX.....	146
2.1. DEFINITION	146

2.2.	LES POINTS FORTS DE LA SIMULATION	146
2.3.	LES LIMITES DE LA SIMULATION	146
3.	ROLE DE LA SIMULATION DANS LA CONCEPTION DES RMSS	147
3.1.	POSITIONNEMENT DE LA SIMULATION DANS LA METHODOLOGIE GLOBALE	147
3.1.1.	<i>Utilisation de la simulation pendant la phase de développement du système</i>	<i>148</i>
3.1.2.	<i>Utilisation de la simulation pendant la phase opérationnelle du système</i>	<i>148</i>
4.	METHODOLOGIE DE CONDUITE D'UNE SIMULATION D'UN RMS.....	148
4.1.	ANALYSE DU PROBLEME	150
4.1.1.	<i>Modèle géométrique du système.....</i>	<i>150</i>
4.1.2.	<i>Modèle informationnel du système</i>	<i>151</i>
4.2.	IDENTIFICATION DES MODULES DE BASE DU SYSTEME	151
4.3.	DEVELOPPEMENT DE LA BIBLIOTHEQUE VIRTUELLE	152
4.4.	MODELISATION ET IMPLEMENTATION	153
4.4.1.	<i>Classification des modèles de simulation.....</i>	<i>154</i>
4.5.	EXPERIMENTATIONS SUR LE MODELE.....	155
4.6.	RAPPORT ET CONCLUSIONS	156
5.	CAS D'APPLICATION	156
5.1.	DESCRIPTION	156
5.2.	ANALYSE DU SYSTEME.....	157
5.3.	IDENTIFICATION DES MODULES DE BASE DU SYSTEME	158
5.4.	DEVELOPPEMENT DE LA BIBLIOTHEQUE VIRTUELLE	159
5.5.	MODELISATION ET IMPLEMENTATION	162
5.6.	EXPERIMENTATION (SCENARIOS DE RECONFIGURATION)	162
5.6.1.	<i>Configuration initiale</i>	<i>162</i>
5.6.2.	<i>Nouvelle Configuration</i>	<i>163</i>
5.7.	RAPPORT ET CONCLUSION (QUELQUES RESULTATS DE LA SIMULATION).....	165
5.7.1.	<i>Configuration initiale</i>	<i>165</i>
5.7.2.	<i>Nouvelle configuration.....</i>	<i>167</i>
6.	CONCLUSION	168
CHAPITRE 6 : APPLICATION SUR DES CAS D'ETUDE.....		171
1.	INTRODUCTION.....	171
2.	DESCRIPTION DES CAS D'ETUDE.....	171
2.1.	CAS D'ETUDE N°01: FAURECIA, USINE NOGENT-SUR-VERNISSON.....	171
2.2.	CAS D'ETUDE N°02: DAHER, PLATE-FORME LOGISTIQUE DE MARIGNANE.....	174

3.	APPLICATION DE LA METHODOLOGIE	175
3.1.	ANALYSE DU SYSTEME A DEVELOPPER.....	176
3.2.	PROCESSUS : DEFINITION DES BESOINS/EXIGENCES DES PARTIES PRENANTES	176
3.2.1.	<i>Sous-système : AGV.....</i>	<i>177</i>
3.2.2.	<i>Sous-système : système de pilotage/contrôle.....</i>	<i>179</i>
3.3.	PROCESSUS SPECIFICATION DES EXIGENCES DU SYSTEME	181
3.3.1.	<i>Sous-système : AGV.....</i>	<i>181</i>
3.3.2.	<i>Sous-système : système de pilotage/contrôle.....</i>	<i>182</i>
3.4.	PROCESSUS : CONCEPTION DE L'ARCHITECTURE, REALISATION, ET VERIFICATION	184
3.4.1.	<i>Sous-système : AGV.....</i>	<i>184</i>
3.4.2.	<i>Sous-Système : Système de contrôle</i>	<i>184</i>
3.5.	PROCESSUS VALIDATION DU SYSTEME	185
4.	LA BIBLIOTHEQUE VIRTUELLE.....	185
4.1.	MAQUETTE NUMERIQUE	187
4.1.1.	<i>Maquette numérique Faurecia</i>	<i>188</i>
4.1.2.	<i>Maquette numérique Daher</i>	<i>189</i>
5.	LES RESULTATS DE LA SIMULATION	190
6.	CONCLUSION	191
	CONCLUSION GENERALE ET TRAVAUX FUTURS	193
	ANNEXE 01 : IMPLEMENTATION DE LA METHODOLOGIE SUR POLARION	195
1.	INTRODUCTION.....	195
2.	POLARION.....	195
3.	IMPLEMENTATION DE LA METHODOLOGIE SUR POLARION	196
3.1.	PAGE D'ACCUEIL DE LA METHODOLOGIE	196
3.2.	PROCESSUS DEFINITION DES BESOINS DES PARTIES PRENANTES	196
3.3.	PROCESSUS SPECIFICATION DU SYSTEME	197
3.4.	PROCESSUS DE CONCEPTION DE L'ARCHITECTURE DU SYSTEME	198
3.5.	PROCESSUS VERIFICATION DU SYSTEME	198
3.6.	PROCESSUS VALIDATION DU SYSTEME	199
3.7.	LES ELEMENTS DU TRAVAIL IMPLEMENTES.....	200
3.7.1.	<i>Les éléments du travail généraux</i>	<i>200</i>
3.7.2.	<i>Les éléments de travail des systèmes reconfigurables.....</i>	<i>202</i>
3.8.	LE CYCLE DE VIE D'UNE EXIGENCE	203
3.9.	LE CYCLE DE VIE D'UN TEST DE VERIFICATION/VALIDATION.....	203

3.10. LE CYCLE DE VIE D'UN RISQUE	204
RÉFÉRENCES	205

Liste des figures

FIGURE 1 : L'ÉVOLUTION DES PARADIGMES DES SYSTEMES MANUFACTURIERS. (KOREN 2010)	18
FIGURE 2 : COMPARAISON DES PARADIGMES DML, FMS, RMS SELON QUELQUES CRITERES. (LAMECHE ET AL. 2017)	21
FIGURE 3 : EXEMPLE ILLUSTRATIF DU CHANGEMENT DE CONFIGURATIONS D'UN RMS PENDANT SA PHASE OPERATIONNELLE.	23
FIGURE 4 : ORGANISATION DU SYSTEME ET LES RESSOURCES RECONFIGURABLES. (Z.M. BI ET AL. 2008).....	26
FIGURE 5 : LES DE STRUCTURATION D'UNE USINE SELON LA PERSPECTIVE RESSOURCES. (H. P. WIENDAHL ET AL. 2007).....	28
FIGURE 6 : DEMARCHE GENERALE DE CONCEPTION D'UN SYSTEME MANUFACTURIER.	29
FIGURE 7 : LES DIFFERENTS POINTS DE VUE D'UN CYCLE DE VIE D'UN SYSTEME. (HASKINS 2006)	56
FIGURE 8 : CARTOGRAPHIE DES PROCESSUS DU CYCLE DE VIE DU SYSTEME (AFIS 2009).	58
FIGURE 9 : PROCESSUS GENERAL DE MANAGEMENT DE PROJET. (AFIS 2009)	61
FIGURE 10 : EXEMPLE DE PROCESSUS CONTRACTUELS (CAS D'UN SYSTEME UNIQUE). (AFIS 2009).....	62
FIGURE 11 : LES PROCESSUS, LES METHODES ET LES OUTILS EN INGENIERIE SYSTEME. (EIA/ANSI 1999)	65
FIGURE 12 : L'HERITAGE DES NORMES DE L'INGENIERIE SYSTEME.....	66
FIGURE 13 : LA COUVERTURE DU CYCLE DE VIE DU SYSTEME PAR LES 3 NORMES PRINCIPALES D'IS. (AFIS 2009).....	66
FIGURE 14 : LES PROCESSUS DE LA NORME IEEE-1220. (IEEE 2005).....	67
FIGURE 15 : LES PROCESSUS DE LA NORME EIA/ANSI-632. (EIA/ANSI 1999).....	68
FIGURE 16 : LES PROCESSUS DE LA NORME ISO/IEC/IEEE-15288. (ISO/IEC/IEEE15288 2015).....	69
FIGURE 17 : LE MODELE DE DEVELOPPEMENT EN CASCADE SELON ROYCE. (ROYCE 1970).....	71
FIGURE 18 : LE MODELE EN SPIRALE. (BOEHM 1988)	73
FIGURE 19 : LE MODELE EN 'V'. (AFIS 2009)	75
FIGURE 20 : REPRESENTATION DU MODELE EN DOUBLE V. (CLARK 2009)	78
FIGURE 21 : LA TRANSITION DE L'IS VERS L'APPROCHE DIRIGEE PAR LES MODELES (MBSE) (MURRAY 2012).	79
FIGURE 22 : LA FONDATION DE L'OOSEM. (MURRAY 2012)	81
FIGURE 23 : LES ACTIVITES DE L'OOSEM DANS LE CONTEXTE DU PROCESSUS DE DEVELOPPEMENT D'UN SYSTEME. (MURRAY 2012)81	
FIGURE 24 : LES ACTIVITES DE L'OOSEM ET LES ARTEFACTS DE MODELISATION. (MURRAY 2012)	82
FIGURE 25 : LE PROCESSUS HARMONY DE DEVELOPPEMENT DES SYSTEMES ET LOGICIELS INTEGRES. (HOFFMANN 2011)	83
FIGURE 26 : LES ELEMENTS DU PROCESSUS HARMONY-SE. (MURRAY 2012)	84
FIGURE 27 : LES ACTIVITES PRINCIPALES D'IS DE L'APPROCHE MBSE VITECH. (MURRAY 2012).....	86
FIGURE 28 : LA METHODOLOGIE VITECH, ILLUSTRATION DEVELOPPEMENT INCREMENTALE. (MURRAY 2012)	87
FIGURE 29. LE SYSTEME LOGISTIQUE MODULAIRE (EDUCATIF) DLWL-800A. (DOLANG 2017).....	100
FIGURE 30 : CP FACTORY, L'USINE CYBER-PHYSIQUE. (FESTO-DIDACTIC 2017).....	101
FIGURE 31 : LES ETAPES DE LA METHODOLOGIE PROPOSEE POUR CONCEVOIR UN SYSTEME MODULAIRE.	102
FIGURE 32 : IDENTIFICATION DES INTERFACES ENTRE LES FONCTIONS DU SYSTEME.	104
FIGURE 33 : LA MATRICE DSM DU SYSTEME (NON REORDONNEE).....	105
FIGURE 34 : LA MATRICE DSM DU SYSTEME (REORDONNEE).....	106
FIGURE 35 : IDENTIFICATION OU DEDUCTION DES MODULES DU SYSTEME.....	107
FIGURE 36 : SYNOPSIS DU SYSTEME LOGISTIQUE.	108
FIGURE 37 : FLUX ENTRE LES FONCTIONS DU SYSTEME.....	110

FIGURE 38 : LA MATRICE DSM DU SYSTEME (NON REORDONNEE).....	111
FIGURE 39 : LIENS ENTRE LES FONCTIONS DU SYSTEME (APRES SUBSTITUTION ET SIMPLIFICATION).....	112
FIGURE 40 : LA MATRICE DSM APRES LA SUBSTITUTION DES FONCTIONS DU SYSTEME.....	113
FIGURE 41 : FLUX ENTRE LES FONCTIONS DU SYSTEME ET MISE EN EVIDENCE DES MODULES IDENTIFIES.....	115
FIGURE 42 : DEMARCHE GLOBALE D'IMPLEMENTATION DE LA NORME ISO/IEC/IEEE-15288.....	119
FIGURE 43 : CANEVAS DE LA METHODOLOGIE PROPOSEE POUR LA CONCEPTION D'UN RMS.....	120
FIGURE 44 : LA RECONFIGURATION DU SYSTEME SANS EVOLUTION.....	122
FIGURE 45 : LA RECONFIGURATION DU SYSTEME AVEC EVOLUTION.....	123
FIGURE 46 : DEMARCHE GENERALE DE CONCEPTION D'UN SYSTEME MANUFACTURIER.....	125
FIGURE 47 : LES PROCESSUS ET LES ACTIVITES DE LA METHODOLOGIE DE CONCEPTION DES RMS.....	126
FIGURE 48 : EXEMPLE, DIAGRAMME DE CONTEXTE D'UN AGV.....	129
FIGURE 49 : EXEMPLE, DIAGRAMME CAS D'UTILISATION D'UN AGV.....	130
FIGURE 50 : LES LIENS DE TRAÇABILITE POSSIBLES ENTRE EXIGENCES (GENDRE & VIRELY 2013).....	133
FIGURE 51 : POSITIONNEMENT DE LA SIMULATION DANS LA METHODOLOGIE GLOBALE DE CONCEPTION D'UN RMS.....	148
FIGURE 52 : LA METHODOLOGIE GLOBALE POUR LA SIMULATION DES RMSs.....	149
FIGURE 53 : CONCEPT DE MODULARITE DANS LES SYSTEMES MANUFACTURIERS.....	152
FIGURE 54 : PROCESSUS CREATION D'UNE BIBLIOTHEQUE VIRTUELLE.....	153
FIGURE 55 : CLASSIFICATION DES MODELES D'UN SYSTEME.....	155
FIGURE 56 : MODELE GEOMETRIQUE DU SYSTEME.....	157
FIGURE 57 : ILLUSTRATION DU MODELE INFORMATIONNEL DU SYSTEME.....	158
FIGURE 58 : FORME GEOMETRIQUE DU MODULE IDENTIFIE 'BUFFER ENTREE MAGASIN'.....	159
FIGURE 59 : MODELE (VUE GEOMETRIQUE) DU MODULE IDENTIFIE.....	160
FIGURE 60 : LES CONFIGURATIONS POSSIBLES DU MODULE VIRTUEL DEVELOPPE.....	160
FIGURE 61 : INTERFACE GRAPHIQUE UTILISATEUR DU MODULE DEVELOPPE.....	161
FIGURE 62 : LA CONFIGURATION INITIALE DU SYSTEME.....	163
FIGURE 63 : LA NOUVELLE CONFIGURATION DU SYSTEME.....	164
FIGURE 64 : ETAT DE REMPLISSAGE DU 'BUFFER_01', CONFIGURATION INITIALE.....	166
FIGURE 65 : ETAT DE REMPLISSAGE DU 'BUFFER_02', CONFIGURATION INITIALE, CADENCE AUGMENTEE.....	167
FIGURE 66 : ETAT DE REMPLISSAGE DU 'BUFFER_01' APRES LA RECONFIGURATION DU SYSTEME.....	167
FIGURE 67 : ETAT DE REMPLISSAGE DU 'BUFFER_02' APRES LA RECONFIGURATION DU SYSTEME.....	168
FIGURE 68 : IMPLEMENTATION D'UN SYSTEME DE PILOTAGE MODULAIRE.....	169
FIGURE 69 : SITE DE FAURECIA SIEGES AUTOMOBILES A NOGENT-SUR-VERNISSON, HORIZON 2018.....	172
FIGURE 70 : FLUX (PHYSIQUE) : STOCK DE MASSE → ZONE DE PICKING, ET ZONE DE PICKING → MAGASIN DECENTRALISE.....	173
FIGURE 71 : FLUX (PHYSIQUE) : MAGASIN DECENTRALISE → ZONE TPA.....	173
FIGURE 72 : DAHER, PLATE-FORME LOGISTIQUE MARIGNANE, VUE GENERALE.....	174
FIGURE 73 : DAHER, PLATE-FORME LOGISTIQUE MARIGNANE, FLUX MATERIELS.....	175
FIGURE 74 : LES PARTIES PRENANTES DU SYSTEME LOGISTIQUE.....	177
FIGURE 75 : AGV, DIAGRAMME DU CONTEXTE.....	178

FIGURE 76 : AGV, DIAGRAMME DES CAS D'UTILISATION.....	178
FIGURE 77 : AGV, EXEMPLE D'UN DIAGRAMME D'EXIGENCES DE L'AGV.	179
FIGURE 78 : SYSTEME DE PILOTAGE, DIAGRAMME DE CONTEXTE DU SYSTEME DE CONTROLE.	180
FIGURE 79 : SYSTEME DE PILOTAGE, DIAGRAMME DES CAS D'UTILISATION.	181
FIGURE 80 : AGV, DIAGRAMME MACHINE A ETATS, AGV ON-OFF.....	182
FIGURE 81 : AGV, DIAGRAMME MACHINE A ETATS, AGV ON.	182
FIGURE 82 : SYSTEME DE CONTROLE, INTERACTION AVEC L'AGV.....	183
FIGURE 83 : SYSTEME DE CONTROLE, INTERACTION AVEC L'AGV (SUITE FIGURE 82).	184
FIGURE 84 : LA MAQUETTE NUMERIQUE DE FAURECIA, USINE NOGENT-SUR-VERNISSON.	188
FIGURE 85 : LA MAQUETTE NUMERIQUE DE DAHER, PLATE-FORME LOGISTIQUE MARIGNANE.	189
FIGURE 86 : LA MAQUETTE NUMERIQUE DE DAHER, SUITE (ZOOM SUR LA ZONE 1, FIGURE 85).	189
FIGURE 87 : LA MAQUETTE NUMERIQUE DE DAHER, SUITE (ZOOM SUR LA ZONE 2, FIGURE 85).	190
FIGURE 88 : LA CHARGE DE TRAVAIL SUR L'AGV N° 01.	190
FIGURE 89 : L'ETAT DE CHARGE DE LA BATTERIE DE L'AGV N° 01.	191
FIGURE 90 : IMPLEMENTATION DE LA METHODOLOGIE SUR L'OUTIL POLARION, PAGE D'ACCUEIL.....	196
FIGURE 91 : IMPLEMENTATION DE LA METHODOLOGIE SUR L'OUTIL POLARION, PROCESSUS 'DEFINITION DES BESOINS DES PPS'....	197
FIGURE 92 : IMPLEMENTATION DE LA METHODOLOGIE SUR L'OUTIL POLARION, PROCESSUS 'SPECIFICATION DU SYSTEME'.	197
FIGURE 93 : IMPLEMENTATION DE LA METHODOLOGIE SUR L'OUTIL POLARION, PROCESSUS 'CONCEPTION DE L'ARCHITECTURE DU SYSTEME'.....	198
FIGURE 94 : IMPLEMENTATION DE LA METHODOLOGIE SUR L'OUTIL POLARION, PROCESSUS 'VERIFICATION DU SYSTEME'.....	199
FIGURE 95 : IMPLEMENTATION DE LA METHODOLOGIE SUR L'OUTIL POLARION, PROCESSUS 'VALIDATION DU SYSTEME'.....	199
FIGURE 96 : LES ELEMENTS DU TRAVAIL, PARTIE 1.	200
FIGURE 97 : LES ELEMENTS DU TRAVAIL, PARTIE 2.	201
FIGURE 98 : LES ELEMENTS DU TRAVAIL, PARTIE 3.	202
FIGURE 99 : LES ELEMENTS DU TRAVAIL DES SYSTEMES RECONFIGURABLES.	203
FIGURE 100 : CYCLE DE VIE D'UNE EXIGENCE.....	203
FIGURE 101 : CYCLE DE VIE D'UN TEST DE VERIFICATION/VALIDATION.	204
FIGURE 102 : CYCLE DE VIE D'UN RISQUE.	204

Liste des tableaux

TABEAU 1 : LES PARADIGMES DES SYSTEMES MANUFACTURIERS ET LEURS DECLENCHEURS. (KOREN 2010).....	17
TABEAU 2 : LES PHASES DU CYCLE DE VIE D'UN SYSTEME SELON LA NORME. (ISO/IEC/IEEE15288 2015)	55
TABEAU 3 : DEFINITION DES FONCTIONS DU SYSTEME.	103
TABEAU 4 : EXIGENCES FONCTIONNELLES DU SYSTEME.	109
TABEAU 5 : EXEMPLE D'UNE MATRICE DE TRAÇABILITE ENTRE LES PPS ET LES EXIGENCES.	132
TABEAU 6 : EXEMPLE D'UNE MATRICE DE TRAÇABILITE ENTRE LES EXIGENCES.	133
TABEAU 7 : LIENS ENTRE LE CHANGEMENT DE PRODUITS ET LES RECONFIGURATIONS DU SYSTEME.....	136

TABLEAU 8 : EXEMPLE D'UNE MATRICE DE TRAÇABILITE ENTRE LES EXIGENCES ET LES FONCTIONS SYSTEME.....	139
TABLEAU 9 : EXEMPLE D'UNE MATRICE DE TRAÇABILITE ENTRE LES FONCTIONS ET LES COMPOSANTS D'UN SYSTEME.....	140
TABLEAU 10 : EXEMPLE D'UNE MATRICE DE TRAÇABILITE ENTRE LES EXIGENCES ET LES TESTS DE VERIFICATION.	142
TABLEAU 11 : EXEMPLE D'UNE MATRICE DE TRAÇABILITE ENTRE LES EXIGENCES ET LES TESTS DE VALIDATION.	144
TABLEAU 12 : LES MODULES IDENTIFIES DU SYSTEME.	159
TABLEAU 13 : DESCRIPTION DES PARAMETRES DE L'INTERFACE UTILISATEUR REPRESENTEE PAR LA FIGURE 61.....	162
TABLEAU 14 : LES MODULES DE LA BIBLIOTHEQUE VIRTUELLE.....	187

Introduction générale

L'industrie manufacturière a subi plusieurs changements de paradigmes. Ces changements sont induits par : 1) les nouvelles conditions du marché et de l'économie et 2) les besoins sociétaux émergents conduits par les clients. Les besoins sociétaux peuvent résulter de : la volonté d'avoir plus de produits pour satisfaire les goûts et les préférences individuelles, le faible pouvoir d'achat d'une certaine population qui entraîne une baisse des prix des produits, etc.. L'industrie a répondu à ces besoins du marché et de la société en développant de nouveaux types de systèmes manufacturiers. Par exemple, la nécessité sociétale de réduire le coût de l'automobile a été réalisée par l'invention de la chaîne d'assemblage qui, en 1913, était un nouveau type de système de production qui a permis l'introduction du paradigme de production en masse.

L'objectif de chaque paradigme repose sur les nouvelles conditions du marché ou sur les nouveaux besoins de la société. De nouveaux paradigmes deviennent possibles à mesure que de nouveaux outils technologiques sont introduits et par la suite utilisés pour créer de nouveaux types de systèmes de production. Pour chaque nouveau paradigme, un nouveau type de système de production est développé : un système basé sur un nouvel outil technologique et répond aux impératifs du paradigme. Par exemple, l'émergence du paradigme de la personnalisation de masse (mass customisation) a été motivée par la demande de la société pour une variété de produits élargie. La production d'une variété de produits plus large est devenue possible avec l'invention des systèmes flexibles ou les FMSs (Flexible Manufacturing Systems). La nouvelle technologie qui a permis l'émergence de ces systèmes était le mini-ordinateur qui a d'abord été intégré dans les années 1970 aux contrôleurs des machines CNC et à des dispositifs d'automatisation industrielle.

Actuellement, on est dans l'ère de la mondialisation où la production personnalisée est le paradigme de production le plus adapté. Cette mondialisation a créé un nouvel environnement sans précédent pour l'industrie manufacturière, une concurrence féroce, de courtes fenêtres d'opportunité du marché, des introductions fréquentes de nouveaux produits et des changements rapides de la demande. En effet, la mondialisation est un défi, mais elle présente à la fois des menaces et des opportunités. Pour capitaliser sur les opportunités, l'industrie doit offrir des produits novateurs et des systèmes de production réactifs. Le succès dans un tel environnement nécessite une structure d'entreprise qui peut répondre rapidement à l'évolution du marché et des besoins des clients. Cette entreprise devrait être équipée d'un système de production qui peut être rapidement changé et reconfiguré pour répondre à la demande volatile. La reconfiguration du système de production est nécessaire pour pouvoir produire exactement et assez rapidement les quantités et le mix des produits demandés par le marché à un moment donné.

Les systèmes de production reconfigurables ou les RMSs (Reconfigurable Manufacturing Systems) sont le nouveau concept qui a été introduit afin de permettre aux entreprises de rester compétitives dans un environnement caractérisé par des changements fréquents et imprévisibles. Le principal défi d'un RMS est sa conception. La plupart des méthodes proposées dans la littérature ne traitent pas le problème de conception d'un RMS dans son ensemble, mais elles se limitent à une partie du problème. Cela est à cause de la complexité d'un RMS qui rend difficile la mise en place d'une méthode globale, standard et détaillée qui répond à tous les problèmes de conception d'un RMS.

L'objectif de cette thèse est de proposer une méthodologie générique pour la conception des systèmes manufacturiers reconfigurables. Pour atteindre cet objectif, un état de l'art sur les RMSs et sur leurs méthodes de conception est nécessaire, cela est fait dans le chapitre 1. La méthodologie proposée repose beaucoup sur les principes de l'ingénierie des systèmes (IS) et principalement sur la norme ISO/IEC/IEEE-15288. L'IS est une nouvelle discipline pour le développement des systèmes complexes. Le chapitre 2 propose un état de l'art sur cette discipline et sur les méthodes de conception des systèmes complexes. L'objectif de ce chapitre est de proposer les méthodes, les techniques, et les outils qui peuvent être adaptés et utilisés dans la méthodologie de conception des RMS décrite dans cette thèse. Le chapitre 3 traite la modularité, une caractéristique très importante des RMSs, et propose une méthode pour aider à assurer la modularité des RMSs durant le processus de conception. Cette méthode peut être facilement intégrée dans la méthodologie globale de conception des RMSs qui est détaillée dans le chapitre 4.

Le choix d'une configuration adaptée pour faire face aux changements et aux incertitudes de l'environnement de production est difficile. La simulation peut aider à déterminer la meilleure stratégie pour atteindre l'objectif assigné par l'évolution du système de production. Elle est généralement, utilisée pour évaluer et comparer des scénarios définis au préalable et permet également de prédire les conséquences de la modification d'un ou plusieurs composants du système ainsi que le choix d'une stratégie de pilotage plutôt qu'une autre. Donc, un autre objectif de cette thèse est de proposer un outil de simulation de flux pour aider à la gestion (choix de configuration) des RMSs, principalement durant leur phase opérationnelle. Le chapitre 5 détaille l'outil proposé, son développement et son utilisation.

Le chapitre 6 décrit les deux cas d'étude sur lesquels ont été appliqués la méthodologie et l'outil d'aide à décision proposés pour supporter la conception des RMSs. Cette thèse se termine par un ensemble de conclusions et de quelques propositions pour des travaux ultérieurs.

Chapitre 01 : État de l'art sur les systèmes manufacturiers reconfigurables

1. Introduction

Les entreprises manufacturières du 21^{ème} siècle rencontrent de plus en plus de changements imprévisibles et fréquents du marché engendrés par une concurrence mondiale qui se traduit par l'introduction rapide de nouveaux produits et une demande de produits en variation permanente. Pour rester compétitives, les entreprises doivent concevoir des systèmes manufacturiers qui produisent non seulement des produits de haute qualité à faible coût, mais qui permettent aussi une réponse rapide aux changements du marché et aux besoins du client (Koren & Shpitalni 2010). L'examen critique des systèmes de production dédiés ou les DMLs (Dedicated Manufacturing Lines) et les systèmes de production flexibles ou les FMSs (Flexibles manufacturing Systems), montre que ces systèmes ne peuvent pas satisfaire les exigences imposées par le marché qui sont principalement : le coût et la qualité des produits et la réactivité du système manufacturier aux changements du marché. Par conséquent, pour rester compétitives, les entreprises doivent concevoir des systèmes de production qui assurent des produits peu coûteux et de haute qualité avec une réactivité adéquate aux changements du marché. Les systèmes de production reconfigurables ou les RMSs (Reconfigurable Manufacturing Systems) dont les composants sont des machines reconfigurables et des contrôleurs reconfigurables, ainsi que des méthodologies pour leur conception systématique et leur gestion, constituent un nouveau paradigme des systèmes de production proposé comme solution aux besoins actuels. Ces systèmes sont censés être assez réactifs pour faire face aux changements soudains du marché tout en maintenant une haute qualité des produits à faibles coûts.

Il est évident qu'il est difficile d'établir une méthode globale, standard et détaillée qui répond à tous les problèmes de conception des RMSs en raison de leur diversité. La plupart des méthodes proposées dans la littérature ne traitent pas le problème de conception d'un RMS dans son ensemble, mais elles se limitent à une partie du problème. Donc, le principal défi d'un RMS est sa conception. Depuis l'introduction du concept RMS à la fin des années 90 (Koren et al. 1999) (Koren 2005), la recherche dans ce domaine a augmenté et s'est considérablement élargie. Ces recherches sur les RMSs couvrent de multiples problématiques et différents niveaux de structuration d'un système manufacturier (H. P. Wiendahl et al. 2007), du plus haut niveau de structuration (réseau et l'usine) au niveau le plus bas (poste de travail et les outils). Ce chapitre présente un état de l'art sur les RMSs et leurs méthodes de conception.

2. L'évolution des systèmes manufacturiers

Comme tout système fait par l'homme, un système de production est développé pour fournir un ensemble de services avec des performances spécifiées dans un environnement opérationnel défini. Ce

système doit également respecter les contraintes imposées par son environnement telles que les contraintes de sûreté. D'un point de vue externe, un système de production pourrait être défini comme un système qui transforme les matières premières en produits, son but final est de gagner de la valeur, comme le profit, la réputation et des parts du marché (Z.M. Bi et al. 2008). Contrairement, d'un point de vue interne, un système de production pourrait être défini comme une combinaison d'opérateurs, de machines et d'équipements liés par un flux commun de matériaux et d'informations (Chryssolouris 2006). L'environnement des systèmes de production a subi des changements et continue de subir des changements constants plus fréquents et incertains. Cet environnement changeant a forcé les systèmes manufacturiers à évoluer et à s'adapter afin de permettre aux entreprises d'être plus compétitives. Selon Koren (2010), les paradigmes majeurs de la production apparus au cours du temps sont au nombre de quatre : la production artisanale, la production de masse, la personnalisation de masse et la production personnalisée.

2.1. La production artisanale

La production artisanale est définie comme créant exactement le produit demandé par le client, sur demande, et habituellement un produit unique à la fois. Ce paradigme a atteint son zénith à la fois en échelle et en complexité avec la fabrication à la main des voitures. Au début des années 1900, chaque partie d'une automobile a été produite séparément à l'aide de machines-outils à usage général. Le principe de ce paradigme se résume comme suit : les travailleurs qualifiés utilisant des machines à usage général pour fabriquer exactement le produit (un à la fois) que le client a payé. La production artisanale est caractérisée par : de très grande variété de produits, un volume par produit très faible, des machines à usage général pour effectuer toutes les opérations de fabrication, et de la main d'œuvre hautement qualifiée.

2.2. La production de masse

La production de masse, qui a prospéré pendant la majeure partie du 20^{ème} siècle, consiste à produire des quantités extrêmement importantes de produits identiques. Ce paradigme s'exprime par le flux synchronisé des lignes de production qui produisent des composants clés et assemblent le produit fini. Pour maintenir des volumes de production élevés, les machines doivent prendre la place de la compétence humaine autant que possible et les machines sont dédiées à des opérations très spécifiques pour produire le même produit sans modification. Étant donné que des quantités extrêmement importantes sont impliquées, les produits peuvent être produits à faible coût. Réduire le coût de fabrication, et donc le prix du produit, est l'objectif principal de la production de masse. La production de masse est caractérisée par : une faible variété de produits et un volume élevé par produit. Contrairement à la production artisanale qui nécessite une main-d'œuvre hautement qualifiée, la production de masse substitue les machines à la plupart des compétences humaines. Par conséquent, le niveau de compétence moyen de la main-d'œuvre requise pour la production de masse est très faible.

Les principales caractéristiques de la production de masse sont : une variété de produit très limitée, un volume élevé par produit est généré pour réaliser des économies importantes, les machines dédiées et les chaînes d'assemblage qui permettent de réduire les coûts, une main-d'œuvre relativement qualifiée.

2.3. Customisation de masse

La personnalisation de masse a commencé dans les années 80. À mesure que le marché d'un produit mûrit et que les clients deviennent plus riches, ces derniers commencent à chercher une plus grande variété de produits. En réponse à ce besoin, les fabricants commencent à offrir des « options » de produits, chacune comprenant un certain nombre de fonctionnalités supplémentaires qui constituent un « paquet » ajouté à leur produit standard. L'augmentation de la variété de produits à faible coût est l'objectif de ce paradigme. L'introduction des ordinateurs dans les opérations industrielles a rendu possible le développement d'une automatisation flexible qui, à son tour, permet une personnalisation de masse peu coûteuse. La personnalisation de masse ne signifie pas produire des produits uniques, comme dans l'ère de la production artisanale. La personnalisation de masse développe plusieurs ensembles de variations pratiques (options) qui peuvent être produites sur un système de production de masse et offertes aux clients potentiels, dans l'espoir de satisfaire les besoins spécifiques de nombreux clients.

2.4. La globalisation ou la production personnalisée

Le paradigme de production personnalisée promet une valeur optimale au client lorsque le niveau de satisfaction et le prix du produit sont échangés. Dans ce paradigme, les clients sont impliqués dans la conception de leur produit, tandis que dans la personnalisation de masse, les clients ne peuvent que sélectionner la meilleure solution pour répondre à leurs besoins.

2.5. Changement de paradigme

Pour résumer, nous avons discuté quatre paradigmes (voir le tableau suivant).

Paradigme	Production artisanale	Production de masse	Customisation de masse	Production personnalisée
Concentration	L'individu	Le produit	Les parts du marché	L'individu
Les besoins sociétaux	Des produits personnalisés à la demande	Produits à faibles coûts	Large variété des produits	Des produits personnalisés
Modèle commercial	Vendre → Concevoir → Fabriquer	Concevoir → Produire → Vendre	Concevoir → Vendre → Fabriquer	Vendre → Concevoir → Fabriquer

Tableau 1 : les paradigmes des systèmes manufacturiers et leurs déclencheurs. (Koren 2010)

- Production artisanale, dans laquelle chaque produit est conçu et fabriqué pour un client particulier. Le client initialise la séquence Vendre-Concevoir-Fabriquer et pilote la conception.

- Production de masse, dans laquelle seuls quelques modèles de produits sont fabriqués, en supposant qu'il y aura toujours suffisamment de clients pour les acheter. Le client se trouve à la fin de la séquence (Concevoir-Fabriquer-Vendre).
- Personnalisation de masse, dans laquelle les clients sélectionnent un produit à partir d'une liste d'options disponibles avant la production. Le fabricant prend les principales décisions stratégiques concernant l'architecture de base du produit ainsi que le nombre de variantes et d'options offertes en fonction des groupes de clients ciblés. Le client ne peut sélectionner que l'option qui correspond le mieux à ses préférences et à son prix.
- Production personnalisée, dans laquelle les options des produits sont conçues par les clients, vendues, puis produites sur des systèmes de fabrication avancés. On promet aux clients la meilleure adéquation à leurs besoins avec le meilleur rapport coût-efficacité. Il est le plus proche du paradigme de production artisanale.

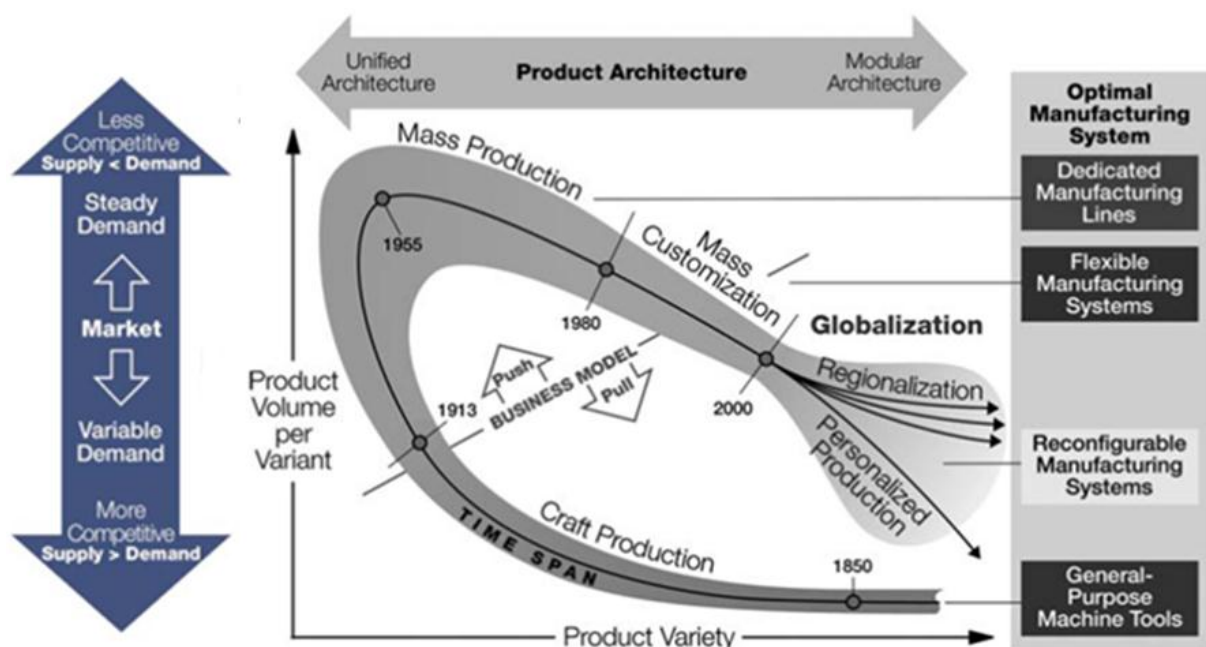


Figure 1 : l'évolution des paradigmes des systèmes manufacturiers. (Koren 2010)

Généralement, de nouveaux paradigmes de fabrication sont établis par des besoins sociétaux émergents ou par de nouvelles conditions du marché. Un modèle illustratif des quatre paradigmes pour les automobiles est illustré par la Figure 1. Les années 1850, 1913 et 1955 sur la ligne du temps dans cette figure correspondent aux changements de paradigme dans la production automobile. Depuis le milieu des années 1950, la tendance de la variété du produit par rapport au volume est vers une plus grande variété et un plus petit volume par variante, une tendance qui est applicable non seulement aux automobiles, mais aussi à de nombreux biens de consommation, tels que les appareils ménagers, le mobilier de bureau, etc. Le graphique (Figure 1) montre comment les changements dans le marché et les besoins des consommateurs poussent au changement de paradigme. La mondialisation stimule

actuellement la production personnalisée, dans laquelle la société exige une plus grande variété et de plus petits volumes par variante de produit. Les fluctuations du marché, qui augmentent chaque jour, obligent les fabricants à produire des volumes encore plus petits pour chaque variante de produit.

2.6. Le paradigme et le type du système de production

La Figure 1 montre la relation entre le marché et les besoins sociétaux (par exemple, les produits hétérogènes) d'une part, et les systèmes de production qui fonctionnent le mieux pour répondre à ces conditions, d'autre part. Les défis de chaque nouveau paradigme ont toujours été satisfaits par un nouveau type de système de production, rendu possible par l'application d'un nouvel outil technologique. Par exemple, comme le montre la Figure 1, le paradigme de personnalisation de masse a été réalisé en utilisant les FMSs, car cette flexibilité de fabrication a permis une production rentable d'une variété de produits. Et les mini-ordinateurs (qui étaient une nouvelle technologie dans les années 1970) ont permis la création des FMSs et de leurs éléments de base qui sont les machines CNC et des robots industriels. Dans le paradigme de la production artisanale, des machines-outils polyvalentes ont été utilisées. Ils ont été remplacés dans le paradigme de la production de masse par les DMLs qui utilisaient l'automatisation fixe pour fabriquer des produits et des pièces en très grandes quantités. Les FMSs ne correspondent pas à des marchés instables avec des fluctuations de la demande de produits comme nous le voyons à l'ère de la mondialisation. La réponse à cette nouvelle situation est les RMSs dont leur capacité de production (c'est-à-dire le volume par produit) peut être ajustée rapidement pour répondre à la demande du marché. Ils peuvent être rapidement outillés pour produire de nouveaux produits et peuvent être améliorés avec de nouvelles fonctionnalités pour produire différentes variétés du produit. Les RMSs améliorent la réactivité de l'entreprise aux nouvelles conditions du marché et lui donnent un avantage concurrentiel.

2.6.1. Systèmes dédiés et les systèmes flexibles

Dans les systèmes de production classiques on trouve deux concepts : les systèmes de production dédiés appelés aussi les DMLs (Dedicated Manufacturing Systems) et les systèmes de production flexibles appelés aussi les FMSs (Flexible Manufacturing Systems). Un DML est conçu pour produire un produit spécifique en grande quantité, donc, tant que la demande est importante le système est rentable. Le concept FMS a été introduit comme réponse aux demandes du marché pour un système manufacturier flexible et réactif. Un FMS est caractérisé par une flexibilité générale qui permet de produire une grande variété de produits, mais qu'est généralement pas utilisée totalement. Un FMS n'a pas été adopté à grande échelle à cause de son faible débit de production, de sa complexité et le coût élevé de la flexibilité générale qu'il offre (Joergensen et al. 2010). Les produits fabriqués sur des DMLs sont de haute qualité, car le procédé est répétitif et bien contrôlé. Le coût des produits est très bas tant que la demande est très élevée ; ce n'est généralement pas le cas dans le marché actuel où les produits sont demandés en petites quantités (petits lots). Le système est conçu spécifiquement pour un

seul produit ou pour un ensemble réduit de produits, donc le système n'a aucune flexibilité pour faire face aux changements du marché. Ainsi, changer la variété (gamme de produits) du système ne serait pas facile, pour chaque nouveau produit, une toute nouvelle ligne, de production ou d'assemblage, dédiée doit être conçue et déployée. De la même manière, tout changement dans le volume de production ne serait pas réalisé graduellement par de petits pas ; une nouvelle ligne doit être déployée. Par conséquent, les DMLs sont très coûteux lorsqu'il s'agit de changements. (Lameche et al. 2017b)

Contrairement aux DMLs où chaque machine effectue quelques opérations simples, les FMSs utilisent des machines capables d'effectuer diverses opérations et, par conséquent, peuvent produire une large gamme de produits différents. Les FMSs, cependant, ne sont rentables ou utiles que lorsque des volumes de produits relativement faibles sont nécessaires, car ils sont lents et coûteux par rapport aux DMLs. Les systèmes flexibles sont particulièrement coûteux à cause de flexibilité générale qui est difficile et coûteuse à la construire et à l'entretenir. Selon (Koren 2010), les FMSs sont coûteux dans le sens où les entreprises achètent généralement des machines avec plus de fonctionnalités dont elles ont vraiment besoin. Ainsi, la flexibilité et la fonctionnalité supplémentaires qu'un FMS peut offrir sont dans de nombreux cas un gaspillage de ressources. Malgré leur flexibilité, les FMSs ne sont pas suffisamment réactifs pour faire face aux changements rapides du marché. Le délai de mise sur le marché est très long. Les produits sont très chers, car le client doit payer pour la flexibilité générale (non personnalisée) (ElMaraghy 2007). Le changement de la productivité ne peut se faire que par des petites étapes, et il est très coûteux, car de nouvelles machines, généralement des machines CNC, doivent être déployées. (Source, papier IJPR)

Cette brève revue critique des systèmes de production existants, en ce qui concerne les exigences du marché actuel, révèle qu'il existe un besoin fort d'un nouveau concept de systèmes de production. Cela a conduit à l'introduction de systèmes de production reconfigurables par Koren dans les années 90 (Koren et al. 1999). Un RMS combine le haut débit d'un DML, la flexibilité d'un FMS et la capacité de répondre efficacement à l'évolution du marché. Le diagramme représenté par la Figure 2 montre une comparaison entre les trois paradigmes manufacturiers (DML, FMS et RMS) selon plusieurs critères.

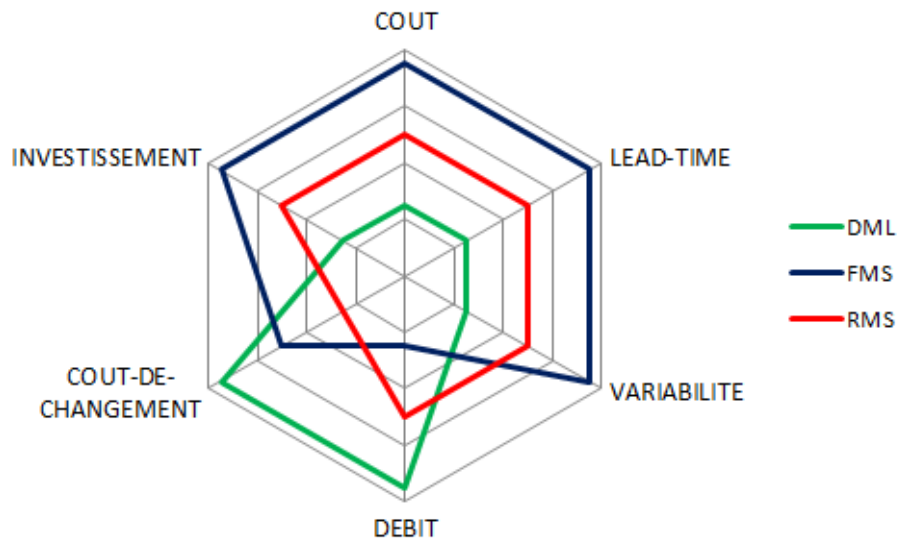


Figure 2 : comparaison des paradigmes DML, FMS, RMS selon quelques critères. (Lameche et al. 2017)

3. Le concept RMS

3.1. Définition d'un RMS

D'après (Koren et al. 1999; ElMaraghy 2007; Mehrabi et al. 2002; Mehrabi, Ulsoy, and Koren 2000), un RMS est un système de production dont les structures physiques et logiques à tous les niveaux de composition du système peuvent être changées rapidement et à moindre coût afin d'ajuster la capacité de production et la fonctionnalité autour d'une famille de produits en réponse à des changements soudains du marché. La composition du RMS est basée sur des modules physiques et logiques standards (cellules, machines, éléments de machines, équipements de manutention, équipements de stockage, contrôleurs, outils, fixations, etc.) pouvant être ajoutés, modifiés, réarrangés, permutés ou remplacés rapidement et de manière fiable selon le besoin pour répondre à l'évolution des exigences du marché. L'emplacement de ces modules peut également être modifié, au niveau de l'atelier, pour réaliser la configuration qui répond le mieux aux exigences du contexte de production. Un RMS offre une flexibilité personnalisée pour une famille de produits particulière, et il possèdera une architecture ouverte qui pourra être améliorée, mise à jour et reconfigurée plutôt que d'être remplacée (Koren 2005) (Koren et al. 1999).

Selon (Mehrabi et al. 2000a) (Mehrabi et al. 2000b) (Mehrabi et al. 2002), un RMS est un système de production qui peut être créé en incorporant des modules (physiques et logiques) de base qui peuvent être réarrangés ou remplacés rapidement et de manière fiable. La reconfiguration autorise l'ajout, la suppression ou la modification des capacités d'un module spécifique, d'un système de contrôle, ou de la structure d'une machine pour ajuster la capacité de production en réponse à l'évolution des demandes du marché ou des technologies. Ce type de système fournira la flexibilité

personnalisée pour une famille de produits particulière, et sera à architecture ouverte de sorte qu'il peut être amélioré, mis à jour, et reconfiguré, plutôt que d'être remplacé.

3.2. Les caractéristiques d'un RMS

Un RMS possède certaines caractéristiques clés qui permettent un haut degré de réactivité du système aux besoins du marché. D'après (Koren 2005) (Koren & Shpitalni 2010) (Koren & Ulsoy 2002) (Rösiö 2012b), ces caractéristiques, au nombre de six, devraient être intégrées dans le système reconfigurable dès la phase de conception pour assurer un degré élevé de reconfigurabilité. Ces caractéristiques sont :

- La personnalisation (flexibilité limitée à une famille de produits) : La flexibilité d'un système ou d'une machine limitée à une famille de produits, donc une flexibilité personnalisée pour cette famille de produits.
- La convertibilité (conception pour des modifications de la fonctionnalité) : La capacité de transformer facilement la fonctionnalité du système et des machines existantes pour répondre aux nouvelles exigences de production.
- L'évolutivité (conception pour des changements de capacité) : La possibilité de modifier facilement la capacité de production en ajoutant ou soustrayant des ressources de production (par exemple des machines) et/ou changement des composantes du système.
- La modularité (les composants sont modulaires) : La classification des fonctions opérationnelles en unités qui peuvent être manipulées et échangées entre les systèmes de production pour un arrangement optimal.
- L'intégrabilité (interfaces simples pour une intégration rapide) : La capacité d'intégrer des modules rapidement et précisément à l'aide d'un ensemble d'interfaces mécaniques, informationnelles, et des interfaces de contrôle qui facilitent l'intégration et la communication.
- La diagnosticabilité (conception pour des diagnostics faciles) : La capacité de lire automatiquement l'état actuel du système afin de détecter et diagnostiquer les causes des produits défectueux en sortie, et de les corriger rapidement.

Des caractéristiques supplémentaires telles que la mobilité des différentes ressources (production, stockage et manutention, etc.) au niveau de l'atelier sont mentionnées dans d'autres travaux tels que (Rösiö 2012b). Selon (Koren & Shpitalni 2010), la personnalisation, la convertibilité et l'évolutivité sont des caractéristiques critiques ou importantes pour la reconfiguration. Par contre, la modularité, l'intégrabilité et la diagnosticabilité ne garantissent pas des modifications de la capacité et la fonctionnalité de production, mais ils permettent une reconfiguration rapide (réaction rapide) aux : occurrences externes imprévisibles (par exemple, les changements du marché), changements du modèle du produit planifié, événements intrinsèques inattendus du système (par exemple une

défaillance de longue durée inattendue d'une machine). Ces caractéristiques devraient être considérées comme des exigences principales de conception.

3.3. La configuration et la reconfiguration du RMS

La Figure 3 illustre le changement de configuration d'un RMS, pendant sa phase opérationnelle, à cause de l'apparition d'événements déclencheurs de la reconfiguration ou DRs. Elle montre également la nécessité d'une phase de reconfiguration pour pouvoir passer d'une configuration à l'autre. Après son déploiement, le système est dans sa configuration initiale. Cette configuration doit être modifiée lorsqu'il est nécessaire. Le processus requis pour pouvoir passer d'une configuration à une autre est appelé processus de reconfiguration.

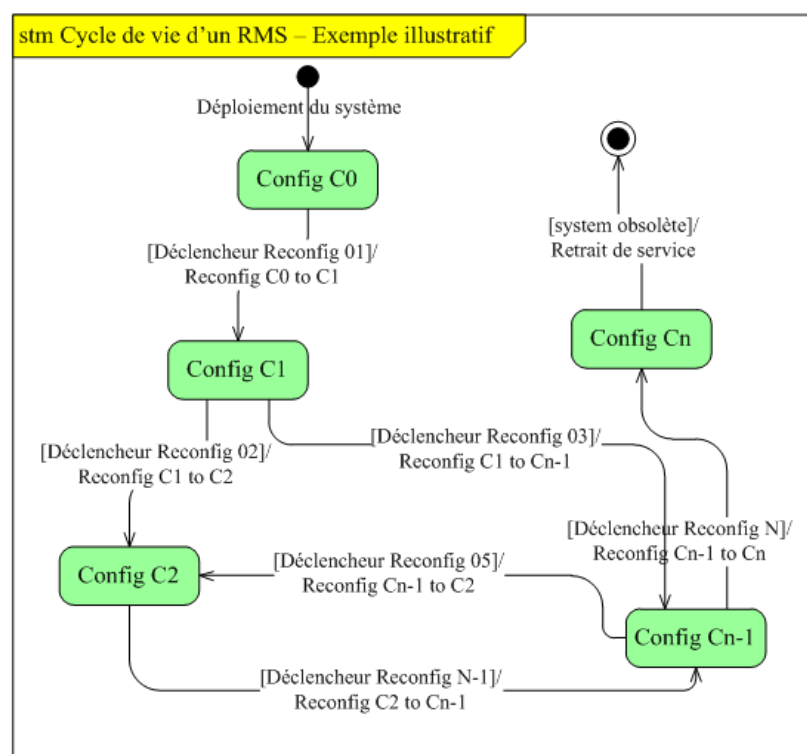


Figure 3 : exemple illustratif du changement de configurations d'un RMS pendant sa phase opérationnelle.

3.3.1. La configuration du système

Une configuration est un état du système qui répond à un contexte particulier. Pour définir la configuration du système, on doit définir un ensemble de paramètres du système qui déterminent cette configuration. Le changement d'un paramètre implique le changement de la configuration du système. Le nombre de machines, le nombre des lignes d'assemblage, le nombre de stations de chargement, le nombre de stations de déchargement, le nombre d'AGVs dans l'atelier, le nombre d'opérateurs, le positionnement de ces objets dans l'atelier, les règles de gestion de la flotte d'AGVs, etc. sont quelques exemples de paramètres de configuration.

3.3.2. Les déclencheurs de la reconfiguration

Un évènement déclencheur de la reconfiguration DR est un évènement dont l'occurrence nécessite la reconfiguration du système. Par exemple, le changement de produit est un DR qui nécessite la reconfiguration du système pour pouvoir fabriquer le nouveau produit. Un autre exemple d'un DR est l'augmentation de la demande. L'augmentation de celle-ci nécessite la reconfiguration du système (ajouter des machines ou des lignes d'assemblage par exemple) pour pouvoir répondre à la nouvelle demande du client. Généralement, il y a deux types d'DRs qui obligent la reconfiguration du système manufacturier :

- **Les DRs extrinsèques au système** : ils dépendent du contexte du système manufacturier et ils n'ont rien à voir le système lui-même. On peut citer comme DR extrinsèque, les variations de la demande du client (augmentée, ou diminuée), les changements de variétés de la demande du client (au début, les produits A et B sont demandés par le client, puis les produits B et C sont demandés). On peut aussi avoir une demande d'un nouveau produit qui n'existe pas encore dans le catalogue du fournisseur, mais qu'il pourrait être fabriqué par le système manufacturier moyennement une reconfiguration. Ou encore l'introduction d'une nouvelle technologie qui n'est pas utilisée dans le système manufacturier, mais qui pourrait être intégrée dans le système.
- **Les DRs intrinsèques au système** : généralement, ce sont les pannes. Si une panne arrive, le système grâce à sa caractéristique de reconfiguration pourrait contourner cette panne et continuer à satisfaire la demande du client.

Donc, si un DR que ce soit extrinsèque ou intrinsèque arrive, le système doit être reconfiguré pour qu'il puisse faire face aux changements de contexte provoqués par l'occurrence de cet évènement. La reconfiguration du système nécessite généralement l'intervention d'opérateurs humains pour faire les changements nécessaires, par exemple ajouter une machine, ajouter une ligne d'assemblage, changer l'outil au niveau de la machine X ou le robot Y, changer les paramètres du programme de la machine X, changer le moyen de transport entre tel et tel poste, changer le support de transport, etc. Mais parfois, la reconfiguration peut se faire automatiquement si les changements nécessaires ne sont pas conséquents et si le système est doté d'une intelligence ou d'une capacité qui lui permet de détecter le besoin de se reconfigurer et les changements à faire. Par exemple, la recalibration d'une machine ou l'augmentation de la cadence peuvent se faire automatiquement.

3.3.3. La reconfiguration du système

Le changement de l'environnement impose au système de production de s'adapter en adoptant une nouvelle configuration qui répond le mieux aux nouvelles exigences de l'environnement et les contraintes imposées. Le passage d'une configuration à l'autre doit être facile et rapide pour assurer l'agilité du système. Le processus de changement de configuration est appelé la reconfiguration et il

est important de disposer des méthodes et outils efficaces pour que ce processus soit facile et rapide afin d'assurer l'agilité du système de production. Donc, la reconfiguration est le processus qui permet le passage du système d'une configuration à une autre et il nécessite généralement des moyens et du personnel pour être exécuté.

En général, les reconfigurations peuvent être divisées en deux catégories : physiques et logiques. (Andersen et al. 2015). La reconfiguration ne se limite pas à la partie physique du système (changer l'outil d'une machine par exemple), mais elle concerne aussi la partie logique (changer le programme de la machine pour utiliser le nouvel outil ajouté). Elle nécessite des modifications physiques et logiques. La reconfiguration logique/logicielle est souvent déclenchée par la reconfiguration physique du système. Selon (Elmaraghy 2007), la reconfiguration physique inclut la modification de la structure du système ainsi que des modifications concernant les cellules, les machines, les équipements de manutention, les équipements de stockage et tous les modules physiques qui composent le système. La reconfiguration logicielle inclut la programmation des contrôleurs, la replanification et le réordonnancement de la production, la formation des opérateurs, etc. D'après (Andersen et al. 2015), les reconfigurations physiques impliquent des changements physiques dans les équipements et la disposition des machines, ce qui nécessite généralement des changements sur les niveaux de structuration inférieurs, tels que les postes de travail ou les cellules. D'un autre côté, les reconfigurations logiques impliquent des changements logiques tels que le réordonnancement et la replanification et sont principalement liées à des problématiques de recherche au niveau du système ou à des niveaux plus élevés.

La reconfiguration du système ne concerne pas le système dans sa globalité seulement, mais elle concerne tous ses niveaux de décomposition : le système, les sous-systèmes, et les composants du système. La reconfiguration peut être classée en trois catégories (légère, moyenne, forte) selon les coûts et les efforts nécessaires pour la réaliser. Ajouter une ligne entière d'assemblage dans une usine par exemple nécessite plus de coûts et d'efforts que de changer l'outil d'une machine. D'après (Z.M. Bi et al. 2008), la reconfigurabilité d'un système de production peut être classée en fonction des niveaux où les mesures de reconfigurabilité sont prises. Comme le montre la Figure 4, la reconfigurabilité à des niveaux inférieurs est principalement achevée en changeant les ressources physiques (matérielles), et la reconfiguration à des niveaux plus élevés est principalement réalisée en changeant les ressources logiques (logicielles) et/ou par le choix des méthodes alternatives ou des structures d'organisation flexibles.

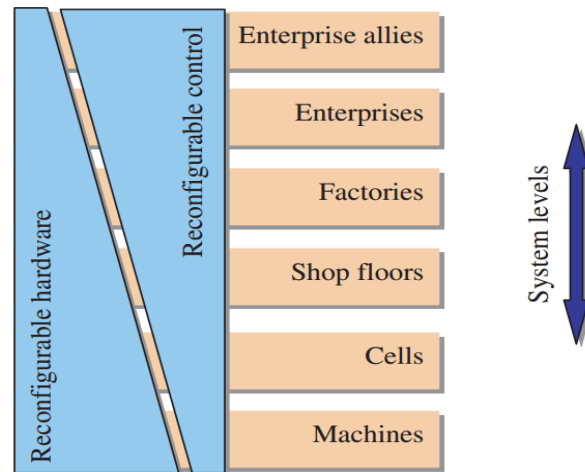


Figure 4 : organisation du système et les ressources reconfigurables. (Z.M. Bi et al. 2008)

La reconfigurabilité doit être incluse dans tous les niveaux du système de production et dans les deux parties : partie physique (matérielle) et partie logique (logicielle). Donc, le système, les sous-systèmes et leurs composants doivent être conçus dès le départ pour être rapidement reconfigurables et à moindre coût en utilisant des modules matériels (physiques) et logiciels (logiques) de base qui peuvent être intégrés rapidement grâce à l'utilisation d'interfaces conçues.

Habituellement, les méthodes de conception conventionnelles se terminent avec l'implémentation et le démarrage du fonctionnement opérationnel du système. Cependant, dans le cas d'un système RMS, des reconfigurations et des reconceptions continues sont nécessaires pour prolonger la durée de vie du système. Cette phase de reconfiguration est souvent décrite comme étant une répétition de certaines étapes dans de la méthode de conception, par ex. revenir aux opérations de regroupement (Tracht & Hogreve 2012) ou revenir en arrière et détailler davantage la conception de certaines configurations conceptuelles du système (Deif & ElMaraghy 2006), ce qui se traduit par des étapes répétées concernant la conception détaillée.

Dans la littérature, on ne trouve pas beaucoup de travaux qui traitent la problématique de reconfiguration du système. On peut citer Azab et al. (2013) qui proposent une approche sous forme de boucle de contrôle pour faire les changements nécessaires afin de répondre aux changements de l'environnement. L'approche fait le lien entre les exigences du changement et les caractéristiques clés qui permettent ou facilitent ce changement. 3 niveaux de changement sont identifiés : la flexibilité : si la flexibilité de la configuration actuelle du système permet de répondre aux nouvelles exigences. La reconfigurabilité au niveau de la machine : si la flexibilité de la configuration actuelle du système n'est pas capable de répondre aux nouvelles exigences et la reconfiguration au niveau de la machine est suffisante pour remplir les exigences. La reconfiguration globale du système : si ni la flexibilité ni la reconfiguration au niveau machine est suffisante, une reconfiguration du système globale est nécessaire pour satisfaire les nouvelles exigences). Les auteurs ont proposé un organigramme qui

détaille le mécanisme du changement afin de répondre aux nouvelles exigences imposées au système manufacturier. [Youssef & ElMaraghy \(2007\)](#) et [Goyal et al. \(2012\)](#) ont suggéré le développement des modèles afin d'optimiser le processus de reconfiguration du système. Dans ([Youssef & ElMaraghy 2006](#)), une métrique de régularité de ce processus a été appliquée dans le but d'évaluer le temps et l'effort requis associés à une reconfiguration. Leur travail a souligné le besoin de considérations pratiques supplémentaires dans la planification d'une reconfiguration.

4. La conception des RMSs

Qu'il s'agisse de développer de nouveaux systèmes reconfigurables ou d'incorporer la reconfigurabilité dans les systèmes existants, les exigences posées aux RMSs peuvent changer de manière significative pendant sa durée de vie, cela rend la conception d'un RMS plus difficile que celle d'un système de production conventionnel. ([Rösiö 2012b](#)) ([Tracht & Hogreve 2012](#)). Pour la conception d'un RMS, [Rösiö & Jackson \(2009\)](#) mentionnent qu'il est nécessaire d'appliquer une vision à long terme du système de production, afin de garantir la faisabilité économique de plusieurs générations de produits et de situations de marché. En d'autres termes, le système devrait être conçu pour être modifiable en termes de fonctionnalité et de capacité.

Les systèmes manufacturiers sont des systèmes complexes, et la reconfigurabilité apporte une complexité supplémentaire à ces systèmes. Donc, la problématique de conception des RMSs est très large, et elle ne peut pas être entièrement traitée dans cette thèse. En conséquence, il est important de définir les limites de cette étude avant d'exposer les barrières à surmonter lors de la conception des RMSs. Dans ([H. P. Wiendahl et al. 2007](#)), l'organisation d'un système manufacturier a été divisée en six niveaux selon deux perspectives différentes, à savoir la perspective ressources et la perspective espace. Il est recommandé que la reconfigurabilité soit étendue à l'ensemble de l'usine (tous les niveaux de composition). Nous n'aborderons pas la conception de systèmes manufacturiers reconfigurables à tous les niveaux de composition, notre travail se limitera au niveau de l'atelier et à tous les niveaux de composition inférieurs du système (voir Figure 5).

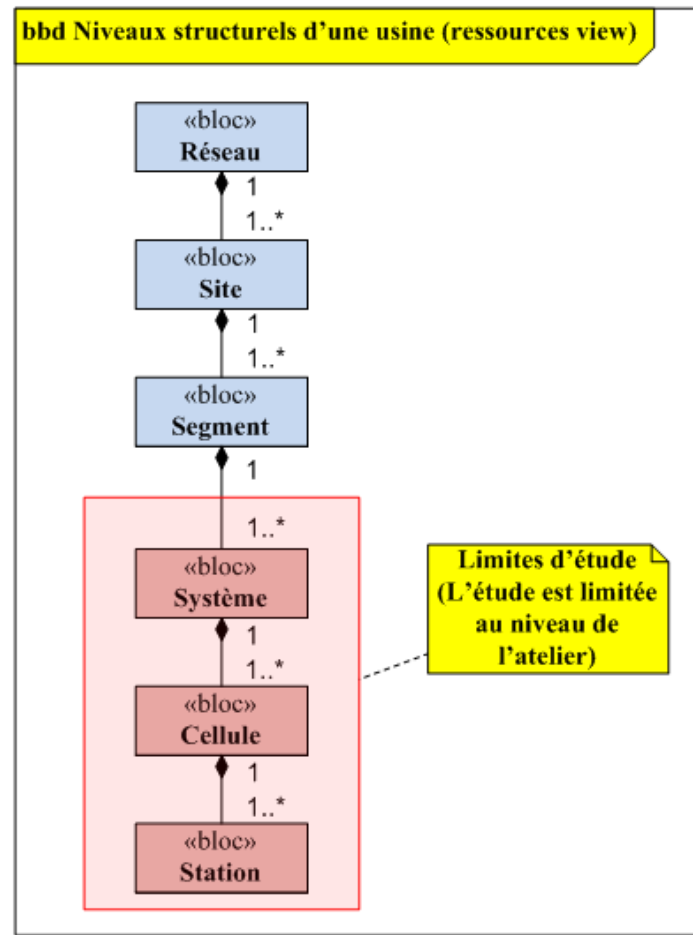


Figure 5 : les de structuration d'une usine selon la perspective ressources. (H. P. Wiendahl et al. 2007)

La Figure 6 présente un aperçu général du processus de conception d'un système de production. Tout système de production, y compris un RMS, est destiné à fabriquer un ensemble de produits. Donc, la première étape consiste à définir ces produits. Ensuite, le processus de fabrication de chaque produit doit être détaillé. Il y a un lien très fort et une influence mutuelle entre le produit et son processus de fabrication. Une fois que le processus de fabrication de chaque produit est établi, on commence la troisième phase qui est la conception du système de production. Comme nous l'avons déjà dit, notre étude se limitera à la conception du système de production au niveau de l'atelier. Ainsi, la méthodologie proposée dans cette thèse (voir chapitre 4) pour concevoir les RMSs sera limitée à ce niveau. Par conséquent, les produits et leurs conceptions ainsi que leurs processus de fabrication détaillés sont supposés être donnés ou déjà réalisés.

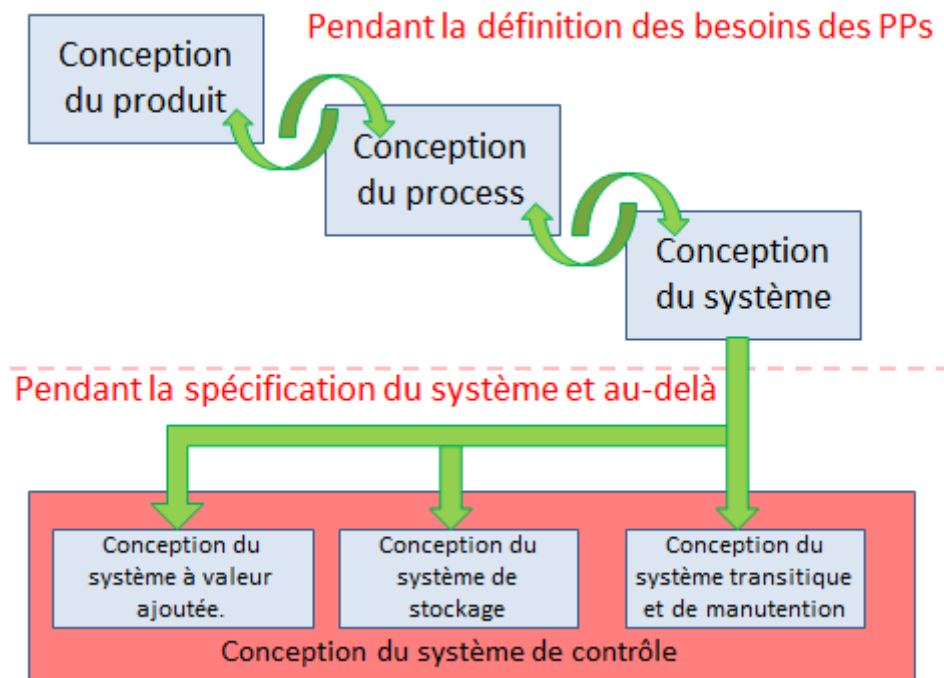


Figure 6 : démarche générale de conception d'un système manufacturier.

Au cours des dernières années, de nombreux travaux ont été réalisés pour atteindre l'objectif ultime qui est de créer une « usine dynamique » capable d'adapter rapidement sa capacité de production et sa variété tout en maintenant un haut niveau de qualité à un coût abordable. Un examen critique du travail effectué dans ce domaine (c'est-à-dire, la conception des RMSs) est nécessaire afin d'avoir une idée générale. Dans la littérature, les solutions présentées pour résoudre le problème de conception des RMSs sont nombreuses, et elles traitent le problème selon différentes perspectives avec différents niveaux de détails. Pour une meilleure compréhension, ces travaux seront exposés suivant le processus global (voir Figure 6) de conception d'un système de production. Pour chaque étape de ce processus de conception, un ensemble de travaux trouvés dans la littérature sera exposé et discuté brièvement.

4.1. Problématique de conception des RMSs

Les RMSs sont des systèmes complexes et la reconfigurabilité apporte plus de complexité, cela rend leur conception plus compliquée que celle des systèmes de production conventionnels. Des recherches antérieures indiquent que les méthodes de conception des systèmes de production classiques ne peuvent pas supporter la conception des RMSs, car, principalement, elles ne tiennent pas compte des exigences de la reconfigurabilité (Rösiö 2012b) (Rösiö & Säfsten 2013). De nombreux auteurs indiquent qu'il manque une méthode de conception systématique des RMSs (Z.M. Bi et al. 2008) (El Maraghy 2006) (Malhotra et al. 2012) (Malhotra et al. 2009) (Malhotra et al. 2010) (Rösiö & Säfsten 2013). Par exemple, Bi et al. (2008) présentent une revue des méthodes existantes pour la conception de l'architecture du système, de sa configuration, et de la conception de son système de

contrôle. Ils arrivent à la conclusion qu'il manque une méthode de conception systématique. De plus, dans la recherche actuelle, l'implémentation réussie des RMSs n'a pas encore été largement rapportée. Certains chercheurs ont étudié les attitudes et les barrières de la reconfigurabilité dans l'industrie et ont conclu que diverses idées fausses et une méconnaissance générale de la reconfigurabilité sont présentes (Rösiö & Säfsten 2013) (Heisel & Meitzner 2006) et que diverses barrières critiques existent quant à son implémentation (Malhotra et al. 2012), (Andersen et al. 2016) (Andersen et al. 2017)

D'après (Rösiö 2012b), il y a une focalisation sur les méthodes et techniques pour parvenir à la reconfigurabilité, mais une perspective générale et complète de la reconfigurabilité est rarement donnée. Par conséquent, l'importance du développement d'une méthode de conception systématique d'un RMS devrait être soulignée, car elle fournit les connaissances nécessaires sur la façon de développer et de réaliser les avantages de la reconfigurabilité.

4.2. Principes de conception des RMSs

Selon (Koren & Shpitalni 2010), les RMSs doivent être conçus selon les principes de la reconfiguration. Plus ces principes sont applicables à un système de production, plus ce système est reconfigurable. L'implémentation de ces principes dans la conception des RMSs conduit à la réalisation de l'objectif finale qui consiste à créer une usine dynamique qui peut rapidement adapter sa capacité de production tout en maintenant des niveaux élevés de qualité des produits. Ces principes sont :

- Un RMS doit fournir des ressources de production réglables pour répondre aux changements imprévisibles du marché et les événements intrinsèques du système.
- Un RMS doit être conçu autour d'une famille de produits, avec juste suffisamment de flexibilité personnalisée pour produire tous les membres de la famille.
- Les caractéristiques de base d'un RMS (voir chapitre 1, section 3.2) devraient être intégrées dans le système dans son ensemble, ainsi que dans ses composants (physiques et logiques).
- La capacité d'un RMS doit pouvoir être rapidement ajustée (incrémenté ou décrétementé) par petits incréments.
- La fonctionnalité d'un RMS doit pouvoir être adaptée rapidement à de nouveaux produits.
- Les capacités d'ajustement intégrées d'un RMS doivent faciliter une réponse rapide à des défaillances imprévues du matériel.

4.3. Conception des produits et formations des familles de produits

Un RMS est conçu autour d'une famille de produits avec suffisamment de flexibilité pour pouvoir les fabriquer tous. Ces produits sont regroupés en familles en fonction de certaines caractéristiques partagées telles que la modularité, la séquence d'opérations de fabrication, etc. Chaque famille de

produits nécessite une configuration du système permettant de les produire, et le passage d'un produit à l'autre ou plus généralement d'une famille de produits à l'autre nécessite la reconfiguration du système. D'après (Koren & Shpitalni 2010), les RMSs constituent une nouvelle classe de systèmes caractérisés par une structure ajustable et une conception autour d'une famille de produits ou de pièces. La conception autour d'une famille de produits au lieu d'un seul produit en particulier permet aux concepteurs de planifier un système qui supporte différentes variantes de la même famille de produits avec un minimum de modifications du système de production. La conception d'un RMS est fortement liée au portefeuille de produits, car le système devrait être capable de produire plusieurs variantes de produits au sein d'une famille de produits, mais aussi différentes générations de la même famille de produits (Rösiö & Jackson 2009) (Andersen et al. 2017) (Goyal et al. 2013). Donc, l'efficacité d'un RMS dépend de la formation de la meilleure série de familles de produits.

La formation des familles de produits a été traitée pendant plus d'une décennie tant dans la recherche que dans la pratique (Roger Jiao et al. 2007). La définition de ce qui constitue une famille de produits peut être réalisée de différentes manières (Abdi & Labib 2004b). Cependant, comme le RMS est construit autour de familles de produits pour réduire la flexibilité et augmenter l'efficacité, il est essentiel de concevoir des familles de produits en même temps que le système de production (ElMaraghy et al. 2013). À cet égard, trois préoccupations peuvent être identifiées dans la littérature ; regrouper les produits dans des familles de produits, concevoir la configuration optimale correspondante et adapter les configurations du système aux familles de produits en évolution (Bryan et al. 2007) (AlGeddawy & ElMaraghy 2011). La littérature propose plusieurs approches pour répondre à ces préoccupations.

Dans (Kashkoush & ElMaraghy 2014), il a été souligné que la formation appropriée des familles de produits est d'une grande importance pour un RMS rentable et productif, et il a été mentionné aussi dans (Koren and Shpitalni 2010) qu'il est nécessaire d'avoir une méthodologie pour pouvoir regrouper les produits dans des familles. Le problème de regroupement des produits en familles selon quelques paramètres ou caractéristiques de similitude n'est pas apparu avec l'introduction des RMSs. Dans (ElMaraghy 2007), il a été mentionné que cette question était déjà abordée pour la première fois en conjonction avec le concept «technologie de groupe» ou Group Technology (GT) qui est une philosophie de production dans laquelle des produits/pièces similaires sont identifiés et regroupés en familles afin d'en profiter de leurs similitudes dans la conception et la production.

En plus des méthodes, pour regrouper les produits en familles, développées avant l'apparition du concept RMS, des techniques et méthodes récentes ont été mises en place en considérant différents critères de regroupement. Dans (Kashkoush & ElMaraghy 2014), une méthode qui répond particulièrement aux systèmes d'assemblage reconfigurables est présentée. Les séquences d'assemblage des produits ainsi que leur ressemblance sont considérées comme des coefficients de

similarité pour regrouper les produits en familles. Les séquences d'assemblage des produits sont représentées sous la forme d'arbres binaires de recherche, un nouveau coefficient de similarité basée sur la séquence d'assemblage est introduit pour mesurer la distance entre toute paire donnée d'arbres binaires de séquences d'assemblage. Une classification hiérarchique est ensuite appliquée pour générer différents groupes de familles de produits qui peuvent être formés sur la base de chaque coefficient de similarité.

Une autre approche pour regrouper les produits en familles et déterminer les configurations correspondantes est présentée dans (Ossama et al. 2014) où un modèle de programmation linéaire mixte a été développé pour former simultanément les familles de pièces et les configurations de cellules correspondantes dans un environnement de production dynamique. Les auteurs proposent aussi, une heuristique de planification de la reconfiguration chargée de déterminer le plan de reconfiguration au niveau système et au niveau machine entre les périodes successives. Galan et al. (2007) proposent une autre méthodologie systématique tirée des méthodes et techniques utilisées dans la technologie de groupe avec des adaptations et des améliorations qui ont été apportées. La méthode commence par calculer une matrice qui résume la similitude entre des paires de produits. Ensuite, grâce à l'utilisation de la méthode AHP (Analytical Hierarchy Processing), une matrice unique qui contient les valeurs de similarité entre les produits est obtenue. Un algorithme est appliqué à cette matrice afin d'obtenir un dendrogramme qui montre les divers ensembles de familles de produits qui peuvent être formés. Dans (Abdi & Labib 2004b), le regroupement des produits en familles puis la sélection des configurations correspondantes sont traités successivement. Les auteurs proposent un lien entre les familles de produits et les configurations du système. La similarité opérationnelle est utilisée comme critère principal pour le regroupement des produits. La sélection de la configuration adaptée pour chaque famille de produits est basée sur un traitement hiérarchique analytique AHP où les objectifs du marché et les objectifs de production sont pris en compte.

Dans (Yigit & Allahverdi 2003) (Yigit et al. 2002) une méthode d'optimisation des produits modulaires pour un RMS est proposée. Dans ces travaux, le problème de la sélection des instances de modules optimales d'un produit est considéré simultanément avec le coût de la reconfiguration, en supposant qu'une configuration particulière du système peut être utilisée pour produire une instance particulière de la famille de produits. Bryan et al. (2007) ont appliqué l'optimisation dans leur travail, ils se sont concentrés sur la formation des familles de produits et la conception du système d'assemblage simultanément. La coévolution est essentielle dans le travail proposé, car les premières et futures générations du système sont conçues en considérant conjointement le système d'assemblage et la formation de la famille de produits grâce à l'optimisation mathématique.

4.4. Planification/Conception du processus de production

Le plan du processus de production est le lien entre le produit (la conception) et le système de production (ressources du système de production). D'un côté, ce plan est lié au produit via le lien entre les caractéristiques du produit et les opérations du processus (chaque caractéristique du produit est réalisée en exécutant une ou plusieurs opérations dans ce plan). De l'autre côté, le plan est lié au système de production par le lien entre les opérations décrites dans le plan et les ressources du système de production (chaque opération dans le plan est liée à une ou plusieurs ressources du système de production). Les liens bien définis entre les produits, les plans du processus et les ressources de production sont importants dans la conception des RMSs. En d'autres termes, créer le lien entre les produits et les ressources du système est impératif dans le cas des RMSs pendant la phase de conception et la phase opérationnelle du système.

La planification des processus fait référence à la planification et à la définition des étapes nécessaires à l'exécution d'un processus, qui devrait être capable de gérer les changements à court et à long terme. La planification des processus et l'établissement d'un lien entre les produits et leur système de production. Le nouvel apport, concernant les RMS, est l'introduction du plan de processus reconfigurable ou le RPP (Reconfigurable Process Plan), un plan qui pourrait être reconfiguré pour gérer de nouveaux produits qui n'ont pas été pris en compte lors de la conception du système (bien sûr, les nouveaux produits devraient partager certaines caractéristiques avec la famille de produits initiale). Ce concept (RPP) a été introduit dans (ElMaraghy 2007), où il est clairement mentionné qu'une reconfiguration efficace des plans du processus de fabrication est un outil essentiel pour les systèmes reconfigurables et réactifs. En outre, l'établissement d'une relation claire entre les caractéristiques du produit, les éléments du plan de processus de fabrication et tous les modules du système de production capable de les produire est un élément clé crucial pour la réalisation de plans de processus reconfigurables.

Donc, la planification du processus de production est un élément clé pour la reconfigurabilité. La disposition d'une méthode ou d'une approche de planification reconfigurable du processus de production facilitera largement la reconfiguration du système. Azab & ElMaraghy (2007a) proposent un nouveau concept pour la planification reconfigurable du processus. L'objectif est de déterminer la meilleure position pour insérer les nouvelles caractéristiques dans la séquence existante en optimisant des critères objectifs tout en respectant des contraintes spécifiques. Un nouveau modèle mathématique est introduit. Le plan principal serait modifié pour répondre aux exigences du nouveau produit et ses caractéristiques rajoutées. Les nouvelles portions du plan du processus correspondant aux nouvelles caractéristiques ajoutées (et leurs opérations d'usinage) sont générées et positionnées d'une façon optimale dans le plan général du processus. Une autre approche, hybride et séquentielle au niveau macro, est proposée dans (Azab & ElMaraghy 2007b). Le point de départ de cette approche et le plan

initial de fabrication, puis l'application des outils de modélisation et des algorithmes pour arriver aux plans (les processus) des nouvelles pièces dont les caractéristiques peuvent excéder les limites de la famille de produits originale. 2 méthodes sont présentées. La 1^{ère}, une planification reconfigurable du processus et la 2^{ème}, une replanification du processus. Une reconfiguration des processus pour optimiser la portée, l'ampleur et le coût de la reconfiguration est réalisée par l'utilisation d'un modèle de programmation entier (0-1) et aussi, comme le problème est combinatoire par nature, un algorithme de recuit simulé a été adapté pour la replanification. De même, [Musharavati & Hamouda \(2012\)](#) proposent trois algorithmes basés sur la simulation le recuit simulé pour résoudre le problème de planification du processus des RMSs.

4.5. Conception au niveau système : approches de conception générale

[Mehrabi et al. \(2000a\)](#) proposent un processus général de conception des RMSs. Dans ce processus de conception, on doit distinguer parmi les problèmes de conception : au niveau du système (architecture du système, configuration), au niveau composant (c.-à-d. cellule, machine, outils, etc.), et du problème de réduction de la phase de reconfiguration. Certaines pistes de recherche dans la conception des RMSs au niveau système ont été identifiées dans [\(Mehrabi et al. 2000b\)](#) : 1) le développement d'une approche systématique pour la conception des RMS au niveau système, 2) l'analyse de l'impact de la configuration du système sur la fiabilité, la qualité et le coût, 3) l'analyse économique des diverses configurations du système et leur sélection, 4) l'analyse et la conception du processus complet à partir de l'identification des besoins des clients.

Plusieurs chercheurs ont proposé des approches générales pour la conception des RMSs. [Bi et al. \(2008\)](#) définissent trois problématiques de conception des RMSs qui doivent être résolues le long du processus de conception. La première est la conception de l'architecture, où les composants et les interactions du système seront déterminés. Le but de cette phase est de définir une architecture modulaire du système capable de produire autant de variantes que possible, afin de faire face aux changements et aux incertitudes. La 2^{ème} problématique de conception concerne la conception de la configuration, dans laquelle les configurations pour des tâches spécifiques sont sélectionnées pour les périodes opérationnelles. Cette conception est réalisée dans la gamme des configurations disponibles spécifiées par l'architecture modulaire du système. La dernière problématique de conception est la conception du système de contrôle, dans laquelle les variables du processus sont déterminées afin que la configuration remplisse sa tâche de façon satisfaisante. D'après [\(Andersen et al. 2017\)](#) même si ces trois types de problèmes de conception sont suggérés principalement comme un cadre pour classer les travaux de conception des RMSs, ils représentent toujours une vue structurée de la conception des RMSs.

[Heisel & Meitzner \(2006\)](#) proposent une séquence d'étapes de conception qui devraient être

réalisées lors de la conception des RMSs. Les étapes vont de l'identification des exigences de reconfigurabilité jusqu'à la conception des modules du système et de leurs interfaces.

Dans le travail de (Benkamoun et al. 2014), les auteurs proposent un cadre d'architecture qui établit une pratique commune pour la création, l'analyse et la représentation des systèmes manufacturiers lors des processus de conception et de reconception. Les auteurs appliquent la terminologie et les approches de la conception axiomatique. La conception d'un RMS est considérée comme un problème qui est résolu en appliquant la logique suivante : analyse des besoins, décomposition du problème en exigences fonctionnelles, conception de l'architecture fonctionnelle et conception de l'architecture physique du système et de ces modules. Dans un autre travail (Benkamoun et al. 2015) proposent un environnement de conception intelligent visant à aider les concepteurs à faire face au problème de la conception pour la reconfigurabilité. La conception des RMSs comprend trois phases interdépendantes : l'analyse des besoins en termes de reconfigurabilité, la conception modulaire et la conception de la plate-forme. Ainsi, l'accent est explicitement mis sur la création des modules et les interfaces correspondantes, tandis que l'exploitation de cette conception modulaire et reconfigurable se réfère à la réutilisation des modules du système. Dans l'approche proposée par Benkamoun et al. (2015), le RMS est considéré comme constitué d'une plate-forme qui peut être développée conjointement avec la plate-forme produit, où une plate-forme est définie, selon (Michaelis & Johannesson 2012), comme un ensemble de sous-systèmes et d'interfaces qui forment une structure commune à partir de laquelle plusieurs variations du système peuvent être développées.

Francalanza et al. (2014) proposent une approche de conception qui inclut l'analyse et la détermination des exigences du système, qui sont déduites sur la base des produits, des processus et des besoins de l'entreprise. Deuxièmement, la synthèse est effectuée en incluant les décisions concernant le niveau ou le degré de la reconfigurabilité, les déclencheurs de la reconfigurabilité, et d'autres éléments spécifiques tels que les machines et leur disposition. Aussi, il est conseillé que la simulation et l'évaluation soient effectuées avant qu'une décision finale sur l'implémentation de la solution choisie ne soit prise. La structure de cette méthode implique que l'ensemble du processus de conception soit un cycle de résolution de problèmes qui est répété de nombreuses fois, jusqu'à ce qu'un système de production satisfaisant soit conçu pour répondre aux exigences initialement énoncées. La méthode proposée couvre la conception de la solution initiale du système, elle ne prend pas en compte la conception continue ou la reconfiguration du système pendant sa durée de vie opérationnelle.

L'approche proposée dans (Rösiö 2012b) est basée sur une conception conventionnelle des systèmes. Elle prend comme point de départ une méthode de développement des systèmes manufacturiers de (Bruch 2012) qui comprend l'initiation, la conception préparatoire et la conception détaillée. En plus de cela, le support pour considérer la reconfigurabilité est incorporé. Une partie

essentielle de ce support est la façon de déterminer le besoin de la reconfigurabilité dans le système et le lien entre ces besoins et les caractéristiques génériques des RMSs. [Schuh et al. \(2009\)](#) proposent une méthode de conception composée de quatre étapes basées sur la conception orientée-objet. La première étape concerne l'identification et la classification des déclencheurs de changement liés au produit, au volume et à la technologie. Ces déclencheurs de changement doivent être classés en fonction de la manière et du moment où le système doit changer, et de la prévisibilité de ceux-ci. La deuxième étape consiste à décrire le système manufacturier au moyen de profils de changement qui précisent comment les propriétés des éléments du système changent avec les déclencheurs de changement. Dans la troisième et la quatrième étape, les interdépendances sont déterminées entre les éléments, et les modules sont créés dans le but de séparer les éléments qui ne changent pas dans le même temps et pour la même raison.

[Deif & ElMaraghy \(2006\)](#) proposent une architecture de conception composée de deux modules ; un module de conception et un module de contrôle divisé en trois couches. Dans le module de conception, les différentes activités de conception sont décrites, tandis que le module de contrôle décrit le contrôle du processus de conception à chaque niveau en fonction d'objectifs stratégiques. La première couche de conception est l'étude du marché, où les exigences de capacité et de fonctionnalité sont déterminées. La deuxième couche est la couche de reconfiguration au niveau du système, génère des alternatives du système ou des configurations basées sur les niveaux requis de capacité et de fonctionnalité. Ces alternatives sont décrites en termes d'aspects physiques, logiques et humains. De plus, une sélection de la configuration la plus réalisable est également incluse dans cette couche en évaluant différents objectifs ou critères de performance. La dernière couche est la couche de reconfiguration au niveau composant, où l'implémentation de la reconfiguration dans les composants du système est traitée ou résolue. L'approche a été appliquée pour concevoir une chaîne d'assemblage automatique de circuits imprimés.

Dans le travail de [Bryan et al. \(2007\)](#), une distinction claire est faite entre la conception initiale du système (1^{ère} génération) et la conception continue (futurs générations du système) pendant la durée de vie opérationnelle du système. Deux phases de conception sont considérées : la phase initiale avec la conception du système et de la famille de produits et une phase de coévolution avec des reconfigurations. Dans la conception initiale, une approche d'optimisation est appliquée afin de décider simultanément quels produits à inclure dans une famille, quelles variantes à produire, le besoin en termes de postes de travail, et l'affectation des tâches à ces postes. Dans la phase de reconfiguration, il est envisagé d'intégrer de nouveaux besoins dans la famille de produits et d'apporter des modifications à la ligne d'assemblage. Dans des travaux ultérieurs, les auteurs améliorent encore cette méthode en abordant le problème de la conception simultanée de familles de produits et des systèmes d'assemblage et la conception de configurations des systèmes d'assemblage

pour des familles de produits en évolution.

Dans leurs travaux (Abdi & Labib 2003) (Abdi & Labib 2004a) (Abdi & Labib 2004b), les auteurs présentent une méthode de conception divisée en trois parties allant de la conception stratégique à la conception tactique. Sur le plan stratégique, la stratégie de conception du RMS est développée, ce qui implique de décider du type de système de production qui correspond le mieux aux exigences actuelles et futures liées au coût, à la réactivité, à la qualité et aux compétences des opérateurs. Le niveau de conception stratégique est la première étape, car il fournit un support décisionnel pour la sélection parmi les types de systèmes manufacturiers génériques, tels que les systèmes dédiés, les systèmes flexibles, ou les systèmes reconfigurables. La deuxième partie de la méthode consiste à établir le lien de reconfiguration, qui relie la conception stratégique et la conception tactique du système dans cette partie, les produits sont regroupés en familles et les liens entre ces familles et les configurations correspondantes du système sont établis. Le regroupement des produits est basé principalement sur le critère de la similarité opérationnelle. La famille de produits sélectionnée sert d'entrée à la conception tactique, qui est la troisième et la dernière étape de la méthode de conception. Dans cette étape, la faisabilité économique et opérationnelle de la configuration pour la famille de produits sélectionnée est évaluée et justifiée en appliquant l'AHP, ce qui conduit à une conception détaillée pour déterminer l'architecture du système et la disposition des machines et des outils. Un cas d'étude est inclus dans (Abdi & Labib 2003) (Abdi & Labib 2004b) pour démontrer l'applicabilité de l'approche proposée.

Récemment, un état de l'art sur les méthodes, cadres ou approches générales de conception des RMSs ainsi que sur les outils de soutien à la des RMSs a été réalisé dans (Andersen et al. 2017). Après une étude comparative, des méthodes trouvées, en termes de structure, de contenu et d'approche principalement, les auteurs ont identifié une méthode générique pour la conception des RMSs. La méthode proposée est composée de quatre phases : la 1^{ère} phase est la planification stratégique, elle couvre la planification et la prise de décisions stratégiques. Son but est de justifier les investissements et le potentiel d'un système reconfigurable. La 2^{ème} phase est la conception de base, elle inclut l'identification des familles de produits et la détermination du degré, du type et du niveau de reconfigurabilité du système. La 3^{ème} phase est la conception avancée, elle concerne la conception détaillée des modules du système en termes d'aspects logiques et physiques. Pendant cette phase, les spécifications détaillées des aspects physiques et logiques du système de production sont prises. La dernière phase est la phase de reconfiguration, le but de cette phase est de choisir parmi des configurations alternatives pendant la phase opérationnelle du système, et de décider quand et comment reconfigurer le système de production pendant sa durée de vie. Cette phase contient souvent une répétition d'étapes à partir de la phase de conception, par ex. en termes d'ajout d'un nouveau produit aux familles de produits, en concevant des modules supplémentaires, ou simplement en sélectionnant une nouvelle configuration parmi le nombre de choix possibles.

Dans la méthode proposée par [Andersen et al. \(2017\)](#), chaque phase comprend un cycle d'étapes, où une décision conduit soit à la phase suivante, soit aux itérations des étapes précédentes. De plus, l'ensemble du processus de conception est cyclique en lui-même, car la phase de reconfiguration entraîne la répétition de certaines étapes dans la conception. La méthode synthétisée correspond approximativement à la structure globale des méthodes conventionnelles de conception de systèmes telles que décrites.

La méthode proposée dans [\(Puik et al. 2014\)](#) et [\(Puik et al. 2013\)](#) s'applique spécifiquement à la phase de reconfiguration du système, et se concentre sur la conception et l'implémentation de nouvelles configurations du système. Le cadre de conception est un cycle d'optimisation, qui inclut le développement du système, l'évaluation des risques et les améliorations en boucle. L'étape du développement s'appuie sur les principes de la conception axiomatique, où des décompositions dans différents domaines sont effectuées [\(Puik et al. 2014\)](#), tandis que l'évaluation des risques repose sur l'analyse des effets du mode de défaillance (FMEA : Failure Mode and Effect Analysis). L'outil, modélisation qualitative et d'analyse des processus (QMAP : Qualitative Modeling and Analysis of Processes), contient trois parties ; des diagrammes de données de haut niveau décrivant les fonctionnalités de base de l'ensemble du processus, des modèles d'activité décrivant en détail les activités et une analyse des risques mettant en évidence les parties du nouveau processus qui présentent le risque le plus élevé.

Le cadre proposé dans [\(AlGeddawy & ElMaraghy 2009\)](#) pour la conception des RMSs, la terminologie de la conception axiomatique [\(Suh 1998\)](#) est appliquée, et la nécessité d'une conception bidirectionnelle ou simultanée des produits et des systèmes de production est soulignée. Dans ce cadre de conception, le produit et le système de production sont couplés dans une structure en boucle à l'aide des matrices de processus et de capacité. Lorsqu'il y a un changement dans le produit, ce changement est traduit dans la matrice du processus, ce qui provoque un changement dans le système si les capacités actuelles ne sont pas suffisantes. À leur tour, ces nouvelles capacités représentent une nouvelle opportunité de développement de produit.

Dans [\(Bensmaine et al. 2013\)](#), le problème de conception des RMS basé sur les spécifications des produits à fabriquer et les capacités des machines de production est traité. Le problème est lié à la sélection des machines reconfigurables candidates parmi un ensemble disponible, qui sera ensuite utilisée pour fabriquer un certain produit basé sur les caractéristiques de ce dernier. La sélection des machines considère deux principaux objectifs respectivement la minimisation du coût total (coûts de production, coûts de reconfiguration, coûts de changement d'outil) et le temps de réalisation totale. Un algorithme génétique est proposé pour résoudre le problème.

Un autre problème important de conception des RMSs au niveau système est le problème de

conception du plan du système (la disposition des machines dans l'atelier et les systèmes de transport qui les relient). Dans (Spicer et al. 2005), les auteurs ont montré que les architectures (la disposition des ressources dans l'atelier) symétriques sont les plus appropriées pour les RMSs. L'architecture ou la structure recommandée (série, parallèle ou hybride) pour un RMS avec les caractéristiques de base est décrite dans (Koren and Shpitalni 2010). Pour la conception d'un RMS avec la structure recommandée, une méthode mathématique a été développée, en appliquant cette méthode ; la configuration la plus favorable pourrait être sélectionnée. Le problème d'implantation ou de conception de l'architecture de systèmes de production reconfigurables a été étudiée dans (Guan et al. 2012). Un algorithme a été proposé afin de résoudre le problème d'arrangement des ressources d'un RMS utilisant les AGVs (Automatic Guided Vehicles) comme moyen de transport.

Cependant, en pratique, la transformation des exigences en un niveau et type de reconfigurabilité requis est une préoccupation. La conception axiomatique a été suggérée pour cela dans certaines contributions (Benkamoun et al. 2015) (AlGeddawy & ElMaraghy 2009), alors que (Terkaj et al. 2009) proposent une approche qui utilise l'ontologie de la flexibilité pour décider des composants physiques du système. (Rösiö 2012b) (Rösiö 2012a) considère ce problème en associant les différentes caractéristiques d'un RMS aux événements déclencheurs de la reconfiguration identifiés afin de décider du type de reconfigurabilité à concevoir dans le système.

Le premier problème, qui détermine le niveau de reconfigurabilité, fait référence à la capacité du système à augmenter sa capacité et à modifier ses fonctionnalités. Dans (Koren 2010) (Koren 2013), Koren a montré comment différentes configurations sont associées à différents degrés d'évolutivité (changement de capacité) et de convertibilité (changement de fonctionnalités). Ainsi, une partie essentielle de cette étape de conception comprend des métriques de performance liées aux différences dans les options de conception reconfigurables. De telles métriques ont été présentées par exemple par (Maler-Speredelozzi et al. 2003).

Le second problème, en ce qui concerne le type de reconfigurabilité, est étroitement lié au niveau du système sur lequel la reconfigurabilité est implémentée (ElMaraghy & Wiendahl 2009). La reconfigurabilité peut couvrir la capacité d'augmenter la capacité, de convertir entre les variantes de produits et de s'adapter aux nouvelles générations de produits. Les concepts couramment appliqués à cet égard sont les classes de changement et les facilitateurs proposés par (ElMaraghy & Wiendahl 2009). En outre (Terkaj et al. 2009) présentent une ontologie sur la flexibilité visant à fournir une méthode standardisée pour analyser la flexibilité. Ainsi, il semble que le concept et la base théorique pour décrire et classer les types et les niveaux de reconfigurabilité et de flexibilité sont plutôt bien établis dans la recherche.

Après l'exposition de ces travaux, on peut en conclure que la conception des RMSs au niveau

système englobe : la formation des familles de produit (voir chapitre 1, section 4.3) ; la planification du processus de production (voir, chapitre 1, section 4.4) ; les méthodes de conception du système de production, par exemple : (Rösiö & Säfsten 2013) (Abdi & Labib 2003) (Rösiö 2012b) (Benkamoun et al. 2014), la sélection de la configuration optimale (Youssef & ElMaraghy 2007) (Abdi & Labib 2003).

4.5.1. La conception/sélection de la configuration du système

Au niveau du système, il peut y avoir plusieurs configurations qui pourraient répondre aux exigences du même contexte. Le développement d'outils et de méthodologies nécessaires à la conception du système et à l'évaluation de ses différentes configurations est nécessaire. L'analyse des performances des diverses configurations du système facilitera grandement la sélection de la configuration la plus appropriée. L'impact de la configuration sur les performances du système en termes de fiabilité, de productivité, de qualité du produit, de capacité d'évolutivité et de coût est étudié dans (Koren, Hu, and Weber 1998).

Selon (Koren et al. 1999), une configuration d'un système de production est définie par un ensemble de machines et les liens entre eux. Les RMSs offrent une configuration dynamique et chaque configuration possède ses propres avantages/inconvénients en termes de coût, qualité et fiabilité du système. Le développement des outils et des méthodologies nécessaires pour concevoir le système, et l'évaluation de ses différentes configurations (en utilisant des critères comme : le cycle de vie du système, la qualité des produits, la fiabilité du système, et les préférences des décideurs, etc.) est nécessaire (Mehrabi et al. 2000b).

L'impact de la configuration sur les performances du système en termes de : fiabilité, productivité, qualité des produits, capacité d'évolutivité (réglage de la capacité de production) et de coût est traité dans (Koren et al. 1998). L'analyse des mesures de performance pour différentes configurations aidera énormément dans la sélection de la configuration du système la plus appropriée. Des principes de sélection des configurations les plus appropriées du système de production sont proposés dans (Spicer et al. 2002) et les différentes configurations sont comparées en termes de : débit de production ; équilibrage de la ligne ; coût d'investissement ; et capacité d'évolutivité (capacité d'ajuster le débit de production).

La structure (série, parallèle ou hybride) recommandée pour un RMS pratique avec les caractéristiques de base est décrite dans (Koren & Shpitalni 2010). Pour la conception d'un RMS avec la structure recommandée, une méthode mathématique rigoureuse est proposée. Elle consiste à calculer le nombre des configurations possibles pour un RMS. La détermination du nombre de machines nécessaires pour la production à partir des données (la quantité des pièces à fabriquer par jour, le temps de fabrication de chaque pièce, et le temps de travail par jour). À partir du nombre de

machines obtenu, les nombres de toutes les configurations possibles sont déterminés par une formule mathématique. Puis le nombre de configurations est réduit selon des critères (exemple : configuration favorable pour un RMS ou non). Après une comparaison entre les configurations restantes et l'évaluation des avantages et des inconvénients de chaque configuration, la configuration la plus favorable est sélectionnée.

Une prémisse de cette approche est que le RMS peut produire une famille de produits dans chaque configuration et qu'on doit sélectionner la configuration optimale pour chaque famille, puis sélectionner continuellement une famille de produits à produire dans chaque période opératoire. Cette prémisse est également présente dans le travail de (Xiaobo, Jiancai, et al. 2000) (Xiaobo, Wang, et al. 2000), qui proposent un modèle stochastique contenant une conception de configuration optimale et une sélection optimale, en utilisant une approche d'optimisation mathématique pour déterminer la configuration optimale pour une famille de produits à partir de plusieurs configurations réalisables.

4.6. Conception des RMSs : niveau machine

Souvent, la reconfigurabilité physique d'un RMS est facilitée par des machines reconfigurables qui peuvent être rapidement converties entre les variétés au sein des familles de produits (Z. M. Bi et al. 2008). Habituellement, les machines reconfigurables incluent les machines-outils, les systèmes de fixation, les machines d'assemblage, les machines d'inspection et les systèmes de manutention qui sont tous essentiels au concept RMS, car ils offrent une flexibilité et une capacité de reconfiguration au niveau des équipements (Katz 2007). Quelques lignes directrices existent dans ce domaine, par exemple les principes de conception des machines reconfigurables proposés par (Katz 2007). Cependant, même avec les contributions actuelles dans ce domaine, leur implémentation effective est limitée, car elles ne sont actuellement pas largement disponibles et sont en cours de développement (H.-P. Wiendahl et al. 2007) (Z. M. Bi et al. 2008) (Chaube et al. 2012).

La conception d'un RMS ne concerne pas seulement le système global ; ses sous-systèmes (généralement des machines) doivent également être conçus selon les principes de conception des RMSs mentionnés plus haut (voir chapitre 1, section 4.2). D'après (Mehrabi et al. 2000b), la structure variable des RMSs complique la conception de leurs composants. Dans la conception conventionnelle des machines, leurs composants ont été optimisés pour une architecture prédéfinie et unique. La tâche principale consistait à optimiser la machine pour une configuration spécifique. Par conséquent, comme il y a plus d'une configuration de la machine, l'optimisation doit être faite pour plusieurs configurations possibles, ce qui est beaucoup plus compliqué qu'une seule configuration.

Barhak et al. (2005) propose une approche en boucle fermée pour le contrôle de la qualité pour accélérer la phase de montée en puissance après la reconfiguration du système. L'élément clé de cette

méthode est l'utilisation d'une machine d'inspection reconfigurable RIM qui s'adapte aux configurations du système manufacturier. La RIM (Reconfigurable Inspection Machine) permet une inspection reconfigurable, rapide et précise en utilisant des capteurs sans contact. Les machines évolutives sont nécessaires dans les RMSs, dans (Spicer et al. 2005), l'architecture pour une machine évolutive est présentée. Un paramètre de conception basé sur une approche mathématique est présenté pour déterminer le nombre optimal de modules à inclure dans une machine modulaire et évolutive. Ce paramètre de conception est important, car il limite la taille de la machine et le nombre de modules et d'interfaces contenus dans la structure de base de la machine.

D'après (Landers et al. 2001), un élément majeur des RMSs est la machine-outil reconfigurable RMT. Contrairement aux machines CNC classiques qui sont des machines à usage général, les RMTs sont conçus pour une gamme d'opérations spécifiques et peuvent être reconfigurés rentablement lorsque les exigences ou les conditions changent. Les RMTs sont synthétisées en utilisant les principes de la conception modulaire pour atteindre la structure nécessaire pour usiner une pièce particulière. Des outils de conception systématiques qui ont été récemment développés pour les RMTs sont examinés dans (Landers et al. 2001), et une comparaison entre les RMTs et les machines-outils traditionnels est faite par Lorenzer et al. (2007). Les auteurs présentent un outil de modélisation pour soutenir les décisions au niveau de la conception des RMTs. L'outil est un logiciel qui permet d'évaluer les performances et la conformité aux exigences des variantes de la structure de la machine. Il permet de calculer les statiques et de simuler les propriétés dynamiques de la structure de la machine comme base d'évaluation. Les modèles des différentes variantes de la machine sont assemblés à partir de la bibliothèque virtuelle des modules qui contient des modèles des modules physiques existants. Donc, plusieurs variantes (configurations) peuvent être construites et analysées efficacement ce qui permet d'améliorer significativement les données de base sur lesquelles les décisions vont être prises.

Dans la littérature, on trouve plusieurs exemples de développement des RMTs : Son et al.(2010) présentent une RMT à broche contrôlable avec trois degrés de liberté, une autre machine RMT type arc est présentée par Dhupia et al. (2007). La RMT de type arc a été conçu pour obtenir une flexibilité personnalisée ce qui lui permet d'être reconfigurée pour usiner une famille de pièces. Aguilar et al. (2013) montrent la conception, le raffinement et la réalisation d'une RMT qui fournit une plate-forme flexible pour le tournage et le fraisage.

La conception d'une machine reconfigurable modulaire RMM (Reconfigurable Modular Machine) pour les RMSs est présentée dans (Xing et al. 2007). La recherche s'est concentrée sur la conception des sous-systèmes pour la RMM en utilisant la commande mécatronique modulaire générique. L'approche comprend : un contrôleur modulaire au niveau physique (hardware) et au niveau logique (software) ; une conception modulaire et une capacité générique « plug-and-play ».

En réponse au manque de méthodes dynamiques pour le routage des pièces dans un environnement de production dynamique, [Valente et al. \(2013\)](#) proposent un algorithme et un outil logiciel pour la gestion de flux de pièces dans un système de transport reconfigurable RTS (Reconfigurable Transportation system). L'algorithme assure le cheminement des pièces au bon endroit en optimisant le temps de production et le temps de transport. L'algorithme proposé n'est pas générique, il est développé spécialement pour un RTS bien défini ce qui empêche son application dans d'autres RTS, mais il peut être adapté. Aussi, [Carpanzano et al. \(2014\)](#) présentent une stratégie de routage des pièces pour un RTS constitué des multiples modules interdépendants. L'approche permet de faire le routage de pièce dynamiquement en s'adaptant au changement de configuration ou aux occurrences des pannes temporaires. L'approche permet de réduire le temps de production et l'augmentation du débit de production. L'approche est développée pour un RTS spécifique et son application directe sans adaptation sur un autre RTS est impossible

4.7. Conception des RMSs : système de contrôle/pilotage.

Le changement de la configuration du système implique un changement des paramètres du système et de ses composants. Par conséquent, le système de contrôle ou de pilotage devrait avoir la capacité de se reconfigurer et de s'adapter à ces nouvelles conditions ou paramètres. D'après [\(Mehrabi et al. 2000b\)](#), la nature modulaire des RMSs nécessite que le logiciel/matériel du système soit sous une forme modulaire ; c.-à-d. composé d'entités séparées totalement découplées du reste de telle sorte que l'addition/modification d'un composant du système soit possible. En outre, il devrait être extensible (c.-à-d. en mesure de répondre à de nouvelles : fonctionnalités, environnements et besoins), modifiable/réutilisable (facile à modifier et utilisable dans différents programmes, si nécessaire), et surtout reconfigurable (pouvant supporter différentes configurations et les interactions internes/externes des modules sans modifications dans le logiciel).

Différentes architectures de contrôle et technologies pourraient être utilisées pour implémenter un contrôleur qui s'adapte aux changements de la partie opérationnelle du système d'un RMS. Le système de contrôle doit être conçu selon les principes d'une architecture ouverte pour soutenir la structure modulaire d'un RMS. L'architecture ouverte du système de contrôle est la technologie qui permet d'intégrer, d'étendre, de remplacer et de réutiliser les composants matériels et logiciels dans un système de contrôle, même après son l'installation [\(Mehrabi et al. 2000b\)](#). Le contrôleur à architecture ouverte ou OAC (Open Architecture Controller) fournit l'infrastructure nécessaire pour implémenter le système de contrôle d'un RMS. [\(Pritschow et al. 2001\)](#) considèrent les systèmes de contrôle à architecture ouverte le facteur clé pour la réalisation des systèmes manufacturiers modulaires et reconfigurables. Les progrès dans le développement des OAC ont été examinés dans [\(Pritschow et al. 2001\)](#) [\(Koren et al. 1996\)](#).

Aussi, les architectures multi-agents sont souvent utilisées pour contrôler des RMSs (Kruger & Basson 2013). À titre d'exemple, une architecture de contrôle des RMSs et sa méthode de conception basée sur : les réseaux de PETRI, les architectures orientées service, les systèmes holonique, et les systèmes multi-agents, a été proposée par Da Silva et al. (2016). Les systèmes multi-agents pour les systèmes manufacturiers semblent apporter une réponse adéquate aux perturbations brusques au niveau atelier. L'utilisation de ce nouveau paradigme a la possibilité d'accroître considérablement la flexibilité, l'efficacité, l'évolutivité et la maintenabilité des systèmes reconfigurables et de fournir une alternative intéressante à l'ensemble actuel des approches disjointes qui sont actuellement appliquées dans ce domaine. Morales-Velazquez et al. (2010) présentent le système MADCON (Multi Agent Distributed CONtroller) qui est une plate-forme à architecture ouverte à base d'unités matérielles et logicielles multi-agents qui permet de développer un contrôleur distribué multi agents, ils proposent aussi une approche de co-conception matérielle et logicielle pour le développement d'un contrôleur intelligent en utilisant une structure simple et intuitive. MADCON a été appliqué sur une machine CNC (de type tour) afin de valider l'architecture proposée. Barata et al. (2008) ont appliqué l'architecture multi agents pour développer un système de contrôle au niveau atelier pour un banc d'essai de taille réduite.

Les structures hiérarchiques des systèmes de contrôle, largement utilisées dans les systèmes de production de masse, pourraient également être utilisées dans le pilotage des systèmes reconfigurables en tenant compte de l'évolution de la configuration du système dans le temps. Odrey & Mejía (2003) présentent une architecture pour le contrôle des systèmes manufacturiers flexibles et reconfigurables. L'architecture est une hybridation entre les systèmes hiérarchiques et les systèmes multi-agents intelligents. L'architecture proposée offre des capacités de réactivité et d'adaptabilité pour la résolution des erreurs dans le contrôle des systèmes de production à événements discrets.

Une architecture de contrôle holonique est souvent utilisée pour le contrôle des RMSs. Dans Kruger & Basson (2013), les logiciels agents et les blocs fonctionnels IEC-61499 sont évalués comme des stratégies alternatives pour mettre en œuvre le contrôle holonique pour un sous-système modulaire d'alimentation d'un système d'assemblage reconfigurable expérimentale. Valente & Carpanzano (2011) proposent une approche pour le développement des systèmes automatisés adaptatifs dans les RMSs. L'approche permet l'identification et la modélisation des tâches d'automatisation, ainsi que la conception et l'implémentation des solutions de contrôle auto-adaptatives et distribuées basées sur l'IEC-61499. Un algorithme dynamique d'ordonnancement des tâches d'automatisation dans le temps est décrit. L'approche a été appliquée sur une cellule de robot fonctionnant dans une ligne pilote d'assemblage.

Dans le but de proposer une méthodologie intégrée qui adopte une approche orientée-objet pour la modélisation, la conception et la programmation du logiciel de contrôle reconfigurable, Bruccoleri

(2007) a développé un logiciel de contrôle pour une cellule robotisée de production. Il a proposé et comparé deux méthodologies de développement différentes, à partir de la conception jusqu'à l'implémentation. La 1^{ère} utilise le langage à contacts (CONT) pour une cellule de production contrôlée par un PLC, la 2^{ème} utilise la modélisation et les techniques de la programmation orientée-objet pour une cellule de fabrication contrôlée par un PC. L'UML (Unified Modeling Language) est utilisé comme outil pour la conception et la modélisation du système de contrôle. Une autre structure de contrôle de haut niveau orienté-objet pour la gestion des erreurs en temps réel est proposée dans [Bruccoleri et al. \(2006\)](#).

Les méthodes de planification, d'ordonnancement, d'affectation des tâches, de gestion des changements de configuration, de gestion des défauts, etc., et toutes les méthodes pour assurer le bon fonctionnement d'un RMS sont considérées comme faisant partie de la conception du système de contrôle ou de pilotage.

Nous pouvons aussi noter les travaux concernant la génération automatique de système de commande qui sont, bien sûr, particulièrement bien adaptés aux systèmes reconfigurables. Lallican propose dans [\(Lallican 2007\)](#) de concevoir un système transitive et se commande en utilisant une approche composant. Adam propose dans [\(Adam 2013\)](#) de générer automatiquement la commande et un simulateur d'un système transitive à base de convoyeurs.

4.7.1. Modélisation des RMSs

Une question qui semble retenir l'attention dans la recherche est la possibilité de modéliser les RMSs, afin de pouvoir analyser, simuler et vérifier le système avant les reconfigurations [\(Kahloul et al. 2013\)](#) [\(Dai et al. 2009\)](#). La modélisation d'un RMS permet d'évaluer les performances de ce dernier. Selon [Meng \(2010\)](#), un modèle dynamique est plus adapté pour un système reconfigurable, il se transforme avec le changement de la configuration du système. Les anciennes méthodes de modélisation engendrent des modèles statiques qui deviennent obsolètes une fois le système reconfiguré. Souvent, les outils utilisés pour la modélisation des RMSs sont statiques, donc le modèle n'est valide que pour la configuration actuelle et une fois le système reconfiguré, le modèle devient obsolète et il faut refaire la modélisation de la nouvelle configuration, ce qui consommera beaucoup de temps et augmenta par conséquent le temps de montée en puissance.

Généralement pour modéliser un système il faut disposer d'un langage de modélisation, une méthode de modélisation (processus de modélisation) et un outil qui supporte le langage et la méthode de modélisation. Dans la littérature quelques langages de programmation ont été utilisés pour modéliser les RMSs, on peut citer : les réseaux de PETRI, le langage universel de modélisation UML et le langage pour modéliser les systèmes SysML (Systems Modeling Language).

Les réseaux de PETRI sont utilisés généralement pour modéliser des RMSs afin de faciliter le contrôle et la supervision. Ils sont largement utilisés dans les RMSs grâce à leurs capacités de modélisation et d'analyse. Un modèle pour décrire le processus de reconfiguration du système manufacturier est développé dans [Meng \(2010\)](#) par l'application des réseaux de PETRI coloré orientés-objet. Le modèle proposé intègre ensemble les méthodes orientées-objet, les idées de raffinement par étapes et les réseaux de PETRI. Donc, les activités du RMS peuvent être encapsulées et modélisées par la méthode proposée, ce qui facilite la construction et l'investigation des RMSs par les développeurs. Dans [Li et al. \(2009\)](#), une méthode composée de trois phases pour la modélisation des RMSs est présentée. Dans la 1^{ère} phase, un diagramme d'activités (UML 2.0) est utilisé pour décrire la configuration du RMS. Dans la 2^{ème} phase, les diagrammes d'activités pour les sous-systèmes formant le RMS sont transformés en réseaux de PETRI. Les relations entre les sous modèles et les sous-systèmes sont synthétisées rapidement en un réseau de PETRI complet pour la configuration initiale du RMS dans la 3^{ème} phase. [Chalfoun et al.\(2013\)](#) présentent un modèle générique pour la modélisation des RMSs. Le modèle est basé sur la décomposition du RMS selon 2 axes. Le premier axe, horizontal, concerne la structure du système et ses configurations. Le second axe, vertical, distingue la partie logique de la partie physique. Aussi, une formalisation et une représentation de chaque composant de ce modèle sont proposées. Le RMS est présenté d'une manière modulaire. Des métas modèles pour la structure, les configurations et les opérations sont présentés par l'utilisation du langage de modélisation des systèmes (SysML).

4.8. Conception des RMSs : gestion du système

4.8.1. Réduction du temps de montée en puissance

Une autre problématique critique dans la reconfiguration est le temps de montée en puissance, qui définit le temps écoulé entre la reconfiguration du système et le moment où la production atteint sa production planifiée en volume, variété et qualité ([Koren et al. 1999](#)). Après la reconfiguration d'un RMS, le système de production doit être « calibré » avant qu'il puisse produire le volume et la qualité nécessaires. Cette phase de calibration est appelée montée en puissance, et peut prendre un temps considérable dans le cas des systèmes de production conventionnels. Pour qu'un RMS soit pratique, il est nécessaire de réduire considérablement la durée de la phase de montée en puissance. Cet objectif nécessite des diagnostics, des calibrations, et des méthodologies appropriées. Certaines pistes liées à cette problématique ont été identifiées dans [Mehrabi et al. \(2000b\)](#) :

- Le développement des approches systématiques et les principes fondamentaux pour identifier les causes racines de défaillance des composants, et les variations de qualité et de processus.
- La conception des composants robustes pouvant fonctionner de manière fiable et en toute sécurité sous différentes conditions de fonctionnement.

Des exemples d'approches pour réduire la phase de montée en puissance sont les prédictions d'erreurs humaines liées à la qualité (Elmaraghy et al. 2008) et l'application de l'analyse des variations du flux SVA (Stream-of-Variation Analysis). Dans Barhak et al. (2005), une méthodologie en boucle fermée pour le contrôle de la qualité afin d'accélérer la phase de montée en puissance est présentée. La méthodologie permet un diagnostic rapide des causes-racines grâce à l'intégration d'une machine d'inspection reconfigurable RIM (voir chapitre 1, section 4.6) et l'application de SVA. La machine permet une inspection reconfigurable, rapide et précise alors que la méthodologie SVA est utilisée pour analyser rapidement les mesures et identifier les causes-racines des erreurs produites pendant la production.

Un système de supervision multi-capteurs efficace permettra d'identifier rapidement et avec précision les sources de problèmes reliés au processus et les actions correctives peuvent être appliquées le plus tôt possible pendant la phase de montée en puissance. Mehrabi & Kannatey-Asibu (2001), présentent une méthode qui devrait faciliter le développement des systèmes de supervision des RMSs. Le principe de cette méthode est d'établir les liens entre les données mesurées par les capteurs et les causes probables de défiances ou de pannes en utilisant principalement les règles de la logique floue.

4.8.2. Indicateurs de performances des RMSs

Il y a diverses mesures de performances dans un RMS comme le temps de montée en puissance, le coût, la fiabilité, la disponibilité, les délais, le temps de reconfiguration, etc., qui permettent d'évaluer les performances d'un RMS. Dans Mittal & Jain (2014) l'accent est mis sur les mesures de performance et la façon de trouver la meilleure configuration pour les systèmes manufacturiers reconfigurables. Youssef & ElMaraghy (2008) proposent un modèle pour analyser les performances (la disponibilité du système et les taux de production prévus) des systèmes manufacturiers composés de machines modulaires non fiables avec des unités de production fonctionnant en parallèle. Pour évaluer les performances du système, la technique UGF (Universal Generating Function) a été utilisée. Cette technique est efficace dans l'analyse des performances des systèmes manufacturiers larges multi-états ou multi-configurations dans un temps raisonnable par la réduction drastique du nombre de configurations du système.

4.8.3. Planification de la production

Gérer les changements et les perturbations dus aux fluctuations des demandes, cycle de vie du produit, et les portefeuilles de différents produits exige des décisions efficaces de gestion de capacités et des stratégies de planification de la production. Gyulai et al. (2012) proposent une méthode qui sépare les produits à faibles volumes des produits à volumes élevés de façon dynamique, en leur assignant les lignes de production dédiées ou reconfigurables appropriées, respectivement. Une autre méthode est introduite dans Gyulai et al. (2014), elle appuie la décision à long terme pour déplacer

l'assemblage d'un produit, dont la demande diminue, à partir d'une ligne dédiée à une ligne reconfigurable, sur la base de l'investissement calculé et les coûts opérationnels. [Bruccoleri et al. \(2005\)](#) proposent une approche basée sur le concept multi agent pour les activités de planification de la production dans les entreprises reconfigurables, caractérisées par la complexité, la distribution géographique des capacités de production et composées par des systèmes de production reconfigurables. Dans [Wang & Koren \(2012\)](#), on présente une méthodologie de planification de l'évolutivité des RMSs qui peut progressivement étalonner la capacité du système en le reconfigurant. Un algorithme génétique d'optimisation est développé pour déterminer la façon la plus économique de reconfigurer le système existant. Ajout ou suppression de machines pour répondre aux nouvelles exigences de débit et de rééquilibrer le système en même temps.

En ce qui concerne l'équilibrage des lignes de production dans les systèmes reconfigurables [Borisovsky et al. \(2013\)](#) proposent un algorithme génétique basé sur la représentation de permutation de solutions et un décodeur heuristique. Le problème d'optimisation consiste à concevoir une ligne d'usinage sous forme d'une séquence de postes de travail, où pour chaque station de travail un nombre de machines et une séquence d'opérations sont affectées de sorte que les contraintes décrites sont remplies et l'objectif qui est le nombre total de machines, n , dans ligne doit être minimisé.

5. Conclusion

La mondialisation entraîne des fluctuations du marché qui ont entraîné une grave instabilité pour de nombreuses entreprises manufacturières qui luttent pour survivre. Pour rester compétitives, les entreprises manufacturières doivent s'adapter aux changements rapides du nouvel environnement économique mondial. Donc, la réactivité de l'entreprise est essentielle pour rester rentable dans un monde en évolution rapide. L'entreprise devrait être équipée d'un système de production qui peut être rapidement changé et reconfiguré afin de pouvoir suivre les changements du marché.

Les RMSs sont le nouveau concept qui a été introduit afin de permettre aux entreprises de rester compétitives dans un environnement caractérisé par des changements fréquents et imprévisibles. Le principal défi d'un RMS est sa conception. La plupart des méthodes proposées dans la littérature ne traitent pas le problème de conception d'un RMS dans son ensemble, mais elles se limitent à une partie du problème. Cela est principalement dû à la complexité d'un RMS qui rend difficile la mise en place d'une méthode globale, standard et détaillée qui répond à tous les problèmes de conception d'un RMS en raison de sa complexité. Les problématiques de conception d'un RMS incluent :

- La formation des familles de produits afin de limiter la flexibilité du système de production.
- La définition et la spécification des besoins en termes de reconfigurabilité.
- La conception de l'architecture du système de production

- La conception des machines reconfigurables qui permettent la reconfiguration physique du système.
- La conception du système de pilotage du système.
- La mise en place des méthodes et d'outils pour faciliter la gestion du système pendant sa phase opérationnelle, par exemple, des outils pour supporter le choix des configurations futurs du système et de sa reconfiguration.

Pour répondre à ces problématiques, plusieurs travaux ont mené dans le domaine académique principalement. Malgré les nombreux travaux présentés dans la littérature, on a constaté que certaines difficultés persistent, on peut citer :

- La difficulté d'intégrer les exigences de reconfigurabilité dans le processus de conception

L'identification des besoins et des exigences de la reconfigurabilité est en général vaguement définie, et la littérature examinée ne fournit que des indications limitées des procédures spécifiques à appliquer. La détermination du degré, du type et du niveau de reconfigurabilité est difficile. Ces trois décisions sont fortement liées et sont généralement spécifiques au cas d'étude et sont difficiles à généraliser. La plupart des contributions sur la conception des RMSs supposent que les besoins et les déclencheurs de reconfiguration pendant la phase opérationnelle du système sont identifiés au départ, mais les approches pour identifier ces déclencheurs sont rarement décrites.

- Le manque de couverture du cycle de vie complet du système reconfigurable

Les méthodes de conception des RMS partagent des similitudes en ce qui concerne la distinction entre les phases et en termes d'activités à inclure dans chaque phase, malgré leurs différences de terminologie. Généralement, elles forment une séquence pour définir les exigences et les spécifications du système, concevoir une solution du système, implémenter physiquement la solution choisie, faire fonctionner le système, et évaluer continuellement sa capacité à satisfaire les exigences conduisant à des reconfigurations. Certaines méthodes de conception ne tiennent pas compte de la phase opérationnelle du système et de sa reconfiguration, alors que d'autres méthodes couvrent la totalité du cycle de vie du système. Cependant, dans leur ensemble, les méthodes ne diffèrent pas significativement dans leur phase de conception, mais elles diffèrent dans la couverture de la phase d'exploitation du système et dans la phase de planification initiale.

- Les méthodes de conception proposées manquent de détails

Certaines méthodes de conception représentent simplement des cadres conceptuels alors que d'autres incluent des méthodes et des procédures plus complètes pour guider les concepteurs. Une différence notable entre ces méthodes est la couverture des niveaux de structuration d'un système manufacturier (voir Figure 5). Certaines méthodes couvrent la conception des deux niveaux

supérieurs, tels que l'usine, mais aussi des niveaux inférieurs, tels que les postes de travail et les outils. Certaines méthodes ne prennent en compte que le système manufacturier dans sa globalité, tout en ignorant la conception des niveaux inférieurs. Généralement, plus d'importance a été accordée à la structure du processus de conception plutôt qu'à la façon de réaliser chaque étape.

- Manque d'application industrielle

Les implications pratiques de la conception des RMSs peuvent être considérées comme largement déterminées par les différences existantes entre la conception des systèmes de production conventionnels et la conception des systèmes de production reconfigurables. Ces différences augmentent la complexité du processus de conception, ce qui constitue un défi pour le transfert efficace de la reconfigurabilité dans l'industrie. L'application des méthodes de conception n'a pas encore été largement abordée ou prouvée. Ainsi, la littérature ne présente que quelques cas concrets.

L'examen des solutions proposées pour résoudre les problèmes de conception des RMSs montre que la plupart de ces solutions ne traitent pas le problème de conception dans son ensemble, certaines se concentrent sur le regroupement des produits en familles, d'autres sur l'architecture du système, d'autres sur le système de pilotage, etc. On peut penser qu'il y a peu d'accords sur la structure du processus de conception, ce qui est largement dû, selon [Andersen et al. \(2017\)](#), à la terminologie, au niveau de détail, et l'approche de conception. Par conséquent, on peut conclure qu'il y a un manque de méthode générique pour concevoir un RMS dans son ensemble. Cette conclusion est cohérente avec les conclusions des travaux de recherche précédents selon lesquelles une méthode de conception systématique d'un RMS manque ([Z.M. Bi et al. 2008](#)) ([El Maraghy 2006](#)) ([Malhotra et al. 2012](#)) ([Malhotra et al. 2009](#)) ([Malhotra et al. 2010](#)) ([Rösiö & Säfsten 2013](#)). Par conséquent, nous proposons dans cette thèse, une méthodologie générique pour la conception d'un RMS.

La méthodologie proposée dans cette thèse est basée sur les principes de l'ingénierie des systèmes, qu'est une discipline pour la conception des systèmes complexes. Le chapitre suivant est une introduction à l'ingénierie des systèmes et aussi un état de l'art sur les méthodes de conception des systèmes complexes.

Chapitre 02 : Ingénierie systèmes et les méthodes de conception des systèmes complexes

1 Introduction

Aujourd'hui, les systèmes industriels complexes sont partout, souvent on ne se rend pas compte de leur complexité parce qu'ils nous sont familiers. Les systèmes de transport (automobile, ferroviaire, aéronautique), les équipements industriels (systèmes de communication, systèmes de production), les systèmes d'information dans les différents domaines (commercial, production, finance, logistique) sont des exemples des systèmes industriels complexes que nous utilisons quotidiennement.

La conception de ce type de systèmes est généralement une opération complexe sur le plan technique et sur le plan managérial et organisationnel. Plusieurs ingénieurs spécialisés, plusieurs domaines scientifiques, et des coûts énormes peuvent être impliqués dans le développement de ces systèmes. Les systèmes industriels complexes sont difficiles à concevoir et sont généralement composés de plusieurs sous-systèmes et de composants de différentes technologies. Cela rend la compréhension d'un système complexe dans sa globalité et dans tous ces détails, par une seule personne et même par un ensemble de quelques personnes, une tâche quasiment impossible.

Pour gérer cette complexité, une nouvelle discipline connue sous le nom « Ingénierie système ou IS » a été créée et popularisée dans le domaine industriel. Cette discipline regroupe les bonnes pratiques des industriels pour développer les systèmes complexes réussis en optimisant les coûts et les délais. Ces bonnes pratiques sont généralement, des processus, des méthodes ou des méthodologies qui sont le fruit de longues expériences dans le développement des systèmes complexes. L'IS se base sur un concept fondamental qui consiste à décomposer les systèmes complexes en sous-systèmes plus faciles à comprendre et à concevoir. Dans ce cadre de conception, l'IS fournit les techniques nécessaires pour assister toutes les étapes du processus d'analyse et de développement (définition, analyse, et spécification des exigences, conception de l'architecture, l'intégration, et finalement la vérification, la validation et la qualification du système) qui ensemble déterminent le développement des systèmes industriels complexes.

L'objectif de ce chapitre est de présenter les différentes méthodologies de conception des systèmes complexes industriels, ces méthodologies sont généralement basées sur les processus, les méthodes et les outils de l'IS. Donc, dans un premier temps, on présentera l'IS et son utilité dans le développement des systèmes complexes, on donne une vision globale sur les processus de cette discipline. Ces processus peuvent être adaptés et utilisés pour développer, pour une organisation particulière, une méthodologie pour le développement des systèmes complexes pour un domaine

spécifique comme la logistique interne par exemple. On exposera aussi les trois principales normes de l'IS (IEEE-1220, ISO/IEC/IEEE-15288, EIA-632).

Ces normes regroupent les processus de base de l'IS. Par la suite, on présentera la nouvelle approche ou philosophie de l'IS, c'est l'approche dirigée par les modèles ou l'approche MBSE (Model Based System Engineering). Principalement cette approche utilise les mêmes principes de base de l'IS, mais les différentes activités du processus de développement du système seront réalisées en utilisant principalement les modèles. Pour montrer l'intérêt porté à cette nouvelle approche de développement, on présentera quelques méthodologies MBSE trouvées dans la littérature. Ces méthodologies peuvent être adaptées ou utilisées comme une base pour développer une méthodologie MBSE propre à un organisme particulier pour lui faciliter le développement des systèmes.

2 Définitions

Pour faciliter la compréhension de la suite de ce chapitre, on définit quelques concepts de l'IS.

- **Un processus** : Un processus est une suite logique de tâches exécutées pour atteindre un objectif particulier. Le processus définit « le quoi », c.-à-d. ce qui doit être fait, sans préciser comment chaque tâche est effectuée (Estefan 2008).
- **Une méthode** : Une méthode se compose des techniques nécessaires pour exécuter une tâche particulière. En d'autres termes, elle définit « Comment » exécuter une tâche (dans ce contexte, les mots « méthode », « technique », « practice », et « procédure » sont souvent utilisés de manière interchangeable. À chaque niveau, les tâches ou les activités de chaque processus sont exécutées en utilisant des méthodes. Cependant, chaque méthode est aussi un processus lui-même, avec une suite de tâches à exécuter. En d'autres termes, le « Comment » à un niveau d'abstraction devient le « Quoi » du niveau inférieur suivant (Estefan 2008).
- **Un outil** : Un outil est un instrument qui, lorsqu'il est appliqué à une méthode particulière, peut améliorer l'efficacité de la tâche, à condition qu'il soit appliqué correctement et par une personne avec les compétences et les formations nécessaires. Le but d'un outil devrait être de faciliter l'accomplissement du « Comment ». Dans un sens plus large, un outil améliore le 'Quoi' et le 'Comment'. La plupart des outils utilisés pour soutenir l'IS sont des outils logiciels (Estefan 2008).
- **Une méthodologie** : Une méthodologie est un ensemble de processus reliés, des procédés et des outils. Une méthodologie est essentiellement une 'recette' et peut être considérée comme l'application des processus reliés, des méthodes et des outils sur une classe de problèmes qui ont tous quelque chose en commun (Estefan 2008).

3 Ingénierie système

3.1 Définition

L'IS est une démarche méthodologique, coopérative et interdisciplinaire qui englobe l'ensemble des activités adéquates pour concevoir, développer, faire évoluer et vérifier un ensemble de produits, processus et compétences humaines apportant une solution économique et performante aux besoins des parties prenantes et acceptable par tous (IEEE 2005).

Dans Haskins (2006) et selon l'INCOSE, l'IS est définie comme une approche interdisciplinaire et des moyens qui permettent la réalisation des systèmes réussis. Une autre définition donnée par Eisner (Eisner 2002) cite que l'IS est un processus itératif de synthèse, de développement, et d'application d'un système qui répond d'une façon optimale à toutes les exigences du système. L'IS n'est pas une nouvelle méthode qui viendrait remplacer telle ou telle autre méthode, mais une approche globale et systématique apte à mettre en cohérence les méthodes et les démarches existantes, tel que l'analyse fonctionnelle, l'analyse de la valeur, la sûreté de fonctionnement, la conception à objectifs désignés, les méthodes d'ingénierie de spécialités ou de génie des métiers impliqués, etc., tout en bénéficiant de leurs retours d'expériences (AFIS 2009).

Les efforts en IS embrassent l'ensemble du cycle de vie du système et leur mise en cohérence mobilise l'ensemble des corpus théoriques (sciences de l'ingénieur, sciences humaines, sciences cognitives, génie logiciel, etc.). L'ingénierie système consiste à examiner le problème dans sa globalité, en tenant compte de toutes les perspectives et toutes les variables ainsi que les aspects techniques et sociaux.

3.2 Histoire

La discipline de l'IS a émergé comme un moyen efficace pour gérer la complexité et le changement qui se sont multipliés dans nos systèmes/produits, nos services, et dans notre société. Réduire les risques associés au développement des nouveaux systèmes ou la modification des systèmes complexes existants est l'objectif principal de l'IS (Haskins 2006). Les premiers organismes à s'intéresser à l'IS ont été les grandes institutions américaines de la défense. La NASA (National Aeronautics and Space Administration) et l'USAF (United States Air Force) ont tenté, dans les années 1960, de cadrer le développement des programmes militaires et d'exploration spatiale, au travers d'approches industrielles plus rationnelles. Cela a abouti, en 1991, à la création de l'International Council On Systems Engineering INCOSE (voir chapitre 2, section 3.3.1) premier organisme mondial d'IS, tant par sa date de création que par sa taille.

Une grande partie des disciplines d'ingénierie, telles que le génie civil, le génie logiciel, l'électronique, l'automatique et la productique, après avoir chacune travaillé isolée du reste des autres disciplines en se spécialisant dans son domaine, se retrouvent en 2008 concernées par cette nouvelle discipline agrégative qu'est l'IS.

3.3 Organismes travaillant sur l'IS

3.3.1 INCOSE

L'INCOSE est une organisation à but non lucratif qui se consacre à l'avancement de l'IS et d'élever le statut professionnel des ingénieurs système. Fondée en 1990, INCOSE a plus de dix mille membres représentant un large spectre de métiers (des ingénieurs, des doctorants, des professeurs, etc.). Les membres travaillent ensemble pour faire avancer leurs connaissances techniques, échanger des idées avec des collègues, et de collaborer pour faire progresser l'IS (INCOSE 2016).

3.3.2 AFIS

L'Association Française d'Ingénierie Système AFIS est une association travaillant sur le développement et la promotion de l'IS dans l'industrie française. Les membres de l'association sont soit des organisations (groupes industriels, PME et organisations d'enseignement et de recherche), soit des individus intervenant à titre personnel.

4 Cycle de vie d'un système/produit

Selon Haskins (2006), chaque système développé par l'homme a un cycle de vie, même s'il n'est pas formellement défini. La définition d'un cycle de vie du système permet d'avoir une vision globale et compréhensible du système. Elle permet aussi d'établir un cadre pour répondre aux besoins des parties prenantes PP d'une manière ordonnée et efficace. Cela se fait habituellement en définissant les étapes du cycle de vie, et en utilisant des points de décision pour déterminer l'état de complétude pour passer d'une étape à l'autre. Sauter les phases et éliminer les points de décision peuvent augmenter considérablement les risques (en termes de coûts et de délais), et peuvent affecter négativement le développement technique ainsi en réduisant l'effort d'ingénierie. Les objectifs et les livrables doivent être définis pour chaque phase du cycle de vie. Les processus et les activités du cycle de vie sont choisis et adaptés pour chaque phase et utilisés pour atteindre les objectifs et les livrables de cette phase' (ISO/IEC/IEEE15288 2015).

4.1 Les phases du cycle de vie d'un système

Le modèle du cycle de vie comprend un ou plusieurs modèles de phase, selon les besoins. Il est assemblé comme une séquence d'étapes qui peuvent se chevaucher et/ou itérer, en fonction de la portée, de l'ampleur, de la complexité, de l'évolution des besoins et des possibilités (ISO/IEC/IEEE15288 2015). Le tableau ci-dessous énumère les six phases identifiées du cycle de vie d'un système dans la norme (ISO/IEC/IEEE15288 2015). L'objectif de chaque phase est brièvement

décrit, et les options de décisions sont indiquées. Les phases peuvent se chevaucher, et la phase opérationnelle et la phase de maintenance sont en parallèle. Il faut noter que les possibilités de décision sont les mêmes pour toutes les phases.

Phase	Objectifs	Décisions
Concept	<ul style="list-style-type: none"> • Identifier les besoins des parties prenantes. • Explorer les concepts. • Proposer des solutions faisables. 	Les options de décision : <ul style="list-style-type: none"> • Exécuter la phase suivante. • Continuer cette phase. • Revenir à la phase précédente. • Maintenir l'activité du projet. • Terminer le projet.
Développement	<ul style="list-style-type: none"> • Raffiner les exigences du système. • Créer des descriptions des solutions. • Construire le système. • Vérifier et valider le système. 	
Production	<ul style="list-style-type: none"> • Produire le système. • Inspecter et tester le système. 	
Utilisation	<ul style="list-style-type: none"> • Mettre le système en opération pour satisfaire les besoins des utilisateurs. 	
Support	<ul style="list-style-type: none"> • Maintenir le système en mode opérationnel. 	
Retrait	<ul style="list-style-type: none"> • Retirer le système. 	

Tableau 2 : Les phases du cycle de vie d'un système selon la norme. (ISO/IEC/IEEE15288 2015)

La décomposition du cycle de vie d'un système n'est pas universelle, chaque organisme peut adapter la décomposition qui répond le mieux à ces besoins et qui lui facilite le travail. Pour montrer cette réalité, la figure ci-dessous compare les phases du cycle de vie d'un système du point de vue de la norme (ISO/IEC/IEEE15288 2015) avec d'autres points de vue de décomposition (par exemple selon le point de vue de l'United States Department of Defense DoD. Sur la figure ci-dessous, on voit que la phase de concept de la norme (ISO/IEC/IEEE15288 2015) est alignée avec la phase de pré-acquisition du système et avec la phase de planning dans les départements de la défense et de l'énergie des états unis respectivement. Des points de décision typiques sont aussi présentés sous la ligne dans la figure ci-dessous.

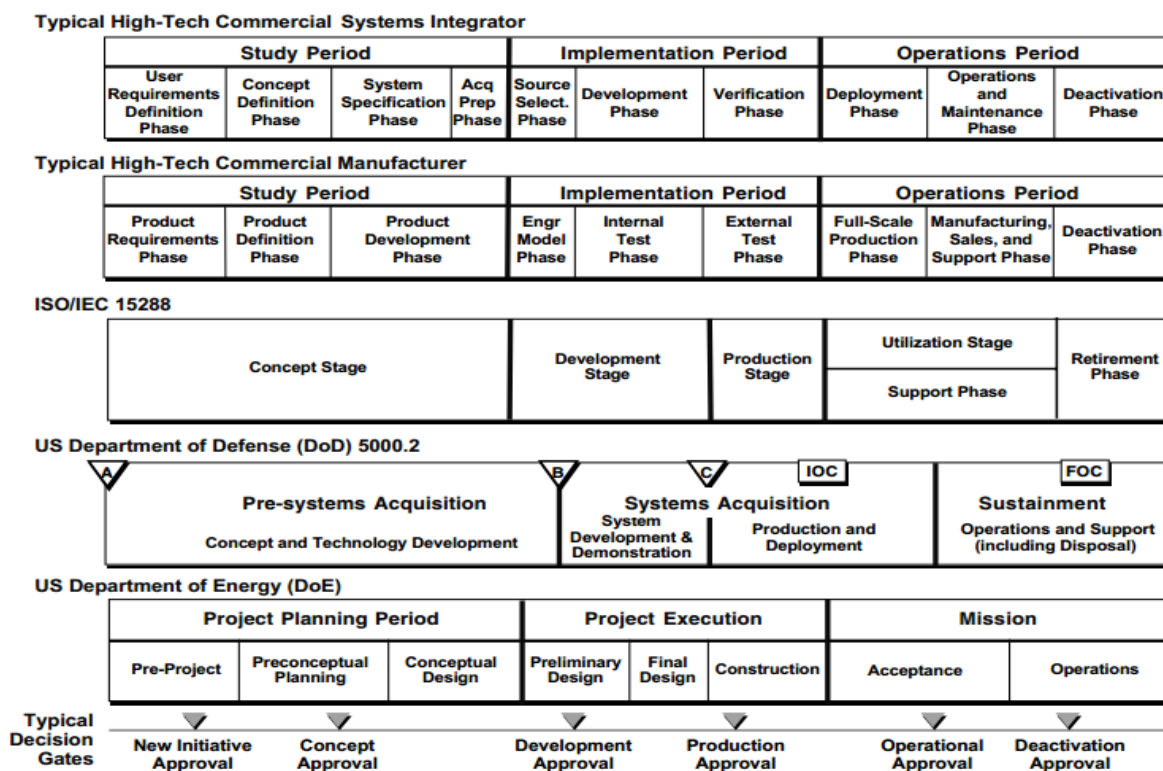


Figure 7 : Les différents points de vue d'un cycle de vie d'un système. (Haskins 2006)

4.1.1 Phase de concept

Cette phase est exécutée pour évaluer de nouvelles opportunités commerciales et développer des exigences préliminaires du système et une solution de conception possible. La première étape dans cette phase consiste à identifier, clarifier et documenter les exigences des parties prenantes.

Pendant cette phase, l'équipe commence des études approfondies pour évaluer plusieurs concepts candidats et éventuellement fournir une justification pour le concept du système qui est sélectionné. Dans le cadre de cette évaluation, des maquettes peuvent être construites (pour les systèmes physiques) ou codées (pour les systèmes logiciels), les modèles d'ingénierie et de simulation peuvent être exécutés, et des prototypes de composants critiques peuvent être construits et testés. Les prototypes sont utiles pour vérifier la faisabilité de concepts et d'explorer les risques et les opportunités.

Ces études élargissent l'évaluation des risques et des opportunités pour inclure une évaluation de faisabilité, de l'impact environnemental, des modes de défaillance et d'analyse des risques. L'objectif principal est de fournir l'assurance que les solutions proposées sont réalisables.

4.1.2 Phase de développement

La phase de développement est exécutée pour développer un système qui répond aux exigences du client et qui peut être produit, testé, évalué, utilisé, maintenu, et retiré. La phase de développement inclut les activités de développement, d'intégration, de vérification et de validation.

4.1.3 Phase de production

La phase de production est exécutée pour fabriquer le système et éventuellement ses systèmes de support et de maintien selon le besoin. Des modifications du système peuvent être nécessaires pour résoudre les problèmes de production, de réduire les coûts de production, ou pour améliorer les capacités du système. Tous ces changements peuvent influencer les exigences du système, et peuvent exiger la revérification ou la revalidation du système.

4.1.4 Phase d'exploitation

La phase d'exploitation ou la phase opérationnelle est la phase pendant laquelle le système est exploité dans les environnements prévus en fournissant les services prévus avec une efficacité opérationnelle continue. Les modifications du système sont souvent prévues le long du cycle de vie du système. Ces modifications améliorent les capacités du système.

4.1.5 Phase de maintenance

La phase de maintien sert à fournir les services de logistique, de maintenance, et de support qui permettent le fonctionnement continu du système et la qualité de ses services. Le système pourrait être modifié après la mise en service pour résoudre des problèmes de maintenance et de support, réduire les coûts de fonctionnement, ou pour prolonger la durée de vie du système.

4.1.6 Phase de retrait

Cette phase est exécutée pour retirer le système et les services opérationnels et de soutien connexes, et pour exploiter et soutenir le système de retrait lui-même. Les activités d'IS de cette phase sont principalement exécutées pour veiller à ce que les exigences d'élimination soient satisfaites. En fait, la planification de l'élimination fait partie de la définition du système au cours de la phase de conception. L'expérience dans le 20ème siècle a démontré à plusieurs reprises les conséquences lorsque le retrait et l'élimination d'un système ne sont pas considérés dès le départ. Au début du 21e siècle, de nombreux pays ont modifié leurs lois pour considérer le créateur d'un système comme responsable de son élimination correcte à la fin de sa phase opérationnelle.

Avec l'augmentation des problèmes liés à l'environnement, le cycle de vie de n'importe quel système doit englober non seulement les phases de développement, de production, et d'exploitation, mais aussi concentrer dès le début sur la phase de retrait de service quand le démantèlement ou l'élimination du système aura lieu.

5 Les processus de l'ingénierie Système

En IS, et selon l'AFIS (AFIS 2009), tout projet implique la réalisation de divers types d'activités : Activités techniques d'analyse du problème, de conception de la solution, d'intégration, de vérification et de validation ; activités de management de ces activités techniques ; activités de gestion des relations contractuelles entre clients et fournisseurs (maître d'ouvrage, maître d'œuvre, sous-traitants). Ces trois types d'activités sont respectivement regroupés en processus techniques, processus de management et processus contractuels. Les activités de support communes aux différents projets réalisés par une entreprise forment une quatrième classe de processus : les processus d'entreprise. Ces processus sont considérés aujourd'hui comme des fondements génériques pour toute mise en œuvre de l'IS. Les processus définis par les normes d'IS (voir chapitre 2, section 6) se classent en quatre groupes :

- **Les processus techniques** qui participent à la transformation des besoins en solution et à la vie opérationnelle des produits/systèmes correspondants,
- **Les processus de management** qui participent à la maîtrise des processus techniques dans le contexte du ou des projets,
- **Les processus contractuels** qui assurent les relations acquéreur-fournisseur (par exemple avec les clients et les sous-traitants pour un projet dans une entreprise),
- **Les processus d'entreprise** qui ont pour rôle de développer le potentiel en IS de l'entreprise en managant les domaines communs aux différents projets d'IS.

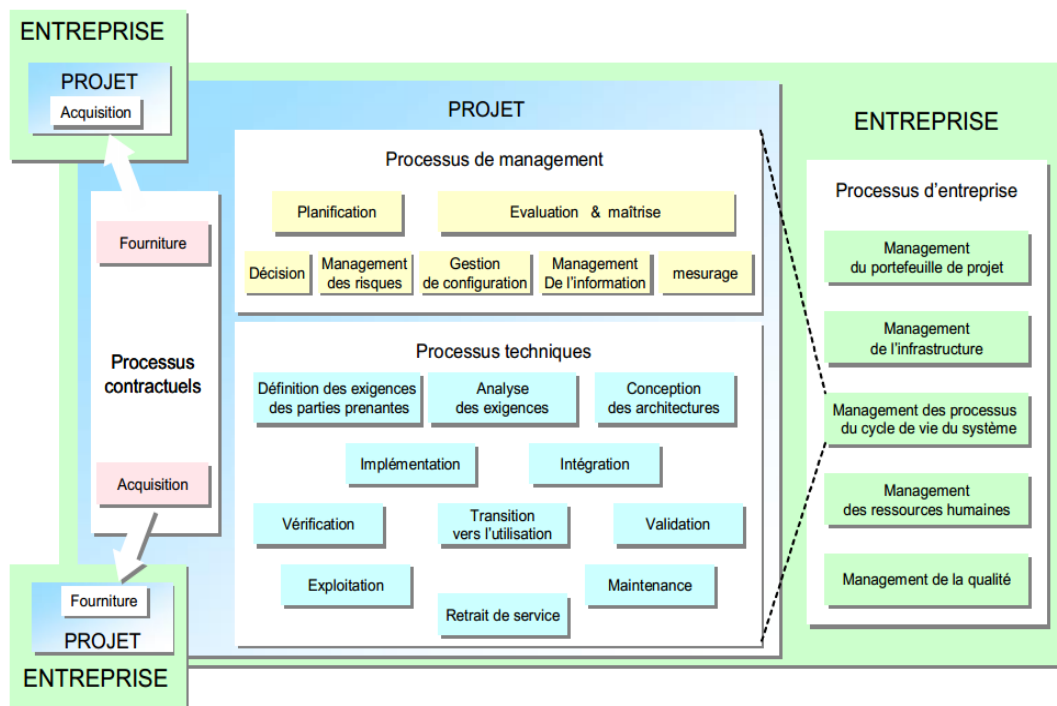


Figure 8 : Cartographie des processus du cycle de vie du système (AFIS 2009).

5.1 Les processus techniques

Les processus techniques de la norme (ISO/IEC/IEEE15288 2015) sont invoqués tout au long des phases du cycle de vie d'un système. Ils sont utilisés pour établir les exigences pour le système comme base pour les efforts visant à créer un produit ou un service efficace ; pour soutenir le système pendant la phase opérationnelle ; et pour soutenir le retrait du système. Sans les processus techniques, le risque d'échec du projet serait trop élevé. Les processus techniques permettent aux ingénieurs système de coordonner les interactions entre les ingénieries spécialistes, les parties prenantes et les opérateurs du système, et la fabrication du système. Ils abordent également la conformité avec les attentes et les exigences des législations de la société. Ces processus conduisent à la création d'un ensemble complet d'exigences qui répondent aux capacités souhaitées dans les limites de la performance, de l'environnement, des interfaces externes, et les contraintes de conception.

- **Processus de définition des besoins des parties prenantes** : L'objectif de ce processus est d'élucider, de négocier, de documenter, et de maintenir les exigences des PPs pour le système d'intérêts dans un environnement défini. Les exigences des PPs gouvernent le développement du système, et sont un facteur essentiel pour définir et clarifier l'objectif du projet de développement.
- **Processus d'analyse des exigences** : L'objectif de ce processus est d'examiner, d'évaluer, de hiérarchiser et d'équilibrer toutes les exigences des PPs et les exigences dérivées (y compris les contraintes) ; et de les transformer en une vue fonctionnelle et technique de la description d'un système capable de répondre aux besoins des PPs. Cette vue peut être exprimée sous forme de spécifications, un ensemble de diagrammes ou tout autre moyen qui permet une communication efficace.
- **Processus de conception de l'architecture** : L'objectif de ce processus est de synthétiser une solution (un système) qui satisfait les exigences. Ce processus exige la participation des ingénieurs système et les ingénieurs spécialisés dans les différents domaines pertinents. Quand des solutions alternatives sont disponibles, des analyses techniques et des décisions sont prises dans le cadre de ce processus pour identifier un ensemble d'éléments du système.
- **Processus d'implémentation** : L'objectif de ce processus est de concevoir, de créer ou de fabriquer un élément du système conforme à la description détaillée de cet élément. L'élément est construit en utilisant les technologies et les pratiques industrielles appropriées. Ce processus représente le lien entre la phase de développement d'un système et celle de sa production.
- **Processus d'intégration** : L'objectif de ce processus est de réaliser le système en combinant progressivement les éléments du système en accordance avec les exigences de conception de l'architecture et la stratégie d'intégration. Ce processus est répété successivement en combinaison avec les processus de vérification et de validation si nécessaire.

- **Processus de vérification** : L'objectif de ce processus est de confirmer que toutes les exigences sont satisfaites par les éléments du système et éventuellement par le système, c.-à-d. le système a été construit correctement. Ce processus établit la procédure pour prendre des mesures correctives en cas de non-conformité.
- **Processus de transition vers l'utilisation** : L'objectif de ce processus est de transférer la garde du système et la responsabilité de maintien d'une entité organisationnelle à une autre. Cela comprend (mais ne se limite pas) le transfert de la garde du système de l'équipe de développement à l'organisation qui exploite et maintient le système.
- **Processus de validation** : L'objectif de ce processus est de confirmer que le système réalisé répond aux exigences des PPs. La validation du système est soumise à l'approbation par l'autorité de projet et les parties prenantes. Ce Processus est invoqué pendant le processus de définition des exigences des PPs afin de confirmer que les exigences reflètent correctement les besoins des PPs et d'établir des critères de validation à savoir que le bon système est construit.
- **Processus d'exploitation** : L'objectif de ce processus est d'utiliser le système pour délivrer ses services. Ce processus est souvent exécuté simultanément avec le processus de maintenance. Il soutient les services du système en fournissant du personnel pour faire fonctionner le système, superviser la performance opérateur système, et la performance du système. Lorsque le système remplace un système existant, il peut être nécessaire de gérer la migration entre les systèmes afin que les PPs pertinentes ne subissent pas une rupture des services. Lorsque le système ou l'un de ses éléments constitutifs atteignent la fin de leur durée de vie prévue ou utile, l'équipe peut initier le processus de retrait de service (voir chapitre 2, section 4.1.6).
- **Processus de maintenance** : L'objectif de ce processus est de supporter le système et de le maintenir pendant la phase d'exploitation. Il comprend les activités pour fournir des opérations de soutien, la logistique et la gestion du matériel. En se basant sur les informations de la supervision de l'environnement opérationnel, les problèmes sont identifiés et corrigés, les actions correctives ou préventives sont prises pour rétablir la capacité du système.
- **Processus de retrait de service** : L'objectif de ce processus est de retirer le système de l'environnement opérationnel avec l'intention de mettre fin à son utilisation de façon permanente ; et de faire face à toutes les matières dangereuses, toxiques ou des déchets conformément aux directives applicables, la politique et les règlements. Le retrait d'un système génère des exigences et des contraintes qui doivent être équilibrées avec les exigences des PPs et autres considérations de conception. Les préoccupations environnementales obligent le concepteur à songer à récupérer les matériaux ou les recycler pour les utiliser dans de nouveaux systèmes.

5.2 Les processus de management de projet

Les activités de management de projet ont pour but de piloter et supporter les autres activités du projet, notamment techniques. Elles sont organisées en processus.

5.2.1. Le processus générique de management de projet

Généralement, le management de projet est décrit de manière générique comme l'application d'un processus général de management aux domaines traditionnels du management de projet. Le processus général de management de projet est fondé sur le cycle PDCA de management des processus. Il est ici appliqué au projet considéré comme un processus. Voici les trois (sous)—processus de management qui découlent des quatre points du cycle PDCA (Figure 9).

- **Plan** : processus de planification du projet.
- **Do** : réalisation des activités techniques selon le plan.
- **Control** : processus de suivi incluant l'évaluation des écarts par rapport au plan.
- **Act**: Processus de maîtrise correspondant aux actions pour maîtriser les écarts constatés.

Les domaines du management de projet à prendre en compte sont, outre la coordination globale qui a un aspect intégrateur des autres domaines, le management du contenu (les tâches), le management des coûts, le management des délais, le management des moyens, le management de la qualité, le management des ressources humaines, le management des risques, le management des données, le management de l'information et de la communication, le management des aspects contractuels et de la réglementation applicable, ainsi que le management des sous-traitances.

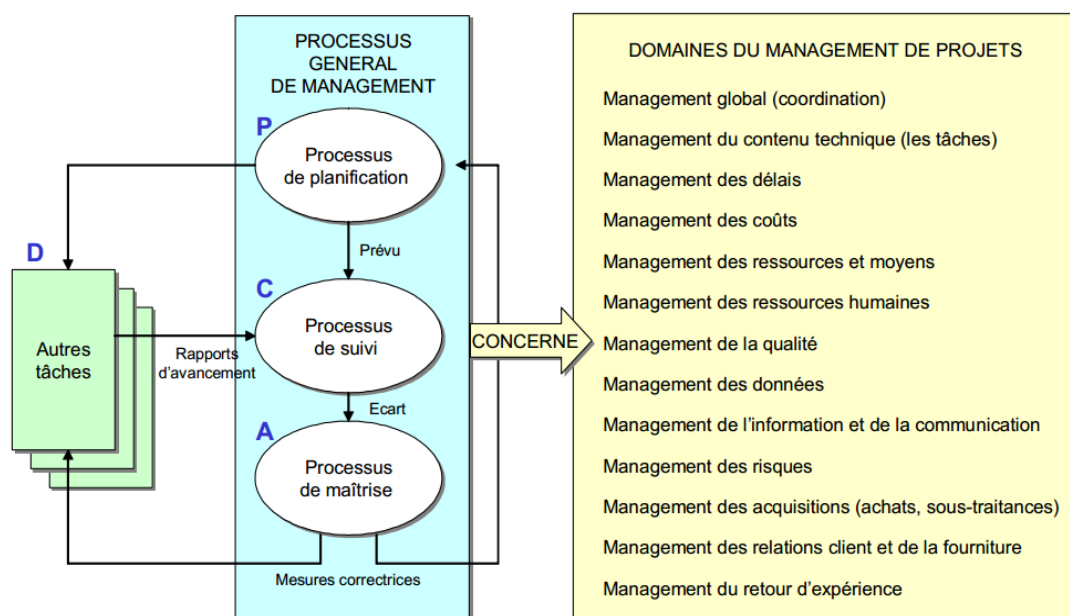


Figure 9 : Processus général de management de projet. (AFIS 2009)

5.2.2. Les processus de management plus spécifiques à l'IS

- **Processus de décision.** Son objectif est de faire les meilleurs choix pour le projet lorsque des alternatives se présentent et/ou que des arbitrages doivent être faits.
- **Processus de management des risques.** Son objectif est de limiter les probabilités d'occurrence et les conséquences d'évènements incertains qui pourraient affecter le système ou le projet.
- **Processus de gestion de la configuration.** Son objet est de maintenir l'intégrité et la cohérence de toutes les entrées et sorties identifiées d'un projet ou d'un processus, et de les rendre disponibles pour toutes les parties concernées.
- **Processus de management de l'information.** Son objet est de fournir aux parties concernées des informations pertinentes, à jour, complètes, valides et, le cas échéant, sécurisées, durant le cycle de vie et même, si nécessaire, au-delà de celui-ci.

5.3. Les processus contractuels

Un projet résulte de la coopération de multiples acteurs : maître d'ouvrage, maître d'œuvre, sous-traitants, équipementiers, fournisseurs de divers types. Un acteur peut lui-même résulter d'un partenariat entre entreprises. Il est nécessaire de définir et contractualiser les obligations respectives des acteurs ou entités concernés par une collaboration.

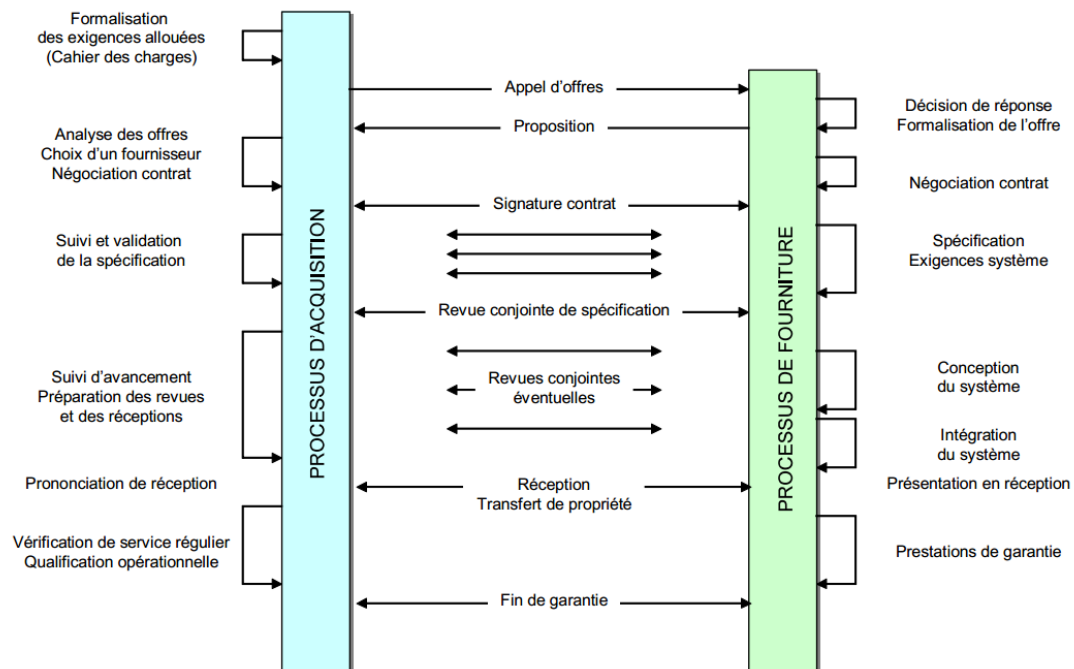


Figure 10 : Exemple de processus contractuels (cas d'un système unique). (AFIS 2009)

La plupart de ces collaborations s'analysent en termes de relations de type acquéreur-fournisseur. Les processus contractuels ont pour objectif de décrire les activités respectives de

l'acquéreur et du fournisseur sous forme de deux processus corrélés : le processus contractuel d'acquisition mené par l'acquéreur, et le processus contractuel de fourniture mené par le fournisseur.

5.4. Les processus de l'entreprise

Les processus d'entreprise ont pour objet de fournir les supports nécessaires aux projets menés dans l'entreprise (l'organisme) et une aide à la décision pour le respect des objectifs de l'entreprise et des contrats établis. Ils répondent notamment aux besoins de développement de l'entreprise, de constitution et de management du potentiel de ressources communes nécessaires aux projets, de coordination inter-projets et, in fine, de management de l'IS dans l'entreprise.

6. Les normes de l'ingénierie système

Les normes de l'IS sont sans doute la partie la plus aboutie de cette discipline. Elles concentrent l'essentiel des avancées méthodologiques dans des référentiels de plus en plus utilisés dans les entreprises. En guidant les praticiens, elles démontrent que l'application de processus de référence structurés conduit à de multiples et importantes améliorations dans la conduite d'un projet : performances, compétitivité, amélioration de la qualité, respects des contraintes de coûts et de délais.

6.1. Objectifs de la normalisation

L'objectif de l'IS, c'est d'abord de maîtriser les processus d'organisation, afin de pouvoir garantir la qualité d'un produit ou d'un service, et donc la satisfaction d'un client. Dans ce cadre, les normes de l'IS définissent des processus sur lesquels peuvent se baser les développeurs dans l'ingénierie ou la réingénierie d'un système. Elles servent de référence pour gérer le système de son concept initial à sa mise à disposition en passant par son développement et sa réalisation. Il ne s'agit pas, pour elles, de définir une des activités d'un service de l'entreprise ou les responsabilités d'une personne, mais de coordonner l'ensemble des activités d'ingénierie de manière à atteindre un but commun.

Ces normes reposent sur l'idée qu'il existe des concepts communs à tous les projets, quel que soit le domaine d'activité ou le système à développer. En identifiant les bonnes pratiques et en assurant la cohérence des activités d'ingénierie, les normes favorisent :

- l'adéquation aux besoins et la qualité des produits,
- l'anticipation des problèmes et la maîtrise des risques concernant tant le projet que le système et son environnement, tout au long du cycle de vie,
- la maîtrise de la complexité des grands systèmes et des produits complexes,
- la tenue des délais et des temps de développement,
- la maîtrise des coûts, avec notamment une anticipation très en amont du coût global du cycle de vie,

- l'efficacité dans la maîtrise de la coopération de la transdisciplinarité et des multiples acteurs,
- la satisfaction de toutes les parties prenantes,
- une meilleure optimisation du compromis global.

6.2. Cadre d'application de la normalisation

Les processus décrits dans les normes peuvent s'appliquer sur l'ensemble du cycle de vie d'un système en incluant la conception, le développement, la production, l'utilisation, le support et le retrait de service. Ils peuvent être appliqués de manière concurrente, itérative ou récursive à un système et à ses composants. Les systèmes considérés peuvent être de petite ou de grande taille, simples ou complexes, uniques ou de grande série, être logiciels, matériels (systèmes physiques), des services ou une composition de ces derniers. Les recommandations des standards s'appliquent aux organisations lorsqu'elles agissent dans le rôle de fournisseur aussi bien que dans celui d'acquéreur. Elles concernent des entités d'une organisation unique ou d'organisations différentes et peuvent aller d'un accord informel à un contrat.

6.3. Limitations d'une normalisation

Le caractère volontairement générique des normes d'IS impose les limitations suivantes :

- Les normes ne détaillent pas les processus du cycle de vie du système en termes de méthodes et de procédures.
- Le format des données n'est pas précisé.
- Dans le cas où l'on souhaite utiliser plusieurs normes simultanément, les conflits éventuels entre recommandations doivent être résolus.

Les processus de cycle de vie décrits par les normes d'IS sont génériques. Les méthodes et procédures qui permettraient de respecter les recommandations des normes ne sont pas données, car elles seront dépendantes soit du cadre de travail de l'entreprise soit du contexte lié au projet. En d'autres termes, les processus décrits dans les standards doivent être complétés d'une part par les politiques et procédures de l'entreprise et d'autre part par des outils qui vont permettre de réaliser les projets. Ainsi, il existe une répartition des responsabilités entre :

- **l'industrie** : qui établit les normes telles que l'ISO/IEC/IEEE-15288, l'EIA632, IEEE1220.
- **L'entreprise** : qui précise les politiques et les procédures *c.-à-d.* les méthodes ;
- **le projet** : qui met en place les outils.

La Figure 11 illustre le besoin de compléter les processus des normes d'IS ainsi que les responsabilités de chacun des acteurs.

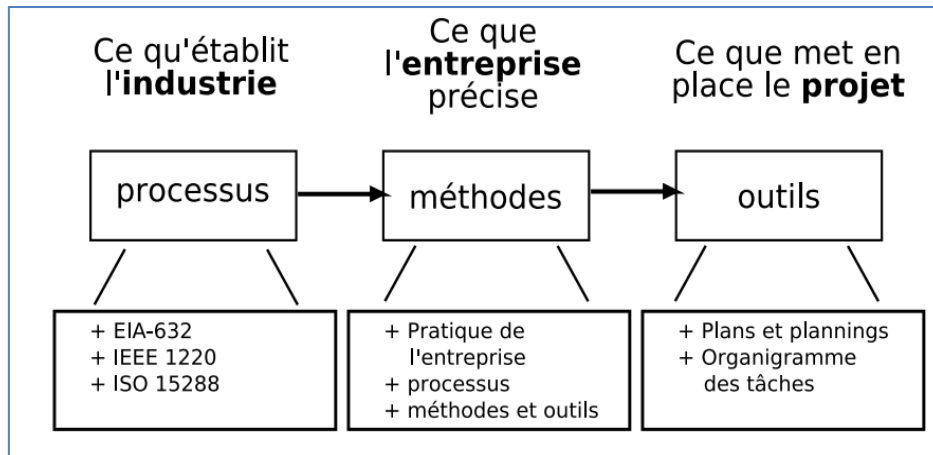


Figure 11 : Les processus, les méthodes et les outils en ingénierie système. (EIA/ANSI 1999)

Les normes d'IS ne spécifient donc pas de méthodes ou d'outils, laissés à la discrétion de l'entreprise. Le rôle de ces normes se limite à :

- **Coordonner les activités des processus :** un standard décrit les grandes lignes des processus, mais doit être interprété pour former un ensemble harmonieux.
- **Établir la vision partagée de la solution système :** la cohérence des exigences et de l'architecture est maintenue et l'effort technique guidé.

6.4. Les principales normes de l'ingénierie système

Le travail des associations d'IS, comme l'INCOSE et l'AFIS, mené en collaboration avec de grands organismes de normalisation comme l'AFNOR, l'ANSI ou l'IEEE a conduit à la rédaction des grandes normes d'IS. Trois de ces normes sont les plus utilisées :

- IEEE-1220: Standard for application and Management of the Systems Engineering Process.
- EIA/ANSI-632: Processes for Engineering a System.
- ISO/IEC/IEEE-15288: Systems and Software Engineering – System Life Cycle Processes.

Ces standards décrivent tout le déroulement des projets en termes de processus en dressant une liste d'activités nécessaires au bon déroulement du projet. Le résultat attendu de ces activités ainsi que des exigences sur leur mise en application complètent ces descriptions. Ces normes définissent, chacune, un processus d'entreprise de référence.

Ce choix d'une approche par processus se base sur l'idée que les types d'activité à réaliser sont invariants par rapport aux projets et secteurs d'application. Il est donc naturel, pour les organismes de normalisation, de définir des processus plutôt que des cycles de vie. Cette approche a pour avantage de donner une cohérence aux projets inter-entreprises et de laisser les projets définir l'application des processus dans le cadre des cycles de vie qu'ils choisissent.

Parce que les trois normes d'IS sont disponibles et utilisées dans la pratique, il est important de reconnaître au moins la distinction entre eux. La Figure 12 montre l'historique de ces normes, on voit sur cette figure que la norme ISO/IEC/IEEE-15288 qui a apparue en (2002) est la plus mature c'est ce qui explique sa large utilisation, cette norme a comme origine la norme EIA/ANSI-632 qui a apparue en même temps avec la norme IEEE-1220 en (1994), ces deux normes ont la même origine. La Figure 13 montre que les trois normes d'IS diffèrent principalement sur les types de processus qu'elles décrivent : en découle une couverture des activités techniques sur le cycle de vie du système différente pour chacune d'entre elles avec plus ou moins de détails.

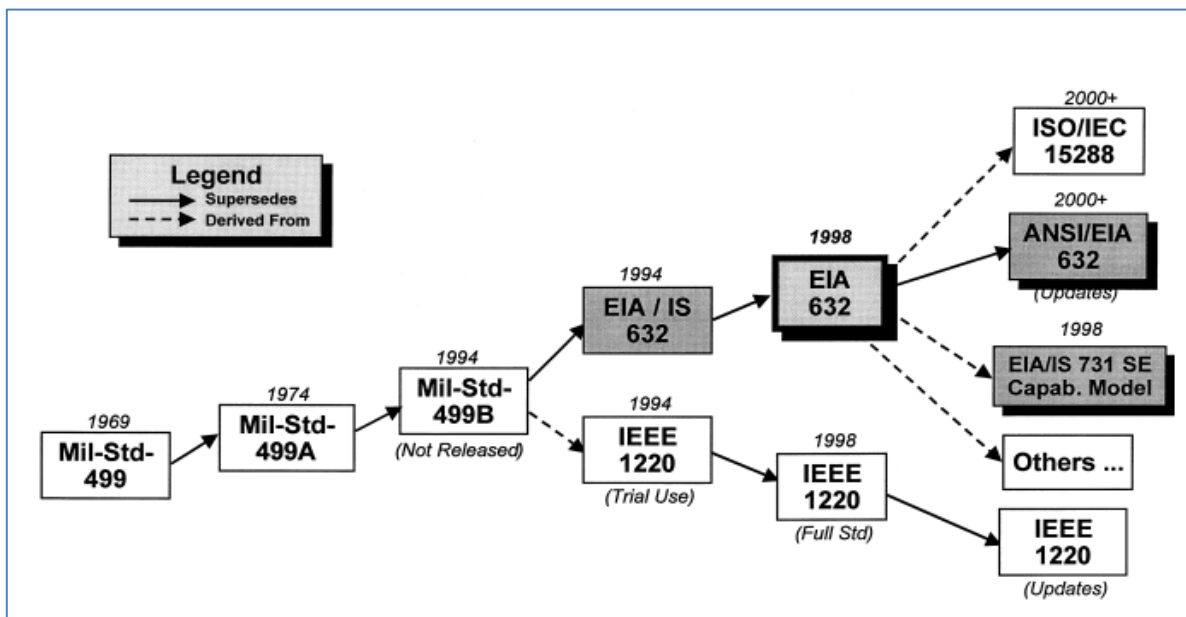


Figure 12 : L'héritage des normes de l'ingénierie système.

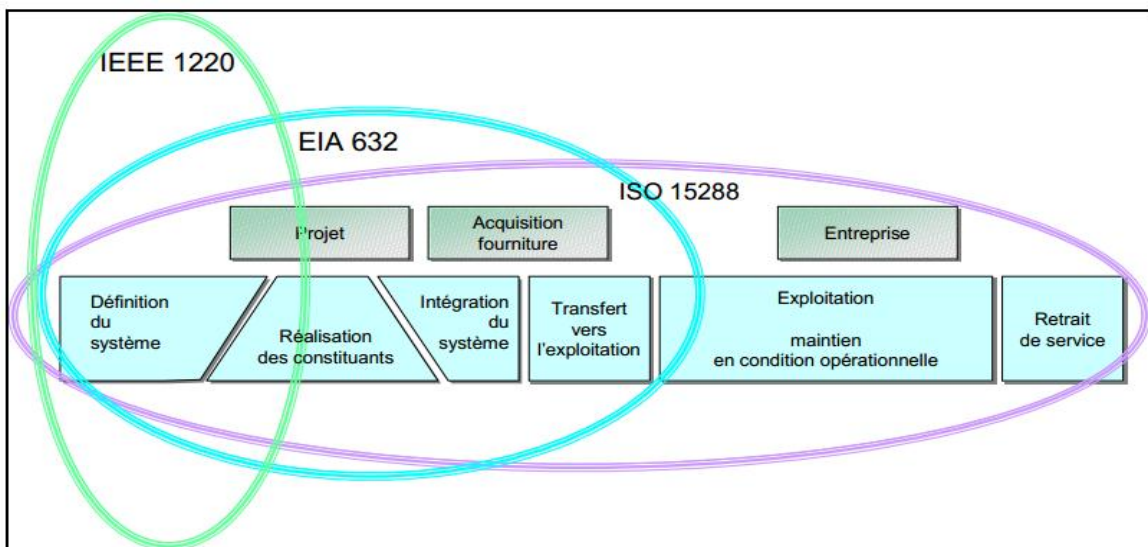


Figure 13 : La couverture du cycle de vie du système par les 3 normes principales d'IS. (AFIS 2009)

6.5. La norme IEEE-1220

Le standard 'IEEE-1220 : *Standard for application and Management of the Systems Engineering Process*' définit les tâches interdisciplinaires requises pour transformer, au long du cycle de vie du système, les besoins des parties prenantes, exigences et contraintes en une solution système. Ce standard est resté relativement stable depuis sa première version de 1994. Il se décompose en huit processus représentés par la Figure 14.

- Les processus : analyse des exigences, validation des exigences, analyse fonctionnelle, vérification fonctionnelle, synthèse, vérification de la conception s'adressent à la transformation des exigences en une solution de conception.
- Le processus d'analyse du système est un support à la résolution des conflits entre exigences, décomposition fonctionnelle, allocation de performances, sélection de solutions de conception, appréciation de l'efficacité du système et gestion des risques.
- Le processus de contrôle concerne la gestion et la documentation des activités d'IS.

La norme IEEE-1220 se focalise sur les processus techniques d'IS allant de l'analyse des exigences jusqu'à la définition physique du système. La portée de cette norme dans le cycle de vie du système est limitée à la phase de définition du système (voir Figure 13).

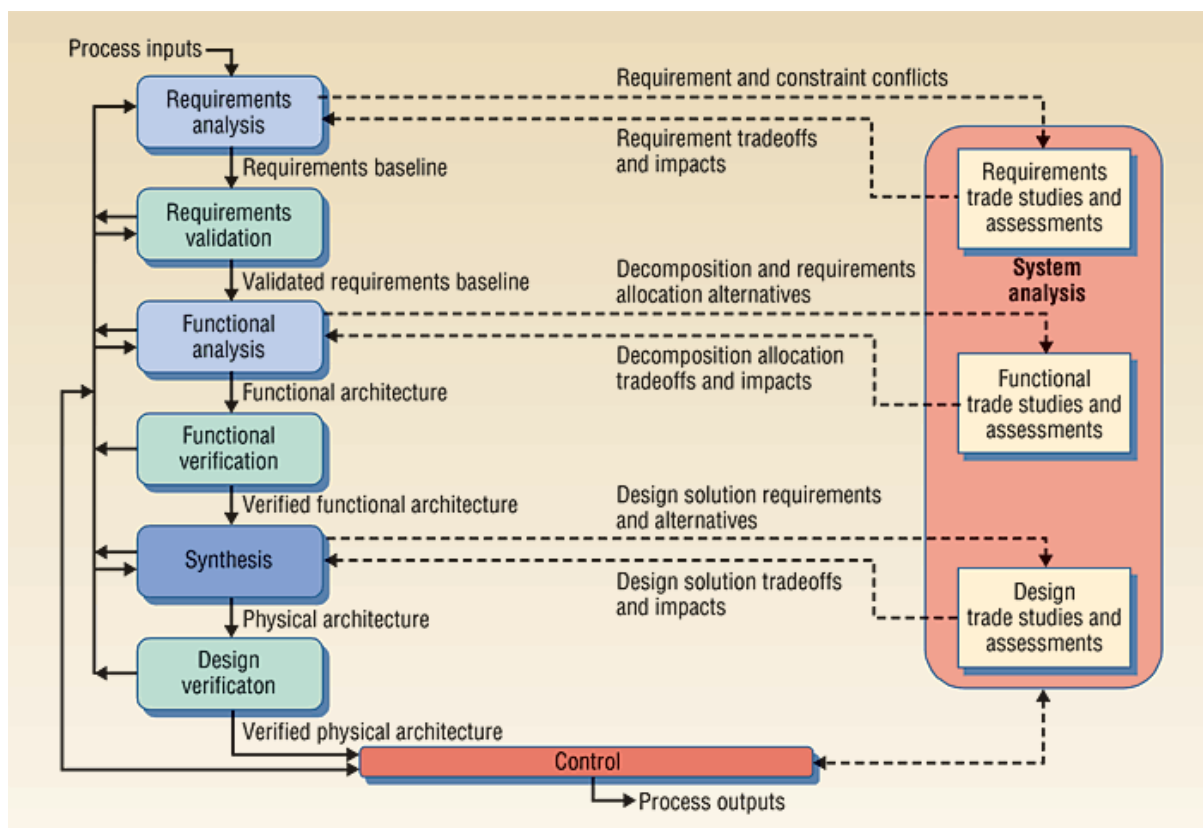


Figure 14 : Les processus de la norme IEEE-1220. (IEEE 2005)

6.6. La norme EIA/ANSI-632

Ce standard est en 1994 et en 1995, l'Electronic Industries Alliance ou l'EIA lance un groupe d'étude pour l'étendre. En 1998, la norme 'EIA/ANSI-632 : *Processes for Engineering a System*' est diffusée. Elle étend le champ de la norme précédente à tous les processus techniques d'IS. Cette norme de l'EIA complète les processus techniques de définition du système en couvrant sa réalisation jusqu'à sa mise en service (transfert vers l'utilisation). De plus elle incorpore les processus contractuels d'acquisition et de fourniture. La Figure 15 montre que les processus techniques et processus contractuels sont encadrés :

- Par les **processus de management** (selon leur forme traditionnelle avec les trois sous-processus de planification, évaluation, pilotage)
- Et par les **processus d'évaluation** des résultats des activités (processus de **vérification** vérifiant que l'activité a été bien faite et processus de **validation** vérifiant que le résultat répond au besoin, les deux justifiant de la conformité, ainsi que le processus d'**analyse système** justifiant des choix réalisés tout au long de la définition et donc de l'optimisation du système).

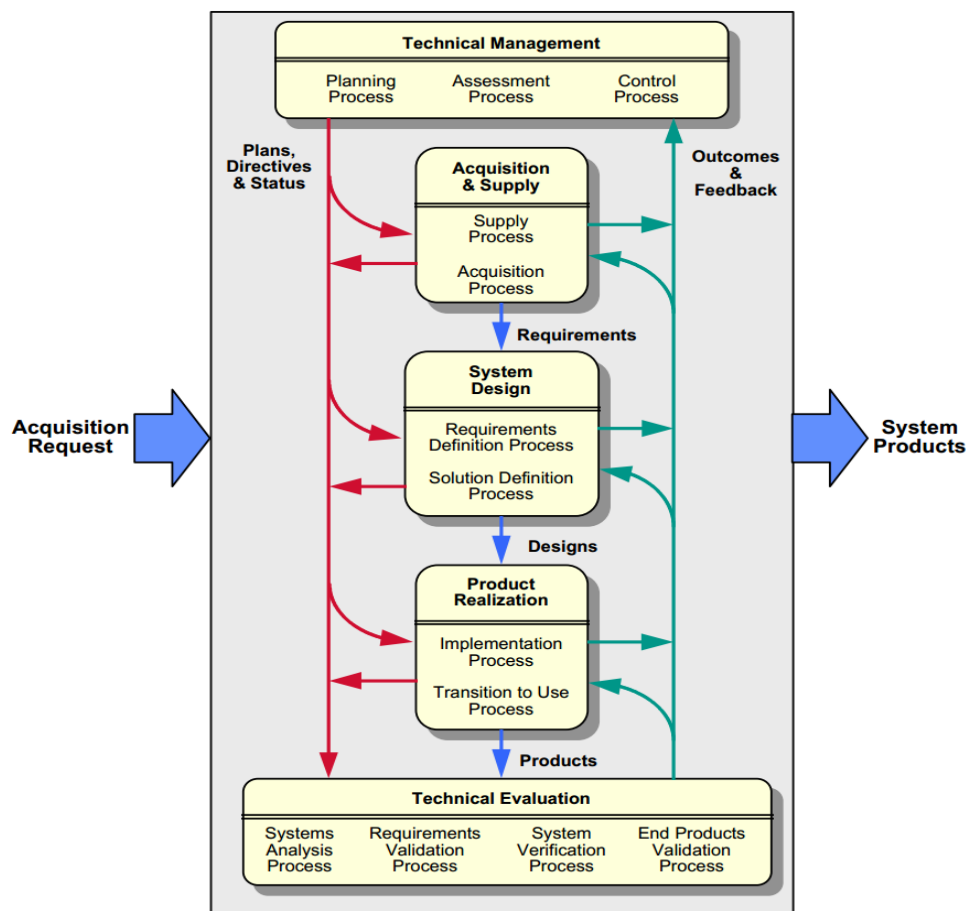


Figure 15 : Les processus de la norme EIA/ANSI-632. (EIA/ANSI 1999)

6.7. La norme ISO/IEC/IEEE-15288

Le standard ‘ISO/IEC/IEEE-15288 : *System Life Cycle Processes*’ est le premier standard de l’organisation internationale de normalisation à traiter les processus de cycle de vie des systèmes. Paru en octobre 2002, plus tardif que les deux précédents (IEEE-1220, EIA/ANSI-632), ce standard s’adresse à l’ensemble du cycle de vie d’un système industriel. La totalité du cycle de vie du système est couverte par cette norme (voir Figure 13). Elle inclut les processus d’exploitation, de maintien en conditions opérationnelles et de retrait de service.

La norme ISO/IEC/IEEE-15288 s’applique aussi à l’ingénierie des systèmes contributeurs qui ont leur propre cycle de vie (systèmes de fabrication, de déploiement, de soutien logistique, de retrait de service). Elle complète les processus s’appliquant aux projets par des processus d’entreprise qui ont pour objectif de développer le potentiel de l’organisme en IS en gérant les domaines communs aux différents projets d’IS. En ce sens, elle est complémentaire aux deux normes précédentes et peut être utilisée de manière conjointe avec ces normes. Les processus considérés dans cette norme sont (voir Figure 16) :

- **les processus techniques** qui participent à la transformation des besoins en solution,
- **les processus de management** qui participent à la maîtrise du projet,
- **les processus contractuels** qui assurent les relations avec le ou les clients et les sous-traitants,
- **les processus d’entreprise** qui ont pour rôle de développer le potentiel en IS de l’entreprise en gérant les domaines communs aux différents projets d’IS.

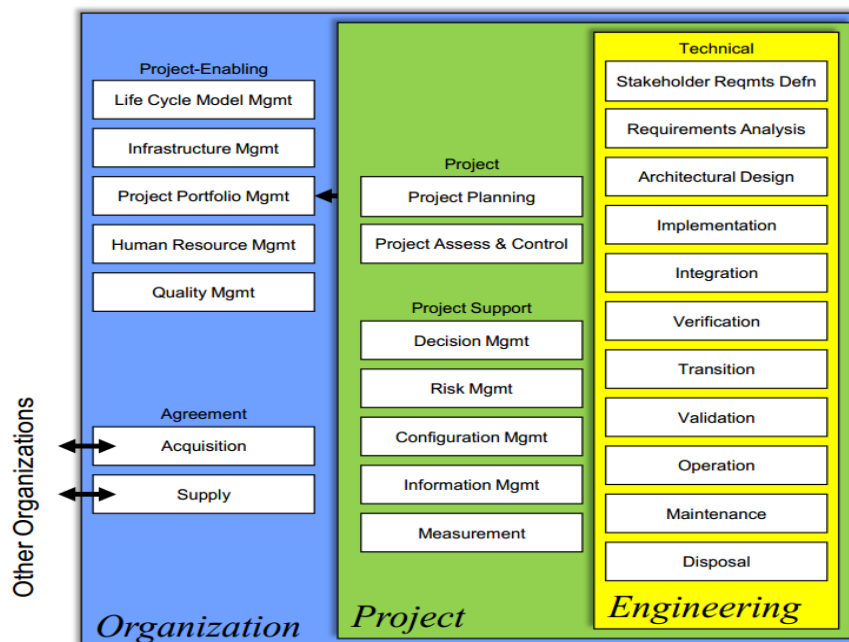


Figure 16 : Les processus de la norme ISO/IEC/IEEE-15288. (ISO/IEC/IEEE15288 2015)

7. Les approches de développement d'un système

Des organisations différentes peuvent entreprendre différentes étapes du cycle de vie d'un système. Cependant, chaque étape est effectuée par l'organisme responsable correspondant en tenant en compte des informations disponibles sur les plans du cycle de vie et les décisions prises dans les étapes précédentes. De même, l'organisation responsable de cette étape enregistre les décisions prises et enregistre les hypothèses concernant les étapes ultérieures du cycle de vie. (ISO/IEC/IEEE15288 2015)

La discussion des étapes du cycle de vie du système ne signifie pas que le projet doit suivre un ensemble prédéterminé d'activités ou des processus à moins qu'ils ajoutent de la valeur en vue d'atteindre l'objectif final. Généralement, les représentations graphiques des phases du cycle de vie du système ont tendance à être linéaires, mais cela masque la véritable nature incrémentale et itérative des processus principaux. Les illustrations qui suivent impliquent la liberté complète de choisir un modèle de développement et ne sont pas limitées à un modèle en cascade ou un modèle en spirale. Pour la phase de développement, comme pour toutes les autres phases, l'organisation sélectionnera les processus et les activités qui répondent aux besoins du projet.

Un système est développé selon un processus, sachant qu'un processus est un ensemble d'activités reliées entre eux et exécutées dans un ordre bien déterminé pour atteindre un objectif précis, donc la modélisation ou la représentation de ce processus sous forme d'un diagramme est nécessaire pour faciliter sa compréhension et son application. Le modèle ou plus précisément le diagramme doit montrer : l'enchaînement des activités du processus, les entrées et les sorties de chaque activité, les flux échangés et les relations entre les différentes activités.

Dans la littérature, plusieurs modèles de cycle de vie du processus de développement ont été créés et appliqués sur les projets de développement des systèmes physiques et logiciels à grande échelle dans le gouvernement, l'industrie et l'académie. Estefan (Estefan 2008) regroupe la plupart de ces modèles en trois familles : le modèle en cascade, le modèle en spirale, et le modèle en 'V'. Ces trois types de modèles peuvent être utilisés comme des méta-modèles pour le cycle de développement. Autrement dit, ils fournissent des modèles qui peuvent être utilisés pour développer des modèles du processus de développement pour un domaine spécifique. Par exemple, la plupart des méthodologies MBSE décrites ultérieurement dans ce document adoptent l'un de ces modèles de développement.

7.1. Le modèle en cascade

Le modèle en cascade est spécialement utilisé pour le développement des systèmes logiciels. Il a été mentionné pour la première fois dans l'article de Royce dans (Royce 1970). Royce a démontré que le développement des grands systèmes logiciels nécessite une approche de développement plus approfondie parce que les approches non itératives (un seul passage) présentent un risque inhérent. Il a

présenté une approche incrémentale et itérative préconisant que les projets doivent passer au moins deux fois par le cycle de cette approche. Selon Royce, un modèle sans itération est risqué et conduit à l'échec. Comme alternative, Royce a proposé un développement plus progressif, où chaque étape suivante est liée à l'étape qui la précède.

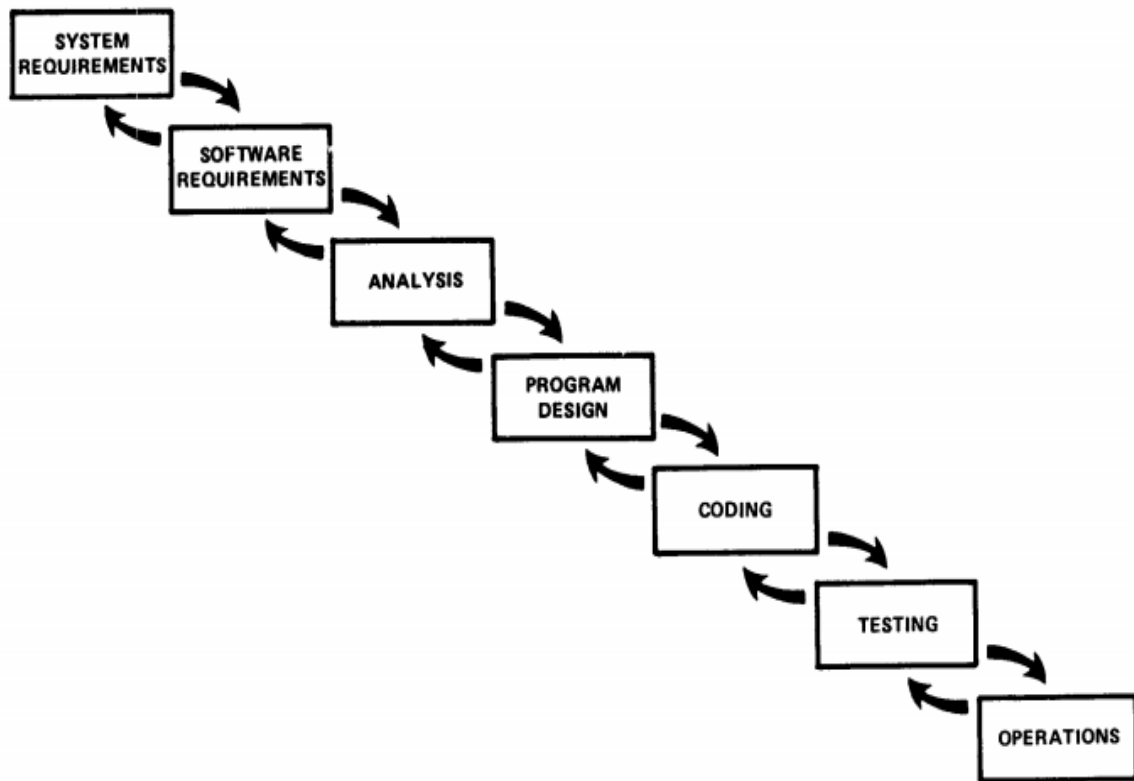


Figure 17 : Le modèle de développement en cascade selon Royce. (Royce 1970)

La Figure 17 montre le modèle en cascade. C'est un processus linéaire organisé en phases individuelles fixes et les résultats de chaque phase sont considérés comme les entrées de la phase suivante. Chaque phase doit être complètement terminée pour pouvoir commencer la phase suivante. Dans le modèle de base, le retour à la phase précédente n'est pas possible. Le diagramme dans la Figure 17 présente une version améliorée du modèle en cascade, le retour ici est possible.

Le modèle en cascade représente le déroulement des projets physiques réels. Dans ce modèle, la correction devient plus difficile et plus coûteuse si une spécification incorrecte est identifiée plus tard. Pour limiter les risques, la seule solution est la décomposition du projet en phases. Ceci est un vrai problème dans le cas de développement des systèmes complexes puisqu'une itération complète du modèle dans ce cas pourrait prendre plusieurs années. En plus, à la fin de la phase de conception, il est pratiquement impossible de répondre à de nouveaux impératifs (changements des exigences). En raison des possibilités très limitées de changements des résultats des phases terminées, une flexibilité minimale est possible en termes de changements des exigences. Ceci est un vrai problème dans le cas

de développement des systèmes complexes qui opèrent dans un environnement variable comme les systèmes manufacturiers reconfigurables.

Le modèle en cascade est généralement destiné aux projets où les exigences, les performances et les processus sont clairement définis et aussi aux projets qui ne nécessitent pas de changements au cours du développement. Si les exigences changent alors qu'on est dans une phase tardive du projet, on aura un produit qui ne répondra pas aux exigences et la résolution de ce problème nécessite d'abandonner le développement en cours et de reprendre à partir de la base. Cela n'est pas très pratique pour le développement des systèmes de production reconfigurables où la flexibilité et la réactivité aux changements du marché sont cruciales et essentielles pour la survie de l'entreprise. Même dans son article (Royce 1970), Royce a décrit explicitement que son modèle est seulement souhaité pour le développement des projets simples. Donc, ce type de modèle, nous ne permettra pas de développer des systèmes agiles et reconfigurables. Il est impossible de développer une méthodologie qui se base sur ce type de modèle et qui pourra supporter les modifications d'une façon plus rapide et plus efficace.

7.2. Le modèle en spirale

Le modèle en spirale a été décrit la première fois par Barry Boehm dans son article de 1986 (Boehm 1986) et puis dans son article de 1988 (Boehm 1988). Il est similaire au modèle incrémental, avec plus de concentration sur l'analyse des risques. Il est composé de quatre phases : planification, analyse des risques, ingénierie et évaluation. Un projet logiciel passe par ces phases d'une façon répétitive en plusieurs itérations. La ligne de départ de la spirale commence dans la phase de planification, les exigences sont collectées et les risques sont évalués. La spirale suivante est construite sur la spirale qu'il la précède.

- **La phase de planification** : les exigences sont collectées pendant la phase de planification.
- **La phase d'analyse** : les risques et les solutions alternatives sont identifiés, un prototype est construit à la fin de cette phase. Si un risque est identifié, des solutions alternatives sont suggérées et implémentées.
- **La phase d'ingénierie** : le système (logiciel) est développé et testé à la fin de la phase.
- **La phase d'évaluation** : cette phase permet au client d'évaluer les sorties du projet afin de les valider avant de passer à la spirale suivante.

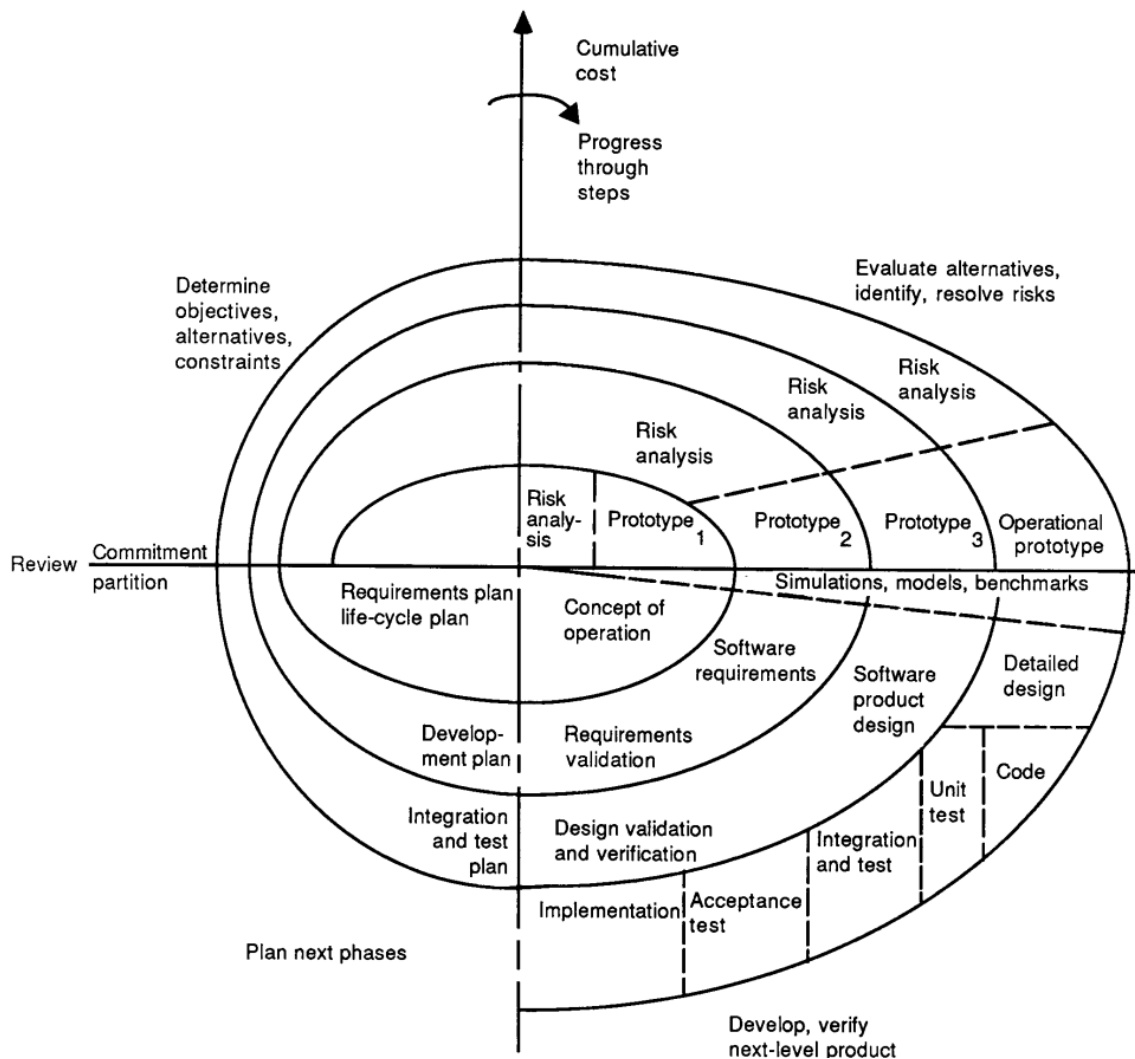


Figure 18 : Le modèle en spirale. (Boehm 1988)

Le modèle en spirale permet une analyse intensive des risques, donc l'évitement des risques est amélioré, il est efficace pour les grands projets et les projets critiques, il exige une forte approbation et contrôle de la documentation, et des fonctionnalités additionnelles peuvent être ajoutées ultérieurement. En revanche, le modèle en spirale est spécifiquement destiné pour le développement des logiciels, donc son adaptation pour développer des systèmes physiques est généralement impossible, à notre connaissance, il n'existe pas une méthode qui utilise ce modèle pour développer les systèmes physiques. Le modèle en spirale implique des coûts de développement élevés, il nécessite une expertise très spécifique pour l'analyse des risques, il n'est pas efficace sur les projets de petite taille, et le succès du projet dépend fortement de la phase d'analyse des risques. Généralement le modèle en spirale est utilisé pour le développement des systèmes logiciels dans le cas où :

- l'évaluation des risques et des coûts est importante,
- les projets présentent des risques moyens ou forts,
- les utilisateurs ne sont pas sûrs de leurs besoins,

- les exigences sont complexes,
- des changements significatifs sont prévus,
- on veut développer une nouvelle gamme de produits.

7.3. Le modèle en ‘V’

Le modèle en ‘V’ (AFIS 2009) aussi connu sous ‘le modèle de vérification et de validation’ est une représentation graphique du cycle de vie de développement d’un système. Issu du monde de l’industrie, le cycle en ‘V’ est devenu un standard de l’industrie logicielle depuis les années 1980, il résume les étapes générales qui doivent être exécutées pour développer un système. Le ‘V’ représente la séquence des étapes dans le cycle de vie de développement d’un projet ou d’un système. Il décrit les activités qui doivent être exécutées et les résultats qui doivent être fournis pendant le développement du système. La partie gauche du cycle (partie descendante) représente la décomposition des exigences, et la création des spécifications du système. La partie droite du cycle (partie ascendante) représente l’intégration des différentes parties du système (composants, sous-système) et leur validation. La vérification du système ou de ses composants est toujours accomplie par rapport aux exigences (exigences techniques). Par contre la validation du système ou de l’un de ses composants est toujours accomplie par rapport aux besoins des utilisateurs (fonctionnement dans un environnement réel).

Exactement comme le modèle en cascade, chaque phase doit être complétée avant de commencer la suivante (pas de concurrence). Le test du système est planifié en parallèle avec la phase de développement correspondante. Le modèle en ‘V’ est une extension du modèle en cascade et il basé sur l’association d’une phase de test pour chaque phase de développement correspondante. Cela signifie que pour chaque phase dans le cycle de développement, il y a directement une phase de test associée. Dans le modèle en ‘V’, la phase de test qui correspond à la phase de développement est planifiée ou préparée en parallèle. Donc, sur un côté du modèle en ‘V’, on a les phases de vérification et de validation (partie ascendante du cycle) et sur l’autre côté, on a les phases de développement (partie descendante du cycle). La phase de réalisation se situe à l’intersection de la phase de développement et la phase de vérification et validation.

Le modèle en ‘V’ est utilisé pour visualiser les processus de l’IS, en particulier pendant les étapes de la conception et de développement. Ce modèle souligne la nécessité de définir des plans de vérification au cours de la définition des besoins, spécification des exigences et conception de la solution. (Haskins 2006)

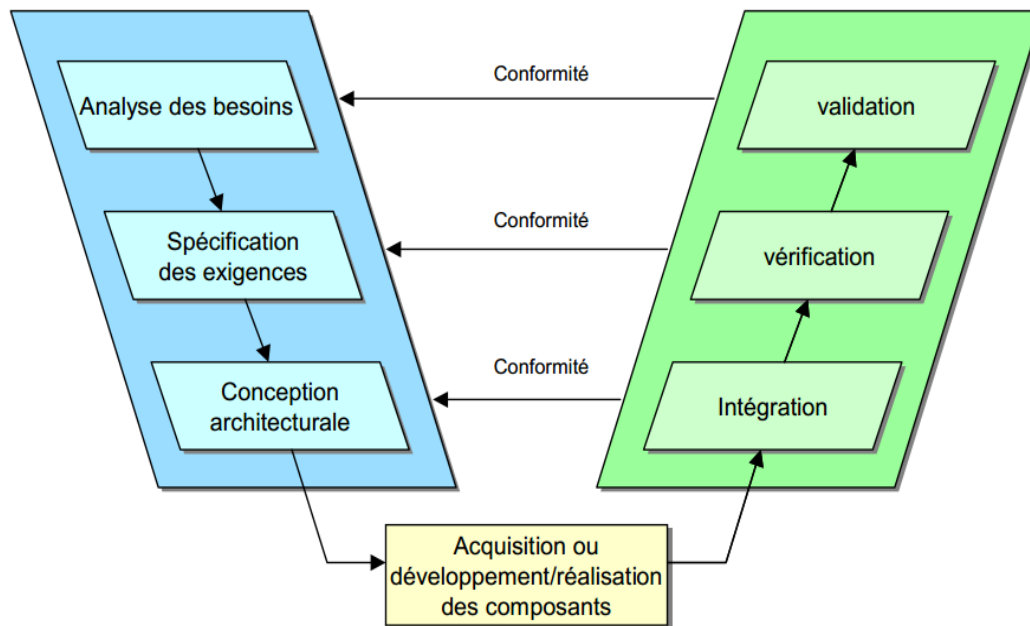


Figure 19 : le modèle en 'V'. (AFIS 2009)

Sans entrer dans les détails, les différentes phases du modèle en V sont les suivantes :

- **Les exigences (expression de besoins et spécifications des exigences)** : exactement comme dans le modèle en cascade, le modèle en 'V' commence par l'identification des exigences. Le plan de test du système doit être créé une fois que les exigences sont identifiées. Ce plan concentre sur la satisfaction des fonctionnalités spécifiées dans les exigences collectées (**validation du système**).
- **La conception générale** : cette phase se concentre sur l'architecture et la conception du système. Elle fournit une vue globale de la solution, la plate-forme, le système, les produits et les services. Un plan de **test d'intégration** est créé pendant cette phase afin de tester la capacité des différentes parties du système de fonctionner ensemble.
- **La conception détaillée** : cette phase est la phase où les composants du système sont conçus. Les plans de test des composants (**tests unitaires**) sont créés pendant cette phase.
- **La réalisation** : pendant cette phase, les composants du système conçus dans la phase précédente sont réalisés (soit développés localement, soit achetés, soit déjà disponibles).

Les avantages du modèle en 'V'

Le modèle en 'V' guide la planification et la réalisation des systèmes. Les objectifs de ce modèle sont les suivants :

- **Simplicité** : Le modèle en 'V' est aussi simple et facile à utiliser. Le fait de planifier les tests avant la réalisation du système réduira énormément le temps de développement, ce qui

implique des chances de réussite plus élevées par rapport à d'autres modèles (comme le modèle en cascade). Les défauts sont détectés en amont pendant le développement, ce qui minimisera les risques associés à ces défauts et évitera leur propagation.

- **Minimisation des risques du projet** : le modèle en 'V' améliore la transparence et le contrôle du projet par la spécification des approches standards et par la description des résultats correspondants aux rôles des responsables. Il permet une reconnaissance précoce des écarts de planification, des risques, et améliore la gestion du projet, ainsi réduisant les risques du projet.
- **Amélioration et assurance de la qualité** : le modèle en 'V' assure la complétude et la qualité des résultats fournis. La définition des résultats intermédiaires peut être vérifiée en avance. Le contenu des produits uniformes va améliorer la lisibilité, la compréhension et la vérifiabilité.
- **La réduction du coût sur tout le cycle de vie du projet et du système** : l'effort de développement, de production, d'opération et de la maintenance du système peut être calculé, estimé et contrôlé avec transparence en appliquant un modèle de processus standard. Les résultats obtenus sont uniformes et facilement retracés. Cela réduira la dépendance de l'acquéreur, du fournisseur, et les efforts pour les activités et les projets suivants.
- **Amélioration de la communication entre les parties prenantes** : la description uniforme et standardisée de tous les éléments pertinents et les termes est la base d'une compréhension mutuelle entre toutes les parties prenantes. Ainsi, le manque de communication entre l'utilisateur, l'acquéreur et le développeur est réduit.

Les inconvénients du modèle en 'V' :

Le modèle en 'V' n'est pas flexible, si des changements se produisent au milieu du cycle, donc les documents de test et les documents des exigences doivent être mis à jour. Il ne permet pas d'avoir des prototypes du système à des phases antérieures, il faut attendre la phase de réalisation du système lui-même.

Quand utiliser le modèle en 'V'

- Le modèle en 'V' doit être utilisé pour le développement des systèmes de petite et moyenne taille où les exigences sont fixes et clairement définies.
- Le modèle en 'V' doit être utilisé quand on dispose amplement de ressources et de l'expertise technique suffisante.
- Haut niveau de confiance du client est nécessaire pour choisir l'approche du modèle en 'V' puisque si aucun prototype n'est produit, il y a un risque très élevé impliqué dans la satisfaction des attentes du client.

7.4. Le modèle en double ‘V’

Le modèle en double ‘V’ (Clark 2009) construit à partir du modèle en ‘V’ pour montrer clairement la complexité associée à la conception et au développement des systèmes complexes. En IS, le modèle en double ‘V’ représente un processus uniforme pour le développement des systèmes ou des projets. Le modèle montre un développement simultané de l’architecture du système et de ses entités selon le modèle en ‘V’. Donc, il y a une intersection entre le cycle en ‘V’ de chaque entité et le cycle en ‘V’ de l’architecture du système (voir Figure 20). Cela montre les interactions et l’enchaînement dans le développement des systèmes complexes.

Le modèle en double ‘V’ est construit à la base du modèle en ‘V’ pour gérer les systèmes complexes. Le cycle en ‘V’ principal gère le système et les branches de ce cycle gèrent les sous-systèmes du système principal. Par exemple, le système GPS (Global Positioning System) est constitué d’une constellation de satellites, un réseau de contrôle au sol, et des millions d’utilisateurs dans le monde. Chaque satellite, chaque centre de contrôle, et chaque récepteur GPS est un système complexe qui pourrait être géré par un processus en cycle en ‘V’ séparément des autres. Le développement du satellite peut affecter la conception, la production ou le prix des récepteurs. Similairement, le développement du récepteur peut affecter la conception, la production, ou le prix des satellites. Donc, le tout doit être intégré dans un système de systèmes qui est développé avec une architecture en ‘V’ plus large.

Faire évoluer les besoins de l’utilisateur en un système qui répond à ces besoins exige une meilleure solution pour chaque entité de l’architecture. Cela peut être visualisé en positionnant les cycles de développement en ‘V’ des entités du système orthogonalement sur le cycle de développement en ‘V’ de l’architecture du système (Figure 20). Chaque entité de l’architecture est développée et réalisée selon un processus en cycle en ‘V’.

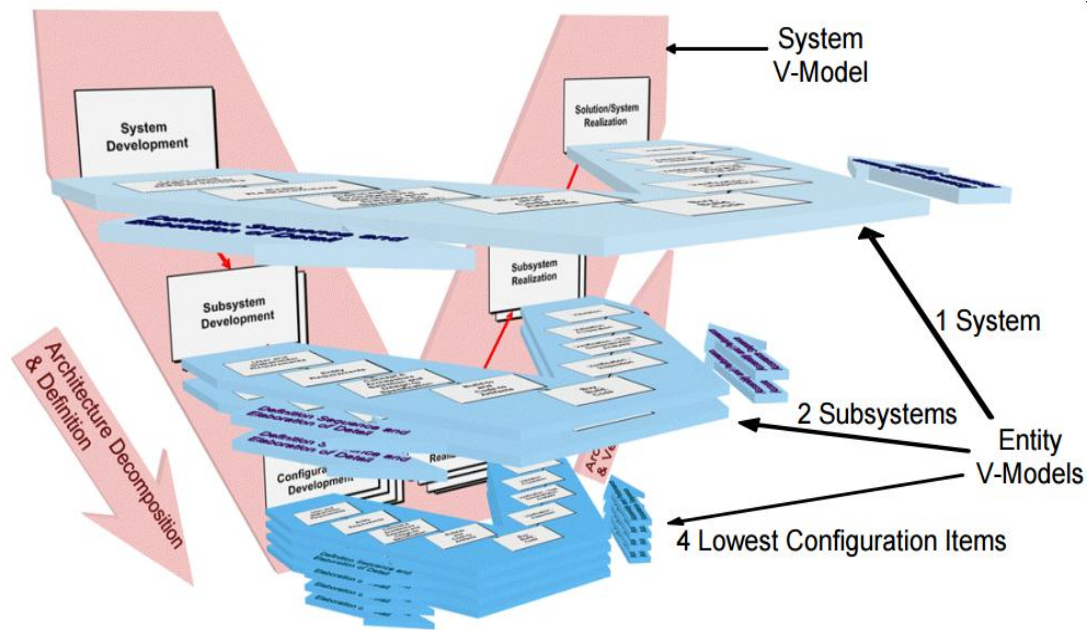


Figure 20 : Représentation du modèle en double V. (Clark 2009)

8. L'Ingénierie système dirigée par les modèles MBSE

8.1. MBSE définition

Selon l'INCOSE l'IS dirigée par les modèles communément connue sous le nom MBSE (Model Based Systems Engineering) est l'application formelle de la modélisation pour soutenir les différentes activités (la définition des exigences, la conception fonctionnelle et architecturale, l'analyse, la vérification et la validation) d'ingénierie système pour développer un système de sa phase de conception jusqu'aux dernières phases de son cycle de vie.

La MBSE fait partie d'une tendance à long terme vers des approches centrées sur les modèles adoptées par d'autres disciplines de l'ingénierie, y compris la mécanique, l'électrique et le logiciel. En particulier, la MBSE est prévue pour remplacer l'approche centrée sur les documents qui a été pratiquée par les ingénieurs système dans le passé et d'influencer la pratique future de l'IS en étant pleinement intégrée dans la définition des processus d'IS.

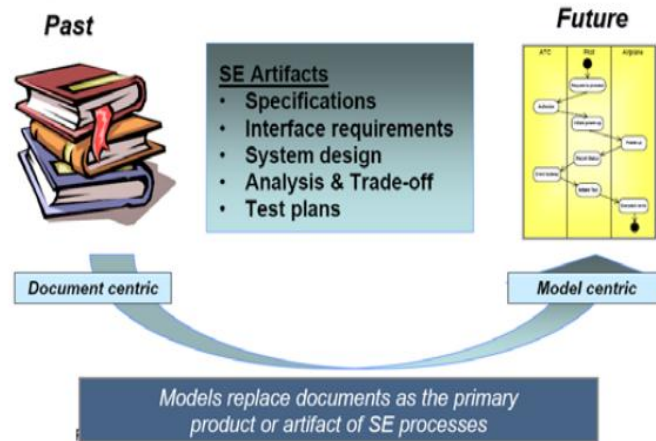


Figure 21 : La transition de l'IS vers l'approche dirigée par les modèles (MBSE) (Murray 2012).

Murray (Murray 2012) décrit la MBSE comme étant un terme générique qui décrit une approche de l'IS qui :

- définit un modèle d'architecture du système comme l'artefact primaire de travail dans l'ensemble du cycle de vie de développement du système,
- combine les meilleures pratiques de l'IS traditionnelle avec des techniques de modélisation visuelles rigoureuses,
- utilise généralement le langage SysML comme un langage standard de modélisation visuelle. Ce langage non seulement soutient les techniques de modélisation rigoureuses entre les ingénieurs des systèmes, il sert aussi de moyen de communication entre les divers types d'ingénieurs (des informaticiens, des électriciens, des mécaniciens, etc.)

8.2. Les avantages des approches MBSE

Les approches MBSE améliorent la capacité de capturer, analyser, partager et gérer les informations associées à la spécification complète d'un système, ce qui entraîne les avantages suivants :

- **L'amélioration de la communication** entre les acteurs du développement (par exemple, les clients, les managers, les ingénieurs système, les développeurs des parties physiques et logiques, les testeurs, et les autres disciplines spécialisées d'ingénierie).
- **Augmentation de la capacité à gérer la complexité** du système en permettant plusieurs points de vue du modèle du système, et d'analyser l'impact des changements.
- **Amélioration de la qualité** du système en fournissant un modèle univoque et précis du système qui peut être évalué pour la cohérence, l'exactitude et l'exhaustivité.
- **Amélioration de l'acquisition des connaissances** et la réutilisation de l'information par la capture des informations de façon plus standardisée et l'exploitation intégrée dans les

mécanismes d'abstraction inhérents à des approches dirigées par les modèles. Cela peut entraîner des temps de cycle réduit et des coûts de maintenance faibles pour modifier la conception.

- **Amélioration de la capacité d'enseigner et d'apprendre les fondamentaux** d'IS en fournissant une représentation claire et sans ambiguïté des concepts.

8.3. Les méthodologies MBSE

Une méthodologie peut être définie comme un ensemble relié de processus, de méthodes et d'outils utilisés pour soutenir une discipline spécifique. Cette notion générale de méthodologie peut être adaptée à la méthodologie MBSE, qui est définie comme un ensemble relié de processus, de méthodes et d'outils utilisés pour soutenir la discipline d'IS.

En 2007, une enquête formelle des principales méthodologies MBSE a été publiée dans le cadre du travail du groupe 'MBSE Focus Group' de l'INCOSE, qui plus tard a été officialisée comme l'initiative MBSE de l'INCOSE. En 2008, cette enquête a été publiée sous les auspices d'une publication technique INCOSE. Le rapport de 2008 a listé six méthodologies MBSE qui peuvent être adaptées pour développer une méthodologie MBSE propre à un organisme particulier. Ces six méthodologies sont :

- INCOSE Object-Oriented Systems Engineering Method (OOSEM),
- IBM Rational Telelogic Harmony-SE,
- IBM Rational Unified Process for Systems Engineering (RUP-SE),
- Vitech Model-Based Systems Engineering (MBSE),
- JPL State Analysis (SA),
- Dori Object-Process Methodology (OPM),

D'autres méthodologies supplémentaires identifiées comme des lacunes depuis l'enquête INCOSE 2008 :

- Weilkiens Systems Modeling Process (SYSMOD),
- Fernandez Process Pipelines in Object-Oriented Architectures (PPOOA),

Ce qui suit est un bref examen de certaines des méthodologies MBSE les plus notables qui ont bénéficié d'une attention dans les divers forums et les publications de l'industrie et sont destinées à servir de candidats à l'adoption et l'adaptation aux pratiques et procédures d'IS d'une organisation. Cette description est basée en grande partie sur le travail de Murray qui a publié un rapport détaillé en 2012 (Murray 2012) sur les méthodologies MBSE, et aussi en partie sur le rapport d'enquête sur ces méthodologies publié en 2008 (Estefan 2008) par Estefan.

INCOSE Object-Oriented Systems Engineering Method (OOSEM)

La méthodologie OOSEM (Object-Oriented Systems Engineering Method) intègre une approche top-down dirigée par les modèles, elle utilise le langage SysML pour la spécification, l’analyse, la conception et la vérification des systèmes. L’OOSEM exploite les concepts orientés-objet avec des méthodes ‘top-down’ traditionnelles d’IS et d’autres techniques de modélisation pour aider à concevoir des systèmes plus flexibles et plus extensibles pouvant accommoder l’évolution des technologies et l’évolution des besoins.

L’OOSEM a été développée au milieu des années 1990 par ‘Software Productivity Consortium’ (maintenant ‘Systems and Software Consortium’) en collaboration avec Lockheed Martin Corporation. La méthodologie a été appliquée en partie au développement d’un système d’information largement distribué à Lockheed Martin, ce système contient du matériel, du logiciel et des bases de données. INCOSE a établi le Groupe de travail OOSEM en 2000 pour aider à faire évoluer la méthodologie OOSEM. L’OOSEM est une approche hybride qui exploite les techniques orientés-objet et les fondations de l’IS. Elle introduit aussi quelques techniques uniques comme indiqué dans la Figure 22.

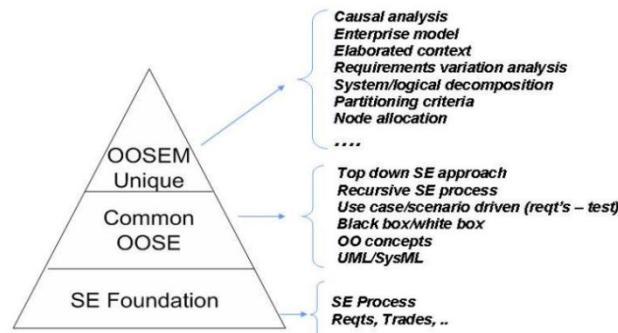


Figure 22 : La fondation de l’OOSEM. (Murray 2012)

La figure ci-dessous montre le processus d’IS supporté par l’OOSEM :

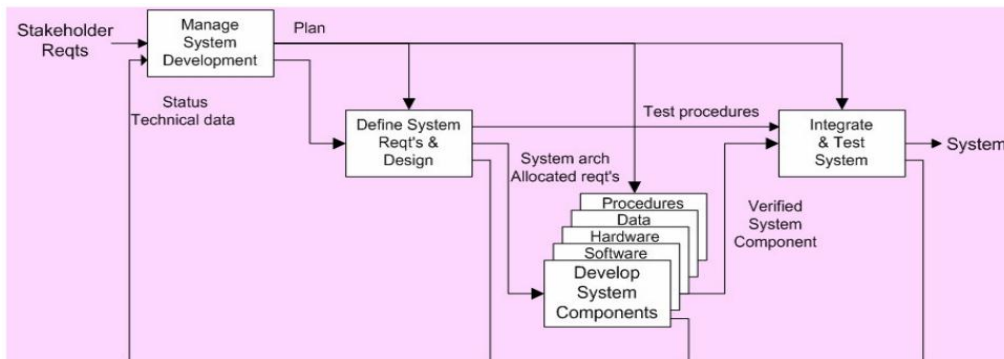


Figure 23 : Les activités de l’OOSEM dans le contexte du processus de développement d’un système. (Murray 2012)

Les principes fondamentaux de l'OOSEM incluent des pratiques reconnues et essentielles pour l'IS. Parmi ces pratiques, on cite :

- Développement intégré des produits, essentiel pour améliorer la communication
- Un processus en cycle en 'V' récursif appliqué sur les différents niveaux de la hiérarchie du système.

Comme indiqué sur la Figure 24, l'OOSEM contient les activités de développement suivantes :

- Analyser les besoins des parties prenantes.
- Définir les exigences du système.
- Définir l'architecture logique ou fonctionnelle du système.
- Synthétiser les architectures candidates.
- Optimiser et évaluer les alternatives.
- Valider et vérifier le système.

Ces activités sont en cohérence avec un processus de développement en cycle en 'V' qui peut être appliqué récursivement et itérativement aux différents niveaux de la hiérarchie du système. Les principes fondamentaux de l'IS comme les processus de gestion (gestion des risques, gestion de configurations, planification, etc.) et l'utilisation des équipes multidisciplinaires doivent être appliqués pour soutenir chacune de ces activités pour qu'elles soient efficaces.

L'OOSEM utilise une approche basée-sur/dirigée-par les modèles pour représenter les différents artefacts générés par les activités du développement en utilisant principalement le langage SysML. Cela, permet aux ingénieurs système de capturer, analyser, et spécifier le système et ses composants avec précision et assurer la cohérence entre les différents points de vue du système.

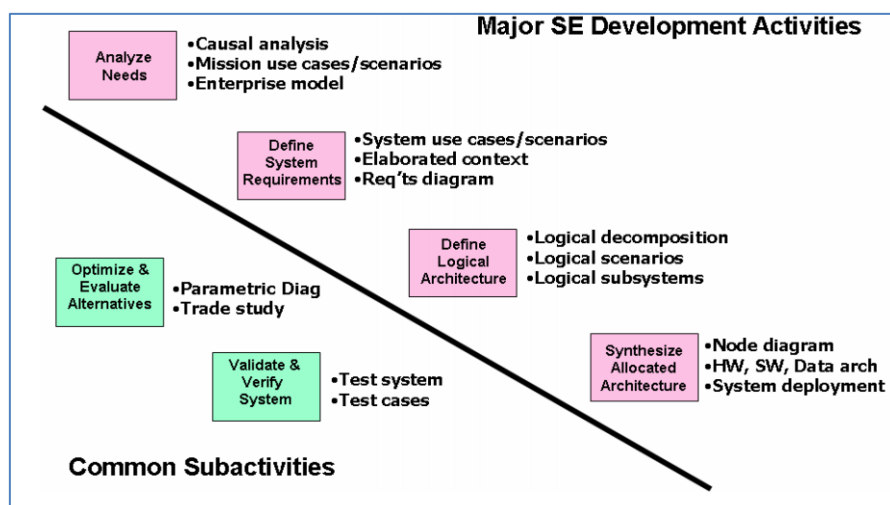


Figure 24 : Les activités de l'OOSEM et les artefacts de modélisation. (Murray 2012)

IBM Rational Telelogic Harmony-SE

Harmony Systems Engineering (Harmony-SE) est une partie d'un processus intégré plus large pour le développement des logiciels et des systèmes physiques connu sous le nom 'Harmony'. Harmony et Harmony-SE ont été développés par l'entreprise ILogix, le fournisseur d'outils de modélisation pour le marché de l'embarqué. La Figure 25 montre graphiquement le processus Harmony de développement des logiciels et des systèmes intégrés.

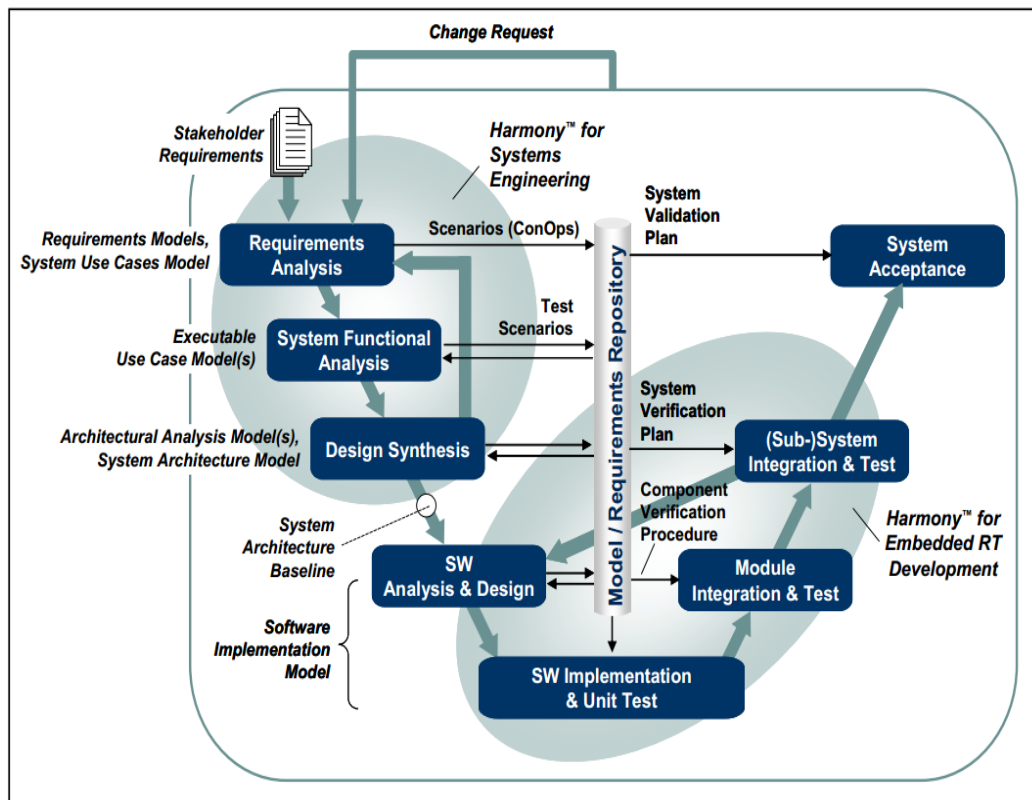


Figure 25 : Le processus Harmony de développement des systèmes et logiciels intégrés. (Hoffmann 2011)

Généralement les éléments du processus Harmony sont supportés par l'environnement de développement Telelogic-Rhapsody, mais ce processus est conçu pour être indépendant de l'outil (il n'y a pas un outil spécifique pour implémenter le processus, plusieurs outils peuvent être utilisés). Le processus Harmony ressemble au processus classique de développement en cycle en 'V'. Dans le processus Harmony, le modèle et les exigences sont sauvegardés dans une base de données centralisée. Les composants d'IS du processus Harmony montrés dans le coin supérieur gauche de la Figure 25 (Harmony-SE) ont les objectifs suivants :

- Identifier/dériver les fonctionnalités requises du système.
- Identifier les états et les modes associés au système.
- Allouer les fonctionnalités/les modes du système à une architecture physique.

Les activités et les artefacts dans le processus Harmony-SE contiennent les activités suivantes :

- L'analyse des exigences,
- L'analyse fonctionnelle du système,
- La conception de l'architecture,

La figure suivante illustre mieux les éléments du processus et les flux de quelques produits principaux du travail (Work items) :

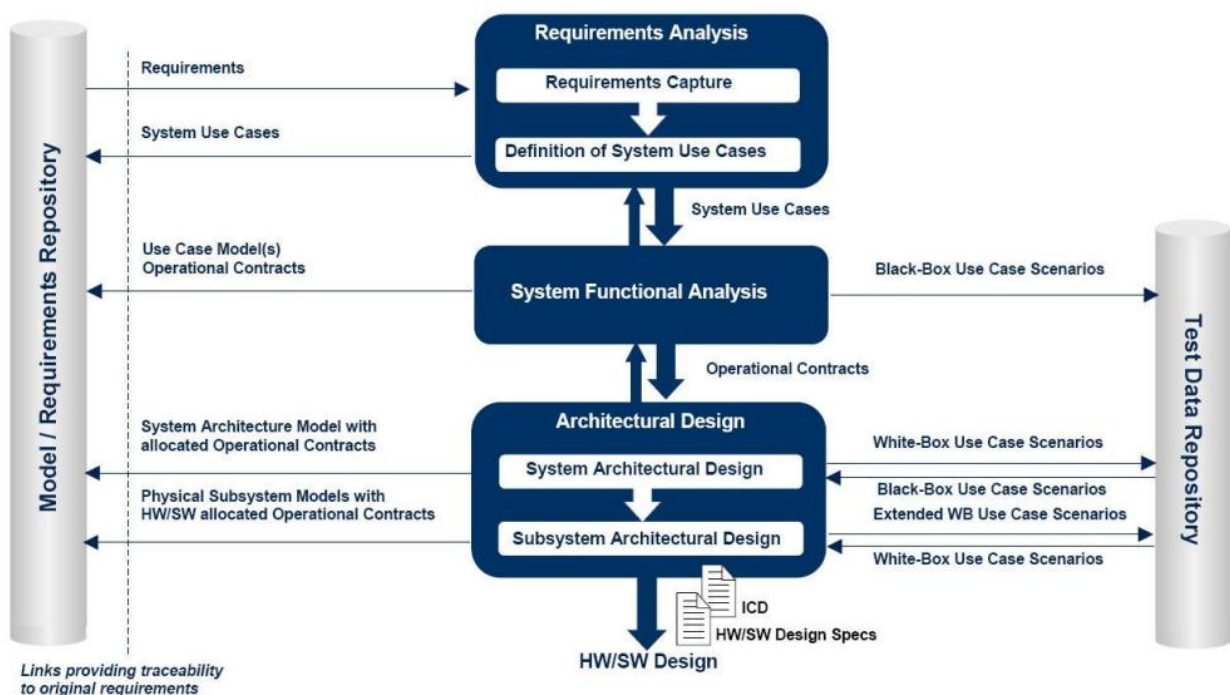


Figure 26 : les éléments du processus Harmony-SE. (Murray 2012)

On note qu'en plus de l'utilisation d'une base de données pour le modèle et les exigences, une deuxième base de données est recommandée pour capturer les scénarios d'utilisation (voir Figure 26). Une description détaillée de chacune des trois activités IS du processus Harmony (analyse des exigences, analyse fonctionnelle du système, la conception de l'architecture) est fournie dans (Hoffmann 2011).

IBM Rational Unified Process for Systems Engineering (RUP-SE)

La méthodologie 'Rational Unified Process for Systems Engineering' RUP-SE est dérivée de la méthodologie 'Rational Unified Process' RUP qu'est une méthodologie d'IBM, elle a été largement utilisée dans le gouvernement et dans l'industrie pour le développement des logiciels. Elle a été créée spécifiquement pour répondre aux besoins des projets en IS. L'objectif de sa création était d'appliquer les bonnes pratiques de la RUP dans le domaine du développement du logiciel pour surmonter les

difficultés de spécification, d'analyse, de conception et de développement des systèmes (physiques). Son but est d'aider les organisations à réduire le temps, le coût, les risques et à améliorer la qualité des systèmes qu'ils développent. La RUP-SE est l'extension de la RUP pour supporter le développement des systèmes par l'approche dirigée par les modèles.

La RUP est basée sur un ensemble de blocs de construction qui décrivent ce qui doit être produit, les compétences nécessaires, et la description, étape par étape, comment les objectifs du développement seront achevés. Les blocs de construction ou les éléments principaux de la méthodologie RUP sont les suivants :

- Les rôles '**le qui**', le rôle définit l'ensemble des compétences et des responsabilités.
- Les produits de travail '**le quoi**', un produit de travail représente le résultat ou le produit d'une activité, y compris tous les documents et les modèles produits le long du processus.
- Les tâches ou les activités '**le Comment**', une tâche ou une activité décrit une unité de travail assignée à un rôle '**le qui**' pour produire un résultat significatif.

Le cycle de vie de la méthodologie RUP est l'implémentation du modèle en spirale pour un développement itératif et incrémental. Il a été créé par l'assemblage des blocs de construction dans des séquences semi-ordonnées. Donc, le cycle de vie de la RUP est une structure de travail découpée, qui peut être personnalisée pour répondre aux besoins spécifiques d'un projet. Le cycle de vie de la RUP organise les tâches en phases et en itérations. La méthodologie RUP-SE est dérivée de la méthodologie RUP, donc elle hérite tous ses principes de base, qui ont été raffinés et adaptés pour améliorer leur utilité pour l'ingénierie des systèmes. La RUP-SE adopte l'approche de la RUP de la conception concurrente et le développement itératif dans l'IS. En plus, elle présente des modèles de flux de travail très flexibles, nécessaires pour identifier la partie physique, la partie logique, et les rôles dans un projet d'IS. Les éléments clés de la méthodologie RUP-SE qui étendent ceux de la RUP à l'ingénierie des systèmes sont les suivants :

- **Nouveaux rôles** : dans la RUP-SE, l'équipe du développement inclut les ingénieurs système en plus des rôles comme les architectes, les développeurs, les testeurs, etc. le rôle d'un ingénieur système est principalement la spécification du système entièrement et le déploiement de celui-ci, et d'aider à établir toutes les exigences du système.
- **Nouveaux éléments de travail et flux de travail** : RUP inclut un support complet pour les préoccupations des systèmes logiciels, comme la facilité d'utilisation, la maintenabilité, les performances, et l'évolutivité. La RUP-SE ajoute des éléments de travail et des flux de travail qui concernent d'autres préoccupations dans le domaine de l'IS, comme la sécurité, la formation et le support logistique.

- **Point de vue IS** : un cadre d'architecture pour la RUP-SE a été développé, il contient des éléments comme les différents niveaux du modèle, les points de vue, et les vues.

Vitech Model-Based Systems Engineering Methodology

La société Vitech offre une méthodologie MBSE qui est indépendante de l'outil (elle peut être implémentée sur plusieurs outils). La méthodologie Vitech est basée sur quatre activités principales de l'IS, ces activités sont concurrentes, liées et maintenues via une base de données commune pour la conception du système (voir Figure 27).

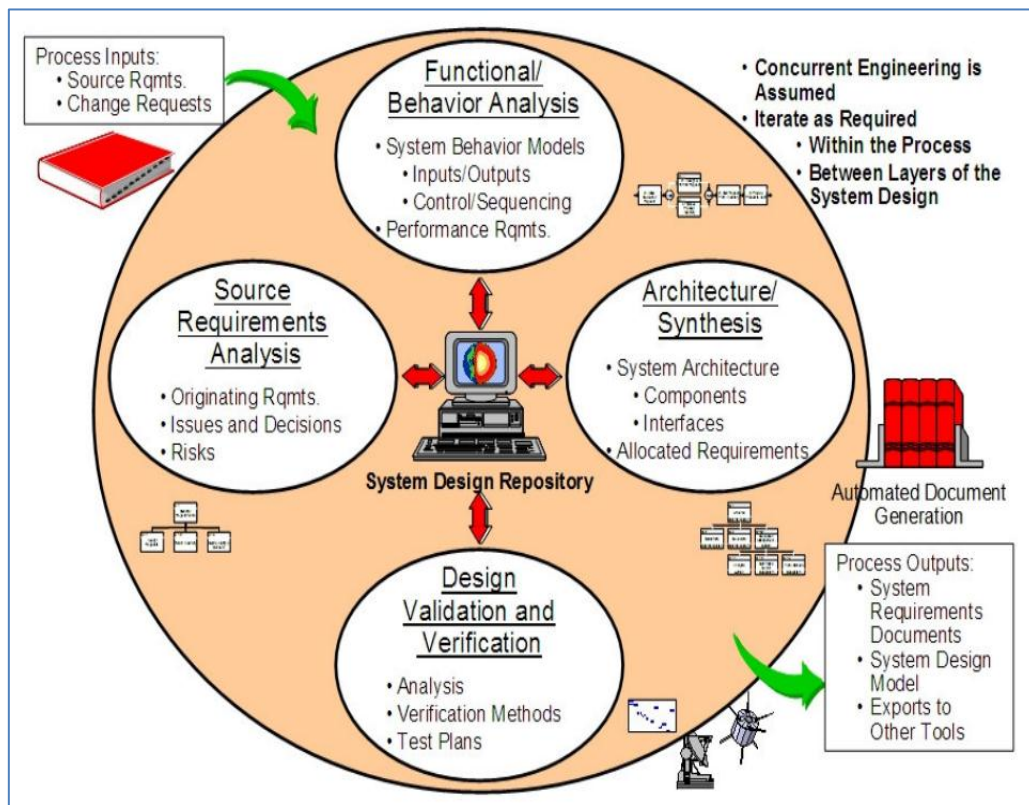


Figure 27 : Les activités principales d'IS de l'approche MBSE Vitech. (Murray 2012)

Dans la méthodologie Vitech, il est souligné qu'un langage de définition ou de description des systèmes SDL (System Definition Language) est nécessaire pour gérer les modèles. Cela signifie qu'un modèle d'information sous forme d'un schéma ou ontologie est nécessaire pour gérer la syntaxe (structure) et les sémantiques (la signification) des éléments des modèles du système. Un langage de modélisation de système a plusieurs avantages tels que fournir un langage structuré, commun, explicite, et indépendant du contexte pour la communication technique servant comme guide pour les analystes des exigences, les concepteurs du système, et les développeurs, et fournit une structure pour les générateurs des vues graphiques, pour les générateurs des rapports, et les vérificateurs de cohérence.

Quelques principes fondamentaux aident à utiliser la méthodologie Vitech :

- Modéliser via un langage de modélisation le problème et la solution, inclure des graphiques sémantiquement significatifs pour rester explicite et cohérent. Cela aide à faciliter la traçabilité du modèle, la cohérence des graphiques, la génération automatique de la documentation et des artefacts, la validation et la simulation dynamiques, et favorise une communication plus précise.
- Utiliser une base de données commune pour la conception du système selon l'approche MBSE.
- Développer le système horizontalement puis verticalement, c.-à-d. faire le travail en complet, niveau par niveau. La méthodologie Vitech utilise un processus incrémental d'IS qui permet d'avoir des solutions intermédiaires à des niveaux de détails augmentant pendant le processus de spécification du système (voir Figure 28).
- Utiliser des outils pour la réalisation des modèles.

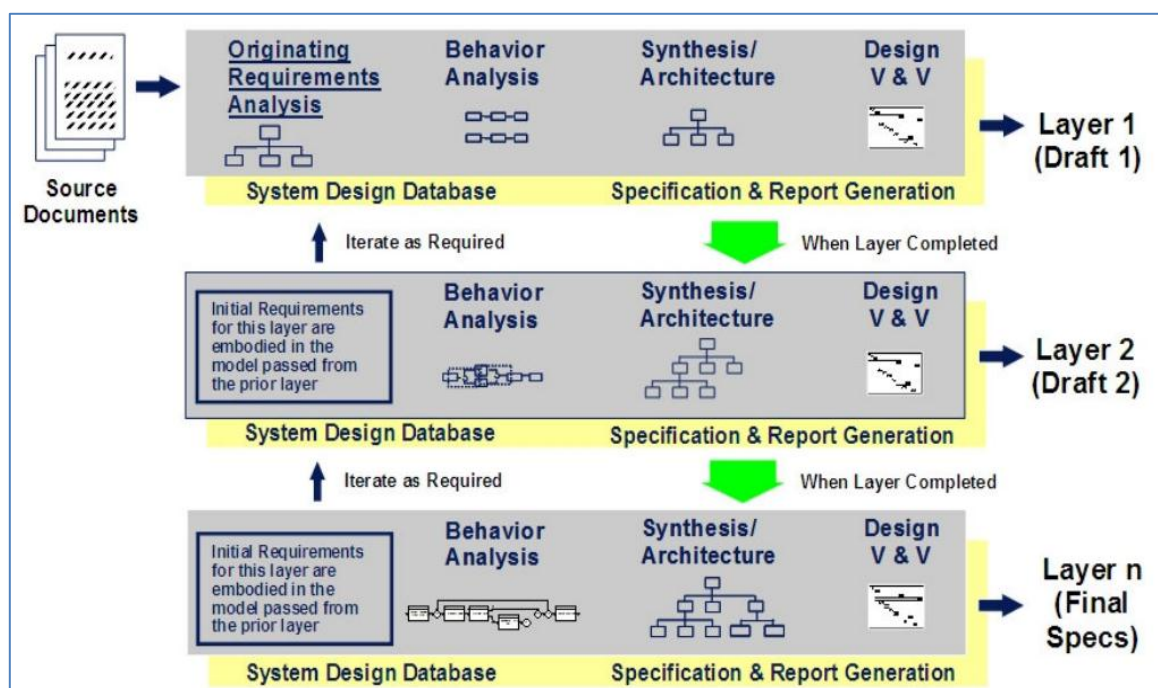


Figure 28 : la méthodologie Vitech, illustration développement incrémentale. (Murray 2012)

Le développement incrémental adopté par l'approche Vitech répète à chaque niveau les activités principales concurrentes de l'IS. À chaque fois qu'un niveau de la conception du système est terminé avec succès, on passe au niveau suivant. Quand on atteint le niveau de détails désiré, la conception est terminée. L'avantage principal de cette approche de développement, par rapport aux approches traditionnelles d'IS qui utilisent le modèle en cascade (voir chapitre 2, section 7.1), est qu'il présente une approche de conception à risque faible puisque des solutions complètes à des niveaux croissants de détail sont disponibles à chaque niveau. La complétude et la convergence sont des

principes essentiels de l'approche de développement adoptée par la méthodologie Vitech. L'équipe d'IS doit compléter un niveau avant de passer au suivant (la complétude). L'équipe ne peut pas itérer en retour plus d'un niveau (la convergence).

Weilkiens Systems Modeling Process SYSMOD

Le processus SYSMOD est introduit dans le livre 'Systems Engineering SysML/UML' (Weilkiens 2007) par Weilkiens membre de l'OMG (Object Management Group). Le processus de modélisation des systèmes SYSMOD (SYStems MODEling process) est une approche pour modéliser les exigences et l'architecture physique et logique d'un système. Elle fournit un ensemble de tâches avec des entrées et des sorties, des directives et des bonnes pratiques. SYSMOD utilise le langage SysML pour la modélisation. SYSMOD inclut l'identification des parties prenantes, la définition des exigences, la définition du contexte du système, l'analyse des exigences, par exemple, la définition du domaine de modélisation, et la définition de l'architecture, fonctionnelle et physique. Voici quelques caractéristiques de la méthodologie SYSMOD :

- une approche orientée utilisateur pour l'ingénierie des exigences et les architectures du système,
- permet différents niveaux de détails et de modélisation,
- des lignes directrices et des exemples fournis pour chaque activité du processus,
- fournir des activités additionnelles, par exemple, pour les architectures fonctionnelles ou des variantes de modélisation.
- une méthodologie indépendante de l'outil (peut-être implémenté sur plusieurs outils logiciels).

Fernandez ISE & Process Pipelines in OO Architectures (ISE&PPOOA)

ISE&PPOOA (Integrated Systems Engineering and PPOOPA) présente un processus intégré, des méthodes et un outil pour l'ingénierie des systèmes mécatroniques. Une partie du processus contient les premières étapes du processus de l'IS applicable sur n'importe quel type de système et non seulement les systèmes logiciels. Le sous-processus d'IS intègre les bonnes pratiques d'IS traditionnelle et l'IS dirigée par les modèles. On cite quelques caractéristiques de la méthodologie ISE&PPOOA :

- ISE&PPOOA fournit un ensemble des directives pour aider les ingénieurs dans l'architecture du système.
- L'un des livrables du projet est l'architecture fonctionnelle du système représentant la hiérarchie fonctionnelle en utilisant le diagramme SysML de définition des blocs. Ce diagramme est complété par des diagrammes d'activités pour les flux fonctionnels généraux du système.

- Un autre livrable est l'architecture physique du système, elle représente la décomposition du système en sous-systèmes et composants en utilisant le diagramme SysML de définition de blocs. Ce diagramme est complété par les diagrammes SysML de définition interne du bloc pour chaque sous-système. Les diagrammes d'activités et d'états sont nécessaires aussi. Une description textuelle des blocs du système est aussi fournie.
- L'architecture du sous-système logiciel est décrite dans PPOOA en utilisant deux vues supportées par un ou plusieurs diagrammes en utilisant la notation UML. Une vue statique de la structure du système et une vue dynamique pour le comportement. Le diagramme d'architecture du système représente les éléments du système et les liens entre eux. La vue du comportement du système est supportée par les diagrammes d'activités UML/SysML représentant une vue interne des flux des actions exécutées par le système en réponse aux évènements.
- L'allocation fonctionnelle bidirectionnelle entre les composants et les réponses du système modélisées sous forme de diagrammes d'activités est considérée et supportée par un outil.

Dori Object-Process Methodology (OPM)

La méthodologie OPM (Object-Process Methodology) est conçue et développée par le professeur Dov Dori et détaillée dans son livre 'Object-Process Methodology' (Dori 2002). C'est une approche pour concevoir les systèmes d'information. Les systèmes sont représentés en utilisant les modèles d'objet et les modèles de processus.

L'OPM utilise un langage de modélisation qui combine un ensemble minimal de blocs de construction. Avec une représentation graphique et textuelle dans un seul type de diagramme. L'OPM utilise deux types d'éléments : les entités et les liens. Les entités sont utilisées pour représenter les éléments physiques ou informationnels qui peuvent être dans le système ou à l'extérieur du système à concevoir. Les entités utilisées dans l'OPM sont des objets, des états et des processus. Un objet est une entité qui existe pour une période de temps, un état est une situation que l'objet peut prendre, un processus est un phénomène qui transforme un ou plusieurs objets en les créant, modifiant leurs états ou en les détruisant. Les liens utilisés dans l'OPM sont des liens structurels et des liens de procédures. Un lien structurel forme une relation entre deux objets et il est persistant. Un lien de procédure est utilisé pour relier un processus à un objet pour indiquer un changement dans l'état de l'objet ou à un autre processus pour indiquer le lancement d'un deuxième processus quand le premier se termine.

Les objets, les processus et les liens entre eux sont représentés en utilisant le diagramme OPD (Object Process Diagram). Possédant un seul diagramme, cette méthodologie ne connaît pas de multiplicité pour représenter les objets et les processus, ce qui améliore l'interchangeabilité entre les différents diagrammes et même entre les différentes méthodes. Une autre caractéristique de l'OPM est

le langage utilisé pour construire les diagrammes (OPD), ce langage peut être traduit en phrases pour décrire textuellement l'histoire du diagramme. La puissance du langage OPD est qu'il est lisible par les humains et aussi interprété par les calculateurs, ce qui le rend un langage interchangeable.

JPL State Analysis (SA)

La méthodologie SA (State Analysis) est une méthodologie développée par 'NASA Jet Propulsion Laboratory (JPL)', c'est une méthodologie destinée au développement des systèmes de contrôle en se basant sur les modèles et les états du système à contrôler. Selon cette méthodologie :

- l'état : une représentation momentanée d'un système qui évolue.
- modèle : décrire l'évolution des états.
- les variables d'états : une représentation abstraite de l'état du système.

Ensemble, les états et les modèles fournissent ce qui est nécessaire pour prédire le fonctionnement du système, le contrôler vers un état désiré, et évaluer ses performances. La méthodologie SA définit un processus itératif pour découvrir les états du système et les modéliser en se basant sur la méthode 'Mission Data System MDS' (Dvorak et al. 2000) pour la conception des systèmes de contrôle. Elle définit un processus pour identifier et modéliser les états d'un système physique et leurs relations avec d'autres états, les objectifs de contrôle de ces états, la planification, la coordination, l'exécution, et les mécanismes de réponse par lesquels ces objectifs peuvent être atteints de manière fiable.

La méthodologie SA est une approche qui se base sur les modèles, elle commence par un modèle du comportement du système à contrôler et ensuite elle procède avec les modèles du système de contrôle nécessaire. Les deux modèles sont ensuite itérativement et incrémentalement raffinés le long du développement dans un jeu qui permet de maintenir la synergie entre eux. Les variables d'états, les mesures, les commandes, et les relations entre eux sont résumées dans des diagrammes 'State Effects Diagrams' qui fournissent des liens vers des modèles formels détaillés du système à contrôler sous différentes formes. Le système de contrôle est modélisé en utilisant des représentations mathématiques appropriées telles que les équations mathématiques différentielles, ou des abstractions de modélisation discrète telle que l'UML, diagrammes d'états, et d'autres moyens.

L'un des principes de la méthodologie SA est la séparation entre le système de contrôle et le système à contrôler. Le besoin d'être explicite concernant la ligne qui sépare ces deux systèmes est causé par la dépendance entre la conception du système de contrôle et les modèles du système à contrôler. Cette séparation n'est pas soigneusement définie comme on l'imagine dans la plupart des systèmes et spécifiquement dans le cas des systèmes complexes. Parfois cette ligne de séparation est la ligne qui sépare la partie physique de la partie logique (logiciel), mais souvent, ce n'est pas le cas.

Donc l'ambiguïté de cette ligne devient une source d'ambiguïté pour la modélisation, cela tend à éroder l'intégrité de la conception du système de contrôle. La méthodologie SA impose ou applique cette délimitation pour rendre l'application des modèles dans la conception des systèmes de contrôle beaucoup plus simple et plus fiable.

9. Conclusion

L'ingénierie système est une discipline qui coordonne les différentes disciplines et les moyens nécessaires pour développer des systèmes réussis en respectant les coûts et les délais. Généralement l'IS est destinée pour les systèmes complexes pour faciliter leur développement, son application sur les systèmes simples n'aura pas grand intérêt. Elle est principalement basée ces des normes qui regroupent les bonnes pratiques des différents industriels, ces pratiques sont acquises après de longues expériences dans le développement et la gestion des systèmes complexes. Ces industriels se sont regroupés en plusieurs organismes, comme l'INCOSE et l'AFIS par exemple, pour travailler ensemble et faire progresser l'IS.

Chaque système à un cycle de vie, afin de faciliter le développement et la gestion des systèmes complexes, ce cycle de vie est décomposé en plusieurs phases. Cette décomposition n'est pas standard, chaque organisme adopte une décomposition qui lui facilite le travail et qui répond le mieux à ses besoins. Pendant chaque phase du cycle de vie d'un système, un ou plusieurs processus d'IS sont exécutés. Chaque processus regroupe un ensemble d'activités qui devront être exécutées dans un ordre bien déterminé par des personnes qualifiées et avec les moyens nécessaires. En IS, il y a quatre groupes de processus, les processus techniques, les processus de management, les processus contractuels et les processus d'entreprise. Du fait de l'invariance des types d'activité à réaliser, ces processus sont considérés aujourd'hui comme des fondements génériques pour toute mise en œuvre de l'IS.

L'IS est principalement basée sur un ensemble de processus, ces processus sont regroupés en trois principales normes. Ces normes définissent les types d'activités à réaliser et les résultats à produire pendant les différentes phases du cycle de vie d'un système. Elles recouvrent des champs différents, de manière d'autant plus approfondie que leur champ est plus limité et ainsi se complètent. Ces trois normes sont :

- IEEE-1220: Standard for application and Management of the Systems Engineering Process.
- EIA/ANSI-632: Processes for Engineering a System.
- ISO/IEC/IEEE-15288: Systems and Software Engineering-System Life Cycle Processes.

Les processus décrits dans les standards doivent être complétés d'une part par les politiques et procédures de l'entreprise et d'autre part par des outils qui vont permettre de réaliser les projets.

Du fait que la norme ISO/IEC/IEEE-15288 est la plus mature, elle sera notre point de départ pour mettre en place un environnement d'IS pour le développement de nos futurs systèmes. Cela n'écarte pas définitivement l'utilisation des normes IEEE-1220 et EIA/ANSI-632 en cas de nécessité. Par exemple, pour la phase de définition d'un système, la norme IEEE-1220 est plus détaillée que la norme ISO/IEC/IEEE-15288.

Un système est développé selon un processus, les activités de ce processus sont exécutées dans un ordre bien déterminé pour atteindre un objectif précis. Pour représenter l'enchaînement de ces activités et les flux échangés entre elles, un modèle est nécessaire, ce modèle guidera les différents membres du projet dans leur démarche de travail. Plusieurs approches ou modèles de développement ont été présentés dans la littérature, ces modèles sont généralement regroupés en trois grandes familles. Le modèle en cascade, le modèle en spirale, et le modèle en V. Des variantes des modèles en cascades et en spirale ont été largement utilisées pour supporter des développements structurés, itératifs et incrémentaux dans les projets de développements logiciels, tandis que le modèle en V et des variantes de ce modèle ont été largement appliqués dans le domaine de l'ingénierie des systèmes.

Le modèle en 'V' est le mieux adapté pour le développement des systèmes physiques, mais il présente des limitations. Par exemple, il ne peut pas être utilisé pour le développement des systèmes complexes, il ne supporte pas le changement des exigences, ces exigences doivent être fixes et clairement définies dès le début, etc. pour remédier à tous ces problèmes, des variantes du modèle en 'V' ont été présentées, par exemple le modèle en double 'V' qui est destiné à faciliter le développement des systèmes complexes. D'autres utilisent le simple modèle en 'V' avec suffisamment de temps passé sur la définition et la conception du système (branche descendante du cycle en 'V') pour gérer les variations des exigences et minimiser les risques.

Dans notre cas, nous allons partir d'un modèle en V pour développer notre système dans sa globalité. Nous proposons, pour développer les composants et les sous-systèmes du système principal d'utiliser aussi le modèle en 'V'. En effet, l'objet de nos travaux concerne les systèmes de production qui sont des systèmes physiques relativement complexes. Pour gérer la variation des exigences et minimiser les risques associés aux ambiguïtés (définition non claire des exigences, des exigences non fixes, etc.), on doit passer suffisamment de temps sur la définition du système avant de passer à sa réalisation, vérification et validation. Les processus de base du modèle en 'V' nous paraissent suffisamment rigoureux pour garantir le succès de la conception d'un système reconfigurable. Nous montrerons par contre que ce modèle en 'V' ne permet pas de montrer la reconfiguration du système,

qui peut intervenir à tout moment au cours de la phase d'utilisation du système. Nous proposerons donc une évolution de ce modèle en V pour les systèmes reconfigurables.

Les approches dirigées par les modèles ou communément connues sous le nom MBSE sont la nouvelle tendance ou philosophie de l'IS. Le but est de faciliter la démarche d'IS en utilisant les modèles comme le moyen principal pour : capter et analyser les exigences, spécifier et concevoir les systèmes, tester et valider et presque dans toutes les activités de l'IS le long du cycle de vie d'un système. Les modèles facilitent aussi la communication entre les différentes disciplines impliquées dans le projet de développement du système. Selon Pearce et Mause (Pearce & Hause 2012), les approches de développement dirigées par les modèles sont actuellement considérées comme étant les meilleures pratiques pour la spécification, la conception, l'analyse et la vérification des systèmes complexes.

Plusieurs organismes ont développé leurs propres approches MBSE. Ces méthodes peuvent être adaptées par d'autres organismes pour développer leur propre approche d'IS basée sur les modèles ou elles peuvent être utilisées comme sources d'inspiration pour développer des méthodes MBSE propres à chaque organisme.

Donc, du fait de la réussite récente des approches MBSE dans la conception et le développement des systèmes complexes, la méthodologie qu'on va mettre en place pour développer nos systèmes (systèmes logistiques internes) sera du même type c.-à-d. une méthodologie basée sur les modèles. Ces modèles sont construits en utilisant des langages de modélisation graphiques, textuels, ou hybrides. Pour le moment, il y a deux langages qui ont prouvé leur efficacité à supporter les méthodologies MBSE. Ces deux langages sont : l'UML utilisé uniquement pour le développement de la partie logiciel d'un système, le deuxième langage c'est le SysML qui est dérivé du premier langage et utilisé principalement pour développer la partie physique d'un système.

On a mentionné auparavant (dans le chapitre 1, section 3.2) que la modularité est une caractéristique très importante des systèmes reconfigurables. Le chapitre suivant traite en détail cette caractéristique et propose une méthode pour assurer une structure modulaire lors de la conception des systèmes de production.

Chapitre 03 : Modularité dans la conception des Systèmes Manufacturiers Reconfigurables

1. Introduction

Comme tout système de production, un RMS est considéré comme un système complexe. Il est conçu pour s'adapter à son environnement. Pendant sa durée de vie, le système prendra différentes configurations. Il va s'adapter et évoluer, ce qui rend sa conception plus difficile qu'un système de production classique (une seule configuration). Dans notre travail, notre objectif est de développer une méthodologie ou une approche globale et simple pour faciliter la conception des RMSs. Notre démarche est d'examiner les différentes méthodes et techniques utilisées dans le développement des systèmes complexes et d'essayer de les adapter à la conception des RMSs. La modularité est l'une de ces techniques. Selon [Baldwin & Clark \(2006\)](#), la modularité d'un point de vue technique permet de gérer la complexité. L'idée de base de la conception modulaire est d'organiser un système complexe en l'occurrence un RMS comme un ensemble de composants distincts qui peuvent être développés indépendamment et ensuite les intégrer ensemble. Selon [Koren & Shpitalni \(2010\)](#), la modularité est une caractéristique clé d'un système reconfigurable et elle doit être intégrée dans le système, dans son intégrité, dès les premières phases du processus de conception afin d'assurer un degré élevé de reconfigurabilité.

Les lignes de production sont généralement conçues pour un seul produit et pour une taille de lot particulière de ce produit. Aujourd'hui, les produits sont de plus en plus personnalisés, et les formats de lots sont de plus en plus variables. Par conséquent, les lignes de production doivent être ajustées de plus en plus souvent, un processus qui coûte du temps, augmente le risque d'erreurs et entraîne des temps d'arrêt. Pour résoudre ces problématiques, les entreprises envisagent d'utiliser des modules 'brancher et produire' ou 'plug and product' comme solution. Afin d'ajuster la production, les travailleurs ajouteront simplement des modules existants ou supprimeront ceux qui ne sont pas nécessaires. Les modules indépendants doivent assumer des tâches spécifiques dans une chaîne de production comme le pressage, le forage ou l'assemblage par exemple. Chaque module contient les outils nécessaires pour réaliser sa tâche, plus la capacité de se configurer et de coordonner tous les processus de fabrication avec les autres modules. Ces modules sont mis en place pour créer un système de production personnalisé et polyvalent.

2. La modularité

2.1. Définition

D'un point de vue conception, la modularité est une approche de conception qui subdivise un système en parties plus petites appelées modules qui peuvent être créés indépendamment et ensuite utilisés dans différents systèmes. Ainsi, la conception modulaire est une conception dont la fonctionnalité est divisée en unités discrètes, cohérentes et autonomes avec des interfaces bien définies qui permettent de remplacer ces unités par des composants similaires ou des produits provenant de sources alternatives avec un impact minimal sur les unités existantes.

D'un point de vue utilisation, [Höltkä-Otto & de Weck \(2007\)](#) mentionnent que la modularité est l'utilisation des modules standards pour former plusieurs produits ce qui permet une grande variété. En outre, [Chun-Che Huang & Kusiak \(1998\)](#) donnent la définition suivante : la modularité est l'utilisation des unités standards pour créer des variantes de produits ou de systèmes. Elle vise à identifier des unités indépendantes, normalisées et interchangeables pour satisfaire une variété de fonctions ou de besoins.

D'autres définitions de la modularité pourraient être trouvées dans la littérature et on peut remarquer que la modularité a été principalement appliquée à la conception des produits modulaires, mais un système de production pourrait être considéré comme un produit complexe et tous les travaux réalisés sur le développement des produits modulaires pourraient être adaptés et appliqués au développement des systèmes manufacturiers modulaires.

Définition de la notion de module

Selon [Baldwin & Clark \(2006\)](#), les modules sont des parties distinctes d'un système plus large. Ils sont conçus et réalisés indépendamment les uns des autres, mais doivent fonctionner correctement dans leur ensemble. [Höltkä-Otto & de Weck \(2007\)](#) définissent le module comme une partie indépendante du système qui est fortement couplé en interne et faiblement couplé avec le reste du système.

2.2. Les types de modularité

En fonction du cycle de vie d'un système, on peut en déduire qu'il existe trois types de modularité ([Baldwin & Clark 2006](#)) : la modularité dans la conception, la modularité dans la fabrication et la modularité dans l'utilisation. Dans notre travail, on s'intéresse particulièrement à la modularité dans la conception et à la modularité dans l'utilisation. Dans notre recherche, nous essayons de répondre aux questions suivantes : comment rendre facile la conception d'un RMS, et comment rendre facile sa gestion (utilisation et reconfiguration ou changement) ?

- Modularité de conception : Un système est modulaire dans sa conception si le processus de sa conception peut être divisé et réparti sur des modules séparés.
- Modularité d'utilisation : Un système est modulaire dans son utilisation si l'utilisateur peut combiner des éléments ou des modules pour arriver à une configuration finale souhaitée. Par exemple, dans le cas d'un système de production, nous pouvons choisir différentes machines, différents outils de manutention, outils de stockage, etc. Ces modules doivent s'adapter complètement parce qu'ils ont été conçus avec des interfaces standards qui sont communes entre ces modules.

2.3. Avantages de la modularité

La modularité présente de nombreux avantages que nous avons jugés importants pour la conception des RMS. Ces avantages sont regroupés en trois familles selon leur perspective d'impact : perspective technique, perspective organisationnelle et perspective économique.

2.3.1. Perspective technique

D'un point de vue technique, la modularité aide à :

- Gérer la complexité du système : la modularité rend la nature complexe d'un système gérable en fournissant une division efficace des différentes tâches de conception du système (Baldwin & Clark 2006). Les modules du système pourraient être conçus, produits, testés, vérifiés et validés indépendamment les uns des autres. Et ces modules peuvent être modifiés, améliorés et évolués sans impact sur le reste du système. En outre, la modularité réduit la complexité de l'architecture du système, de sorte qu'elle soit plus simple et donc plus facile à gérer (Höltkä-Otto & de Weck 2007).
- Améliorer la reconfigurabilité du système : les systèmes modulaires permettent une flexibilité opérationnelle. Les modules du système peuvent être rapidement et facilement configurés pour répondre aux besoins opérationnels changeants.
- Réduire les risques de modifications : la modularité aide à découpler et à réduire les risques de développement du système (Chun-Che Huang & Kusiak 1998). La séparation des modules diminue la probabilité que la modification d'un module ait une incidence négative sur l'autre.
- Améliorer la maintenabilité du système : les systèmes modulaires sont plus faciles à diagnostiquer, à entretenir et à réparer. Voir (Chun-Che Huang & Kusiak 1998), (Gershenson & Prasad 1997), (Abdullah et al. 2006) pour plus de détails.
- Faciliter la modification du système (la reconfiguration) : le système modulaire peut être modifié et mis à jour facilement. Les différentes configurations du système peuvent être réalisées en remplaçant un ou plusieurs modules dans le système sans modifier le système dans sa globalité. (Chun-Che Huang & Kusiak 1998), (Höltkä-Otto & de Weck 2007).

2.3.2. Perspective économique

D'un point de vue économique, les systèmes modulaires sont moins coûteux à développer et à modifier : grâce à la modularité, les coûts élevés de développement ou de modification d'un système complexe sont réduits aux coûts de développement ou de modification de modules individuels. (Hölttä-Otto & de Weck 2007)

2.3.3. Perspective organisationnelle

Le travail en parallèle est plus facile sur les systèmes modulaires. Principalement les tâches de conception pourraient se poursuivre simultanément sur les modules du système. Les interactions entre les personnes et les équipes travaillant sur ces tâches sont simplifiées. Par conséquent, le temps nécessaire pour terminer la conception du système en entier ou une partie de celui-ci est considérablement réduit. (Baldwin & Clark 2006) (Sosa et al. 2003)

2.4. Les inconvénients de la modularité

Malgré ces nombreux avantages, la modularité présente quelques inconvénients :

- La perte des performances : le point le plus faible de la modularité est la perte des performances. Hölttä-Otto & de Weck (2007) ont discuté en détail ce point. Ils mentionnent que la modularité n'est pas conseillée quand l'objectif est d'atteindre de bonnes performances techniques du système. Selon eux, les architectures modulaires ont une tendance à avoir plusieurs modules et à favoriser les performances commerciales. De l'autre côté, les architectures intégrales ou non-modulaires ont une tendance à être couplées et à favoriser les performances techniques.
- La difficulté de changer les interfaces : quand on fait face à une architecture modulaire, les interfaces sont très critiques parce qu'elles représentent les liens qui maintiennent les modules du système. Le changement d'une interface est problématique parce que tous les modules partageant cette interface doivent, aussi, être modifiés.

3. Principes de conception modulaire

Pour réaliser une architecture modulaire, certains principes et règles doivent être respectés dès les premières phases du processus de développement. Les plus importants sont cités ci-dessous :

- Allocation des fonctions : Au cours du processus de développement, une fois les fonctions du système sont définies, elles doivent être affectées ou allouées aux modules du système. Une règle importante doit être respectée lors de la répartition ou au cours du processus de l'établissement de la correspondance (mappage) entre les fonctions du système et ses modules. Cette règle est la suivante : les fonctions ne doivent pas être partagées entre les modules du système.

- Séparation des modules : Pour définir clairement les modules et les séparer, on peut se poser les questions suivantes : l'architecture ou la conception identifie-t-elle clairement les modules ? Chaque module a-t-il un but clairement défini ? Les modules sont-ils subdivisés autant que possible ? Les différents modules sont-ils vérifiés et ils ne reproduisent pas les mêmes fonctionnalités ?
- Simplification des interfaces : les interfaces doivent être les plus simples possible, ceci afin d'une part, de réduire le nombre d'interactions qui doivent être prises en compte lors du contrôle d'un système pour exécuter ses fonctions, et d'autre part de faciliter la réutilisation des modules dans des circonstances différentes.

4. Exemples de systèmes manufacturiers modulaires

4.1. Exemple 01 : machine modulaire

Généralement, dans un système de production, les machines sont considérées comme les modules les plus pertinents ou les modules de base, le reste des modules du système, comme les modules de manutention ou les modules de stockage, sont considérés comme des modules de support. Dans les lignes de production dédiées ou DMLs (Dedicated Manufacturing Lines), les machines sont dédiées pour exécuter une tâche spécifique simple à une cadence élevée, elles sont simples et, généralement, conçues comme un ensemble solide (pas de modules). Les systèmes de production flexibles ou FMSs (Flexibles Manufacturing Systems) reposent principalement sur les machines CNC qui sont très compliquées, elles sont conçues pour exécuter un large éventail de tâches à un rythme très lent. Comme les machines dédiées, les machines CNC sont conçues comme un ensemble solide et les mêmes modules sont utilisés pour exécuter les différentes tâches. C'est au niveau du système de contrôle que les différentes tâches sont sélectionnées et les paramètres sont choisis. Les machines CNC sont caractérisées par une configuration physique fixe (pas de modules à échanger), mais elles sont au moins plus souples dans leur programme de contrôle, mais cela ne signifie pas que leur système de contrôle (matériel et logiciel) est modulaire.

Les deux types de machines mentionnés ci-dessus (machines dédiées et machines flexibles) ne satisfont pas les exigences de la reconfigurabilité en raison de leur structure fixe qui n'est pas modulaire ; cela représente une barrière importante face à la réalisation d'un système manufacturier reconfigurable RMS. Pour surpasser cette barrière, les académiques et les industriels commencent à développer de nouvelles techniques et méthodes pour concevoir des machines reconfigurables. Dans ce nouveau concept de machines, la modularité est une caractéristique essentielle qui doit être considérée dès le début du processus de développement.

Par exemple, Padayachee & Bright (2012) présentent le développement des machines reconfigurables modulaires (Modular Reconfigurable Machines MRMs) qui sont un nouveau type de machines reconfigurable qui exploite la modularité mécanique de son architecture afin de faciliter le processus de reconfigurabilité et d'atteindre un haut niveau de performances cinématiques. Les machines à outils reconfigurables (Reconfigurable Machines Tools RMTs) sont une famille particulière de machines reconfigurables qui utilisent le principe de la modularité dans leur conception. Dans (Dhupia et al. 2007), un prototype d'une machine à outils reconfigurable nommée machine-arc, cette machine a été développée à partir d'une bibliothèque de modules précompilés.

4.2. Exemple 02 : système logistique modulaire

Le système DLWL-800A (Figure 29) est un système logistique modulaire développé par la société dolang (Dolang 2017). Le système repose sur une conception modulaire où chaque module représente une station qui peut être utilisée indépendamment du reste du système. Les modules du système sont assemblés grâce à des interfaces mécaniques standards et un bus de terrain (PROFIBUS-DP) pour assurer l'échange des données. Le système est composé des modules suivants : le module du manipulateur, le module d'inspection visuelle, le module d'identification automatique, le module d'empilage, le module RFID (Radio Frequency IDentification), le module ASAR (Automatic Storage and Retrieval System), le module de contrôle du flux des produits et le module du conteneur.

Le système peut effectuer : l'assemblage et le transfert des produits, l'identification automatique, l'empilage et l'ASRS (Automatic Storage and Retrieval System). La principale caractéristique du système est que chaque station peut fonctionner indépendamment du reste du système. L'utilisateur peut choisir les stations nécessaires et les combiner librement. Dix combinaisons/configurations du système sont possibles. Le système inclut des technologies mécanique, électrique, optique et pneumatique qui impliquent des connaissances multidisciplinaires.

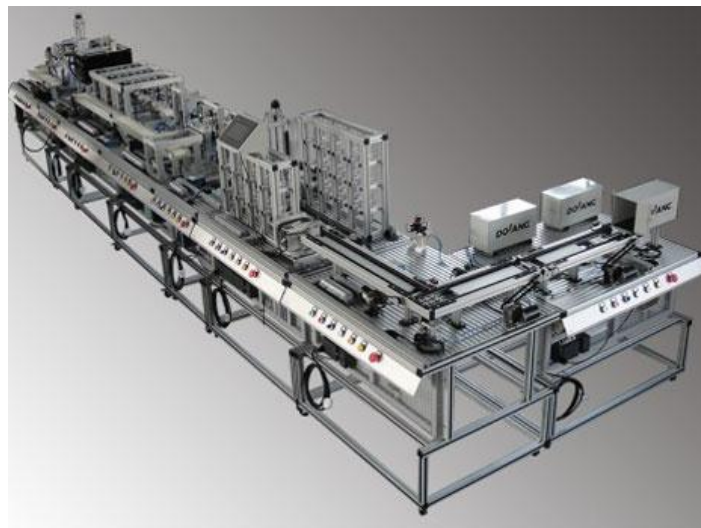


Figure 29. Le système logistique modulaire (éducatif) DLWL-800A. (Dolang 2017)

4.3. Exemple 03 : système de production modulaire

CP Factory met à disposition un système modulaire d'usine intelligente pour l'apprentissage et la recherche. Le système d'apprentissage comprend différents domaines de production : non seulement la ligne d'assemblage, mais aussi le Lean production, la logistique et l'assurance qualité. L'usine CP-Factory est caractérisée par : sa modularité, sa mobilité, ses temps d'installation courts, ses modules Plug & Produce (brancher et produire) et ses interfaces standards. Des stations modulaires, flexibles, combinables et équipées de la technologie industrielle la plus moderne, constituent le cœur du système d'apprentissage. Grâce à des interfaces normalisées, le remplacement des modules s'effectue en quelques minutes. Cela permet une conversion rapide pour changer la configuration du système. Les stations standardisées peuvent être agencées de différentes manières. Un laboratoire constitué de postes de travail individuels peut ainsi se transformer en un de temps raisonnable en une ou plusieurs lignes de production.

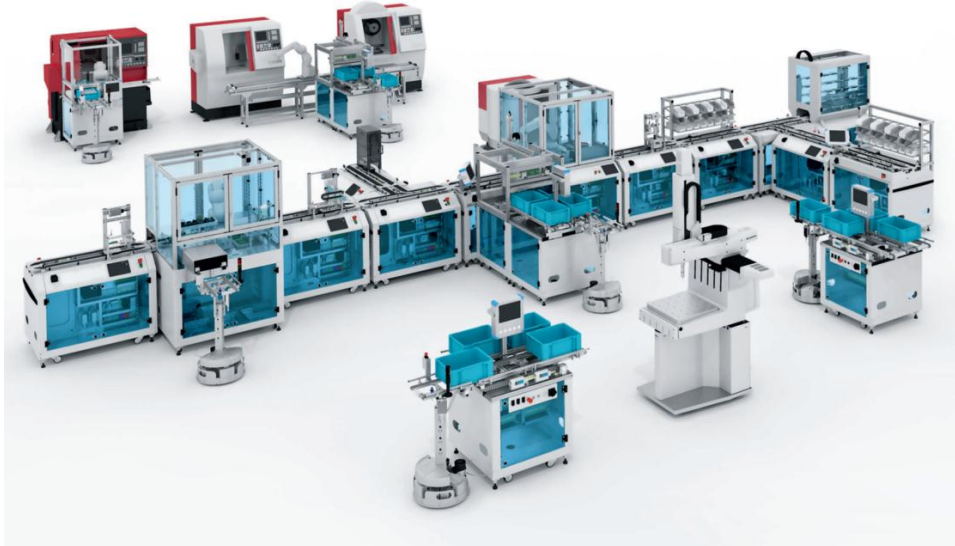


Figure 30 : CP Factory, l'usine Cyber-physique. (festo-didactic 2017)

5. Techniques et méthodes de conception d'un système modulaire

En raison des avantages de la modularité, de nombreuses méthodes pour définir les modules d'un système au cours du processus de conception ont été proposées. Certaines de ces méthodes sont mentionnées dans (Höltkä-Otto & de Weck 2007). Dans (Benkamoun et al. 2014), on donne un aperçu général de la modularité le long du processus de développement d'un système de production : modularité dans la conception du produit, modularité dans la conception du processus de fabrication et modularité dans la conception du système de fabrication. Dans cette section, nous allons nous concentrer sur une méthode qui pourrait être adaptée et appliquée à la conception d'un système de production modulaire. Elle peut être appliquée à l'ensemble du système (au niveau supérieur de la

composition du système) ou à ses sous-systèmes ou modules connus. La méthode est basée sur l'utilisation de l'outil 'Design Structured Matrix (DSM)' qui est un outil ou technique d'établissement de correspondance (mappage) entre les différents modules du système (fonctionnels et/ou physiques).

5.1. Matrice de conception structurelle (Design Structure Matrix DSM)

DSM est un outil ou une technique pour gérer la complexité. Beaucoup de problèmes de conception pourraient être abordés par cette technique par exemple la conception d'un système modulaire. DSM se présente sous forme d'une matrice carrée ; elle est utilisée pour mettre en évidence les liens qui existent entre des entités du même type ou de types différents comme les modules qui constituent un système complexe, ou les tâches qui composent un projet complexe. À l'aide de la DSM, on peut modéliser et analyser les dépendances entre les entités. Elle présente certains avantages visuels particuliers, y compris la concision et la mise en évidence de schémas importants. À mesure que le nombre d'éléments ou d'entités augmente, la DSM maintient la facilité de lisibilité. L'analyse à l'aide de la DSM permet de regrouper les fonctions en fonction de leur appartenance à un module. Cette modularité est utile pour contrôler l'émergence des comportements du système, modifier le système avec un impact minimal et augmenter sa robustesse et sa résilience. La DSM soutient également l'analyse des degrés de séparation entre les éléments, les effets indirects de propagation et les comportements émergents. Dans cette section, nous allons montrer comment la DSM peut être utilisée pour concevoir un système modulaire qui pourrait être un système de production (Lameche et al. 2017a). Cette méthode est résumée dans les étapes suivantes (voir aussi Figure 31) :

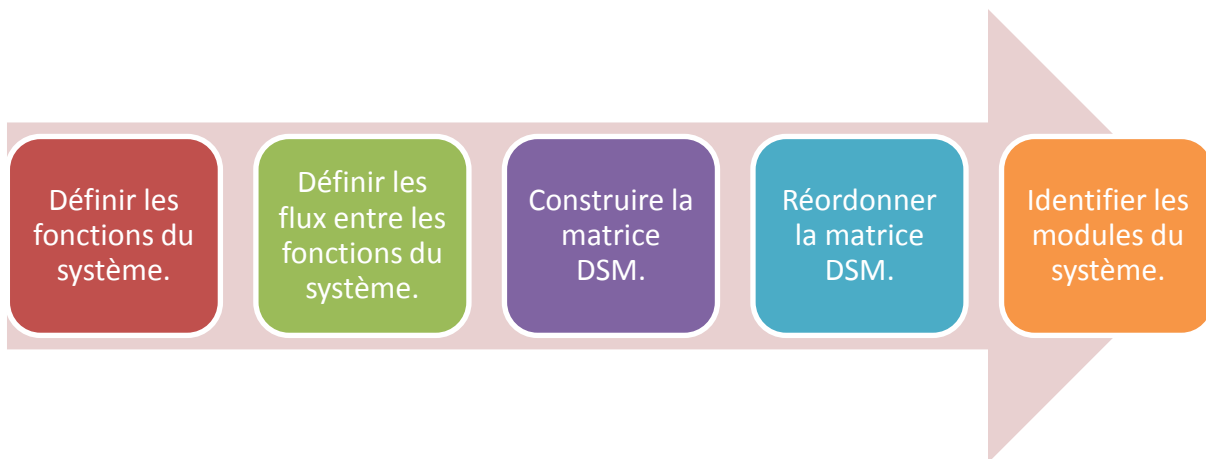


Figure 31 : Les étapes de la méthodologie proposée pour concevoir un système modulaire.

- **Étape 01** : définir toutes les exigences fonctionnelles du système. À partir de ces exigences, définir toutes les fonctions qui doivent être réalisées par le système.
- **Étape 02** : créer le graphique ou le diagramme qui montre les liens ou les flux (matériels ou non matériels) entre les fonctions du système identifiées dans l'étape 01.

- **Étape 03** : à partir du diagramme construit pendant l'étape 02, créer la DSM qui affiche les liens entre les fonctions du système.
- **Étape 04** : réorganiser la DSM pour la rendre la plus diagonale (par blocs) possible. Prenant en compte toutes les contraintes sur les fonctions du système.
- **Étape 05** : à partir de la nouvelle matrice DSM obtenue, regrouper les fonctions du système et identifier ses modules.

5.1.1. Exemple d'application de la méthode DSM

Afin de faciliter la compréhension de la méthode, nous allons traiter un exemple simple pour la définition de l'architecture modulaire.

Étape 01 : déterminer les fonctions du système.

Il existe plusieurs méthodes et techniques qui peuvent être utilisées pour identifier les fonctions d'un système. La méthode FAST (Function Analysis System Technique), par exemple, est l'une de ces méthodes. Les fonctions qui doivent être considérées dans cette analyse sont les fonctions du service, les autres types de fonctions comme les contraintes ne seront pas considérées. Pour cet exemple, on suppose que, après l'analyse des exigences fonctionnelles du système, les fonctions qui doivent être réalisées par le système sont celles listées dans le tableau suivant (Tableau 3).

Fonction du système	Acronyme	Description
Fonction 01	F01	/
Fonction 02	F02	/
Fonction 03	F03	/
Fonction 04	F04	/
Fonction 05	F05	/
Fonction 06	F06	/
Fonction 07	F07	/
Fonction 08	F08	/
Fonction 09	F09	/
Fonction 10	F10	/
Fonction 11	F11	/
Fonction 12	F12	/
Fonction 13	F13	/
Fonction 14	F14	/
Fonction 15	F15	/
Fonction 16	F16	/

Tableau 3 : Définition des fonctions du système.

Étape 02 : Déterminer les flux échangés entre les fonctions du système.

Une fois toutes les fonctions du système sont identifiées, les liens ou les interfaces entre eux doivent être identifiés aussi. Le lien entre deux fonctions peut être un échange d'un flux de matière, un flux d'énergie, un flux de données ou tout autre flux physique ou intangible. Il existe différentes méthodes et techniques pour identifier ces interfaces. La méthode SADT (Structured Analysis And

Design Technique) est l'une de ces méthodes. Pour cet exemple, on suppose que les liens entre les fonctions identifiées précédemment sont ceux montrés dans la figure ci-dessous (Figure 32).

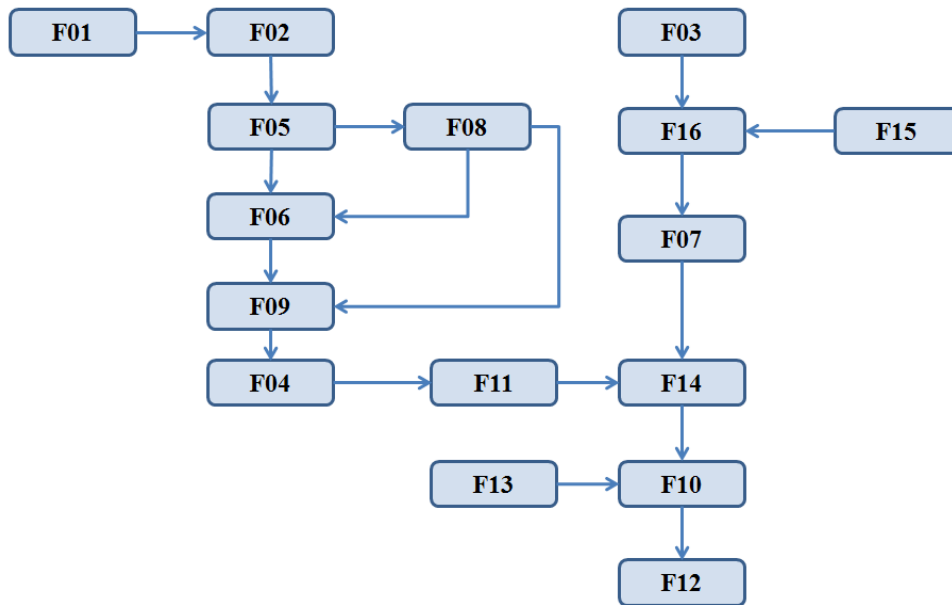


Figure 32 : Identification des interfaces entre les fonctions du système.

Étape 03 : Construire la matrice DSM

La matrice DSM est construite à partir des fonctions du système et les interfaces entre elles. La Figure 33 montre la matrice DSM que nous avons obtenue. Dans cette matrice, les fonctions du système sont listées le long des lignes et des colonnes. Chaque croix ou marque (X) dans la matrice indique la présence d'une relation directe entre deux fonctions du système (par exemple, une dépendance, un flux d'information, d'énergie, physique, ou tout autre flux échangé entre ces deux fonctions). Au lieu d'une simple marque pour indiquer la présence d'une relation ou d'un lien entre les fonctions du système, il est également possible d'utiliser des nombres pour indiquer par exemple : l'intensité de la relation, la probabilité d'un changement dans une fonction provoque un changement dans l'autre, etc. De nombreuses extensions de la DSM sont possibles.

		System functions															
		F01	F02	F03	F04	F05	F06	F07	F08	F09	F10	F11	F12	F13	F14	F15	F16
System functions	F01		X														
	F02					X											
	F03																X
	F04											X					
	F05						X		X								
	F06							X		X							
	F07														X		
	F08						X			X							
	F09				X												
	F10												X				
	F11													X			
	F12														X		
	F13											X					
	F14										X						
	F15															X	
	F16							X									X

Figure 33 : La matrice DSM du système (non réordonnée).

Étape 04 : Réordonner la matrice DSM

La matrice DSM doit être réarrangée pour rapprocher les croix vers la diagonale en échangeant les lignes et les colonnes de la matrice par paire pour obtenir une configuration différente de la matrice. De cette façon, les données ne sont pas modifiées. Ensuite, des ensembles ou des groupes de fonctions doivent être formés. Les fonctions les plus couplées doivent être regroupées ensemble pour former les modules du système tout en essayant de minimiser les interactions entre ces ensembles. Plus il existe d’interfaces entre deux fonctions, plus ces fonctions sont couplées. Les groupements obtenus représentent un cadre utile pour réorganiser l’architecture du système et se focaliser sur les interfaces entre les modules. La Figure 34 montre la nouvelle matrice DSM obtenue.

		System functions															
		F01	F02	F05	F06	F08	F09	F03	F07	F15	F16	F04	F10	F11	F12	F13	F14
System functions	F01	X															
	F02		X														
	F05			X	X												
	F06						X										
	F08																
	F09											X					
	F03							X	X	X							
	F07								X								X
	F15									X							
	F16							X									
	F04											X	X				
	F10											X		X			
	F11																X
	F12																X
	F13												X				
	F14												X				

Figure 34 : La matrice DSM du système (réordonnée).

L'exemple présenté est simple, l'objectif est d'expliquer la méthode proposée qui peut être appliquée pour la conception des systèmes de production modulaires. Dans cet exemple, la déduction des modules du système était facile, on pouvait les identifier sans même avoir recours à la méthode proposée, mais quand il s'agit des systèmes plus complexes, il n'est pas facile de déduire les modules. Dans ce cas, la matrice sera plus large que celle de l'exemple présenté (voir Figure 33). Dans ce cas, grouper les fonctions du système (celles qui sont fortement couplées) pour former les modules du système ne sera pas une tâche facile. Donc, pour remédier (faciliter le groupement des fonctions) à cette difficulté, des algorithmes pour aider à réordonner la matrice DSM pourraient être développés, les croix doivent être rapprochées le plus possible de la diagonale de la matrice. Ensuite, l'identification ou la déduction des modules du système pourrait être une tâche simple.

Étape 05 : Identifier les modules du système

Une fois la matrice DSM est réorganisée et que la plupart des croix (les marques (X)) de la matrice sont aussi proches que possible de la diagonale, on forme des ensembles de fonctions ; ces ensembles devraient regrouper les fonctions les plus couplées (voir Figure 34). Ces ensembles représentent les futurs modules du système ; chaque module implémentera les fonctions de l'ensemble correspondant. Dans cet exemple, nous avons identifié les modules montrés par les figures (Figure 34 et Figure 35). Les croix qui n'appartiennent à aucun ensemble représentent les interfaces entre les modules du système.

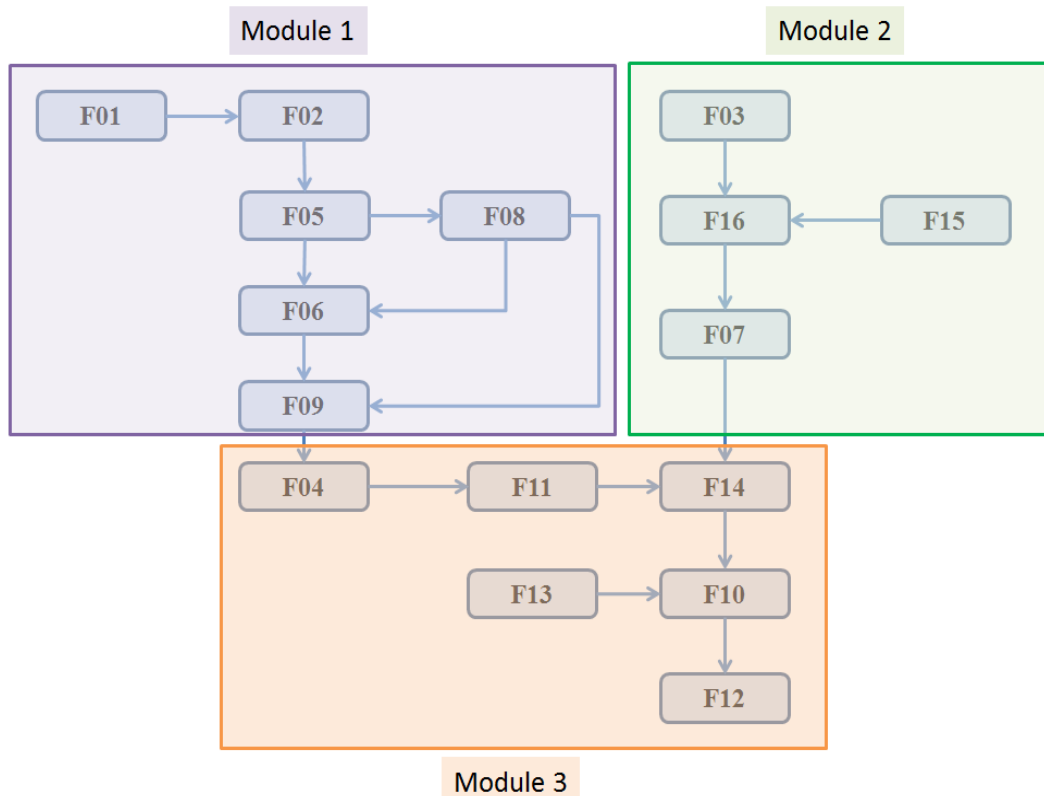


Figure 35 : Identification ou déduction des modules du système.

5.1.2. Application de la modularité pour la conception d'un système logistique reconfigurable

Parmi, les plates-formes d'application de notre méthode générique de conception de système logistique reconfigurable qui est l'objet de notre travail de thèse de Doctorat, un système logistique. C'est un système composé d'AGVs (Automated Guided Vehicles) comme moyen principal pour le transport, à l'aide des chariots, des entités (produits, outils, etc.) entre les différents points de l'atelier, cela offrira au système transitaire (en l'occurrence de transport) plus de flexibilité et d'agilité. Ce système devrait s'adapter à son environnement le plus rapidement possible. Pour la conception de ce système, nous avons utilisé la méthode DSM. La Figure 36 donne un aperçu général de ce système et du flux des produits. Le processus du système logistique est constitué des étapes suivantes :

- **Conditionnement :** Les produits sont reçus de l'extérieur ; ils sont conditionnés dans la zone de conditionnement et envoyés vers la zone magasin pour être stockés dans l'entrepôt (le magasin).
- **Stockage :** Lorsque les produits arrivent à l'entrepôt, ils sont d'abord placés dans une zone d'attente appelée 'zone d'attente en entrée du magasin'. C'est une zone tampon pour les produits qui attendent d'être mis dans l'entrepôt.
- **Déstockage :** Lorsque les produits sont demandés par le client, ils sont recherchés dans l'entrepôt, et envoyés vers la zone de colisage pour être préparés et envoyés aux clients. Lorsque les produits doivent être envoyés vers la zone de colisage, ils sont d'abord placés dans une zone d'attente

appelée ‘zone d’attente en sortie du magasin’. C’est comme une zone tampon pour les produits en attente d’être envoyés vers la zone de colisage.

- Colisage : Une fois la préparation des produits dans la zone de colisage est terminée, ils seront expédiés vers le client.

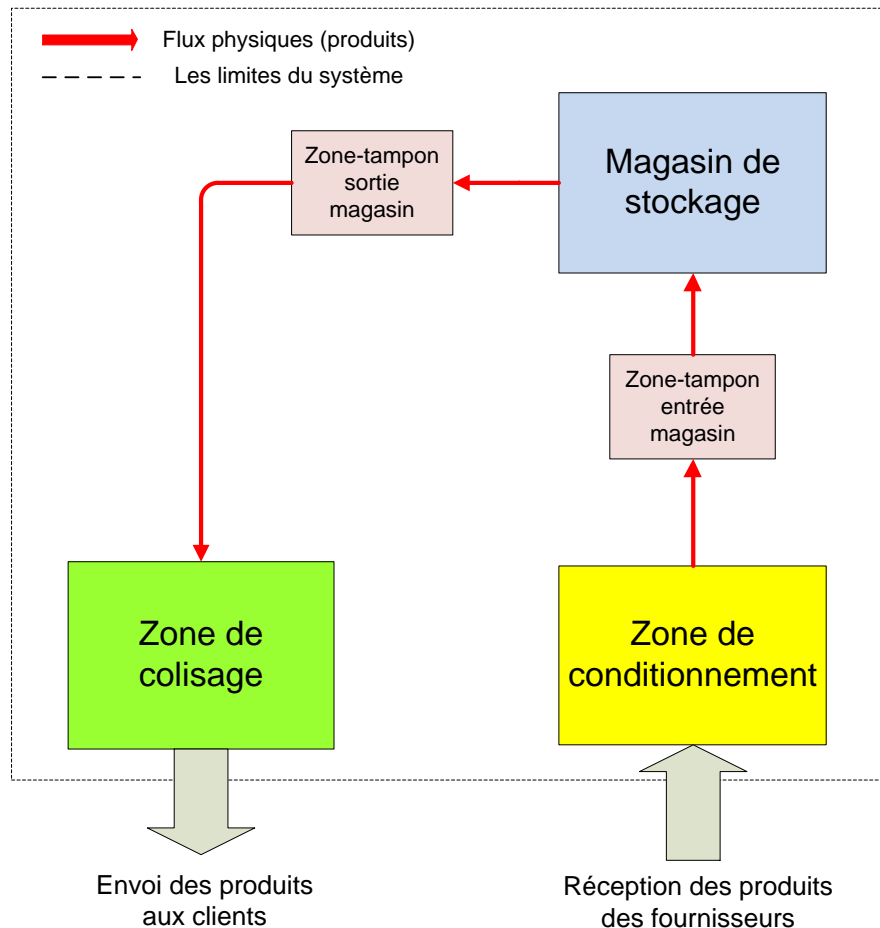


Figure 36 : Synopsis du système logistique.

Les étapes suivantes détaillent l’application de la méthodologie sur le cas d’étude décrit ci-dessus.

Étape 01 : Déterminer les fonctions du système.

Le Tableau 4 résume les exigences fonctionnelles identifiées du système.

Identifiant	Description
F01	Dans la zone de conditionnement, le système doit avoir des points spécifiques (points de travail) où les produits seront conditionnés.
F02	Dans la zone de conditionnement, le système doit avoir des points spécifiques (points de départ) où les chariots pleins seront mis en attente pour être transportés vers le magasin.
F03	Dans la zone de conditionnement, le système doit avoir des points spécifiques (points de réception) où les chariots vides seront reçus.
F04	Dans la zone magasin, le système doit avoir une zone spécifique où les chariots pleins en attente d’être vidés dans le magasin seront mis en attente.
F05	Dans la zone magasin, le système doit avoir dans une zone spécifique où les chariots vides en attente d’être transportés vers la zone de conditionnement

	seront mis en attente.
F06	Dans la zone magasin, le système doit avoir une zone spécifique où les chariots pleins en attente d'être transportés vers la zone de colisage seront mis en attente.
F07	Dans la zone de colisage, le système doit avoir des points spécifiques (points de réception) où les chariots pleins seront reçus.
F08	Dans la zone de colisage, le système doit avoir des points spécifiques (points de travail) où les produits seront préparés pour être expédiés au client.
F09	Dans la zone de colisage, le système doit avoir des points spécifiques (points de départ) où les chariots vides seront mis en attente d'être transportés vers la zone de colisage.
F10	Le système doit transporter les chariots pleins de la zone de conditionnement à la zone magasin.
F11	Le système doit transporter les chariots pleins de la zone magasin à la zone de colisage.
F12	Le système doit transporter les chariots vides de la zone magasin à la zone de conditionnement
F13	Le système doit transporter les chariots vides de la zone de colisage à la zone de conditionnement.
F14	Le système doit gérer les transporteurs, cela inclut : <ul style="list-style-type: none"> - Sélectionner un transporteur (AGV) pour exécuter une mission. - Superviser l'état de chaque transporteur. - Demander des données aux transporteurs.
F15	Le système doit gérer les demandes de transport, cela inclut : <ul style="list-style-type: none"> - Recevoir les demandes de transport. - Ordonnancer les demandes de transport. - Créer les missions de transport. - Affecter les missions de transport aux transporteurs - Superviser l'exécution des missions de transport.
F16	Le système doit gérer les conflits entre les transporteurs au niveau des intersections.

Tableau 4 : exigences fonctionnelles du système.

Étape 02 : Déterminer les flux (interfaces) entre les fonctions du système.

Après l'analyse des flux dans le système et les différentes étapes et opérations par lesquelles les produits doivent passer, la Figure 37 montre les liens entre les différentes fonctions du système identifiées dans l'étape 01 (voir Tableau 4).

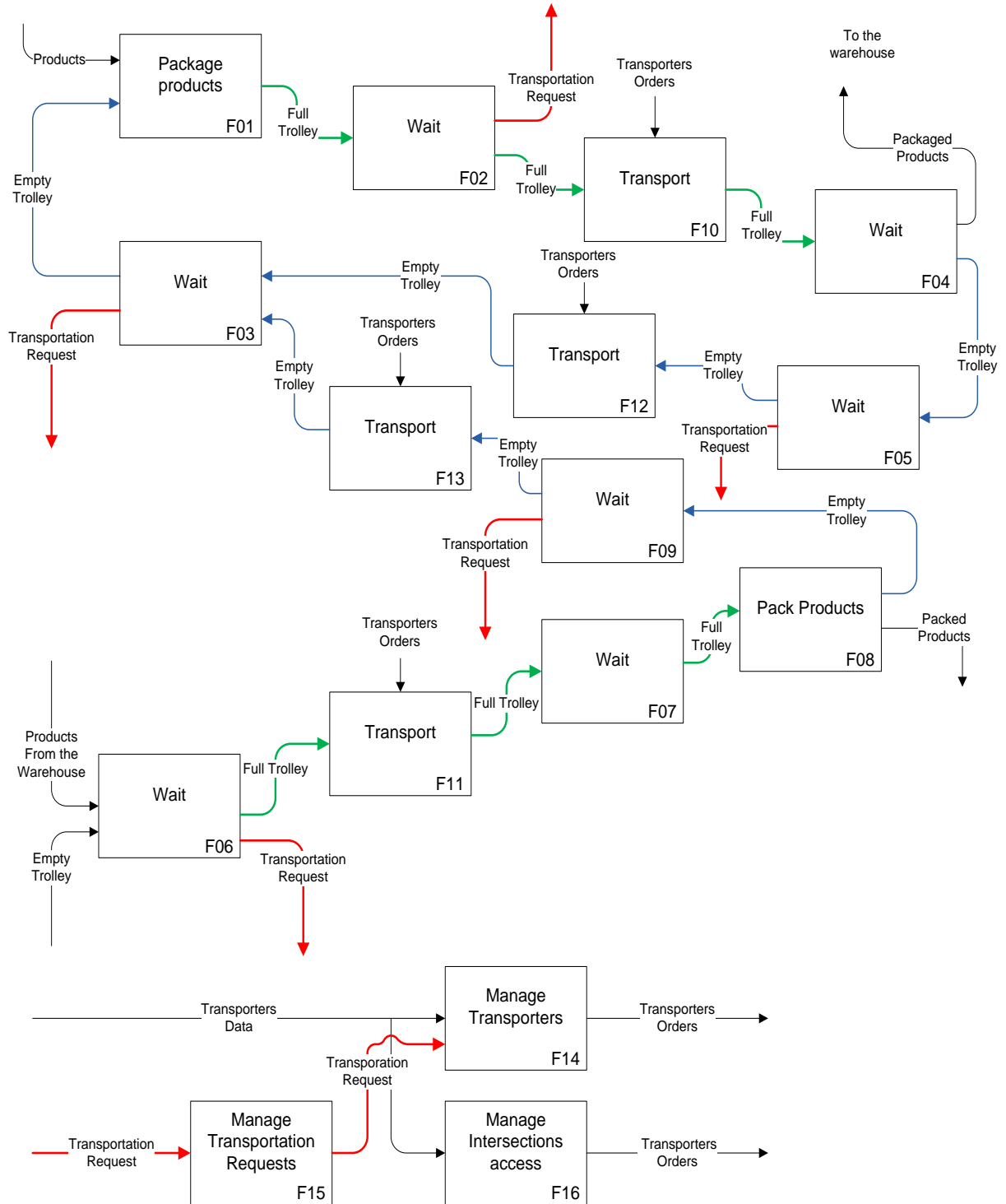


Figure 37 : Flux entre les fonctions du système.

Étape 03 : Construire la matrice DSM

Pour rappel, la Matrice DSM est construite à partir des fonctions identifiées du système et les interfaces entre elles. La Figure 38 montre la matrice DSM que nous avons obtenue à partir du Tableau 4 et la Figure 37.

		System functions															
		F01	F02	F03	F04	F05	F06	F07	F08	F09	F10	F11	F12	F13	F14	F15	F16
System functions	F01		X														
	F02										X						X
	F03	X															X
	F04					X											
	F05						X						X				X
	F06							X				X					X
	F07								X								
	F08									X							
	F09										X			X			X
	F10				X							X					
	F11							X					X				
	F12			X										X			
	F13			X											X		
	F14										X	X	X	X		X	
	F15														X		X
	F16										X	X	X	X			X

Figure 38 : La matrice DSM du système (non réordonnée).

Étape 04 : Réordonner la matrice DSM

La DSM peut être réarrangée pour obtenir des groupes de fonctions qui ont beaucoup d'interactions entre elles tout en en minimisant les interactions entre ces groupes de fonctions. De cette façon, les données ne sont pas modifiées, mais les lignes et les colonnes de la matrice ne sont échangées que par paire pour obtenir une disposition différente de la matrice. L'objectif est de rapprocher les marques (X) ou les croix de la diagonale de la matrice. Les fonctions les plus couplées doivent être regroupées pour former les modules du système. Plus il existe d'interfaces entre deux fonctions ; plus ces fonctions sont couplées. Cependant, dans les systèmes réels, certaines contraintes doivent être prises en considération avant de regrouper les fonctions du système. Certaines fonctions ne peuvent pas être implémentées dans le même module même si elles sont fortement liées, car, par exemple, elles ne sont pas exécutées au même endroit, ou parce qu'elles ne sont pas du même type (fonction mécanique et fonction de contrôle généralement ne sont pas groupées ensemble dans le même module).

		System functions															
		F01	F08	F02	F03	F07	F09	F05	F04	F06	F10	F11	F12	F13	F14	F15	F16
System functions	F01	X	X														
	F08	X	X				X										
	F02			X	X	X	X				X					X	
	F03	X		X	X	X	X									X	
	F07		X	X	X	X	X										
	F09			X	X	X	X						X			X	
	F05							X	X	X			X			X	
	F04							X	X	X							
	F06											X				X	
	F10								X		X	X	X				
	F11					X					X	X	X	X			
	F12				X						X	X	X	X			
	F13				X						X	X	X	X			
	F14										X	X	X	X	X	X	
	F15														X	X	
	F16										X	X	X	X	X	X	

Figure 39 : Liens entre les fonctions du système (après substitution et simplification).

En appliquant la matrice DSM à notre problème industriel, nous avons découvert les limites de la méthode. Parfois, même si deux fonctions sont couplées, elles ne peuvent pas être mises dans le même module. Par exemple les croix/marques (X), dans le coin inférieur droit et légèrement vers la gauche montrent que les fonctions (F10, F11, F12 et F13) sont couplées avec les fonctions (F14, F15 et F16), donc logiquement, ces fonctions devaient être groupées dans le même module, mais malheureusement il y a une contrainte qui empêche cette possibilité. Parce que, les fonctions (F10, F11, F12 et F13) sont toutes des fonctions de transport, et elles seront exécutées par le même module qui est le transporteur (AGV dans ce cas) alors que les fonctions (F14, F15 et F16) sont toutes des fonctions de gestion, et seront exécutées par le système de contrôle. Donc, parfois quelques fonctions couplées du système ne peuvent pas être groupées dans le même module à cause des contraintes liées à la réalité. Le couplage entre ces fonctions se traduit par les interfaces du système.

Pour simplifier l'identification des modules du système, on peut noter à partir du Tableau 4 et la Figure 38 que :

- Les fonctions F10, F11, F12 et F13 sont les mêmes, elles sont toutes des fonctions de transport, et elles seront exécutées par le même module qui est le transporteur (AGV dans ce cas). Toutes ces fonctions peuvent être remplacées par une seule fonction F010.

- Les fonctions F01 et F08 sont les mêmes (points ou stations de travail), mais elles sont exécutées à différents endroits du système. Ces deux fonctions peuvent être remplacées par une seule fonction F001.
- Les fonctions F02 et F09 sont les mêmes (points de départ), mais elles sont exécutées à différents endroits du système. Ces fonctions peuvent être remplacées par une seule fonction F002.
- Les fonctions F03 et F07 sont identiques (points de réception), mais elles sont exécutées dans des emplacements différents du système. Ces deux fonctions peuvent être remplacées par une seule fonction F003.
- Les fonctions F04, F05 et F06 sont les mêmes (mise en attente dans une zone tampon), mais elles sont exécutées à des emplacements différents du système. Ces fonctions peuvent être remplacées par une seule fonction F004.
- Les fonctions F14, F15 et F16 restent les mêmes.

		System functions							
		F001	F002	F003	F004	F010	F14	F15	F16
System functions	F001	X	X						
	F002		X	X		X		X	
	F003	X	X	X				X	
	F004				X	X		X	
	F010			X	X	X			
	F14					X	X	X	X
	F15						X	X	X
	F16					X	X	X	X

Figure 40 : la matrice DSM après la substitution des fonctions du système.

Étape 05 : Identifier les modules du système.

La Figure 40 montre la nouvelle DSM obtenue après la substitution des fonctions (voir étape précédente).

- Le module représenté par le rectangle vert implémente la fonction F001 ; ce module représente le poste de travail où l'opérateur pourrait exécuter les différentes tâches telles que le conditionnement, le colisage ou autre. Il dispose des interfaces mécaniques avec les stations de réception et les stations de départ.
- Le module orange implémente les fonctions F002 et F003 ; ce module représente la station de réception et la station de départ. Il peut être configuré pour être l'une ou l'autre. Ce module dispose d'une interface informationnelle avec le système de pilotage (envoyer les demandes

de transport), et une interface mécanique avec l'AGV (pour échanger des chariots, prise et dépose).

- Le module jaune implémente la fonction F004 ; ce module représente une zone tampon (pour les chariots vides ou pleins). Il dispose d'une interface informationnelle avec le système de pilotage (envoyer les demandes de transport), et une interface mécanique avec l'AGV (échanger des chariots, prise et dépose).
- Le module bleu met en œuvre la fonction F010 qui est la fonction de transport. Ce module représente le transporteur (AGV dans ce cas). Il possède une interface informationnelle avec le système de contrôle (envoyer des données, et recevoir des missions de transport). Il dispose également d'une interface mécanique pour échanger les chariots avec d'autres modules du système (stations de réception, stations de départ et les zones tampons).
- Le module gris implémente les fonctions F14, F15 et F16 ; ce module représente le système de pilotage des transporteurs. Il dispose des interfaces informationnelles avec la plupart des modules du système, il reçoit des demandes de transport des (stations de départ, stations de réception, et des zones tampons), et envoie les missions de transport aux transporteurs. Il reçoit également des données de ces transporteurs pour la supervision.

La figure suivante (Figure 41) montre les flux du système avec la mise en évidence, à l'aide du code couleur, des fonctions exécutées par chaque module du système.

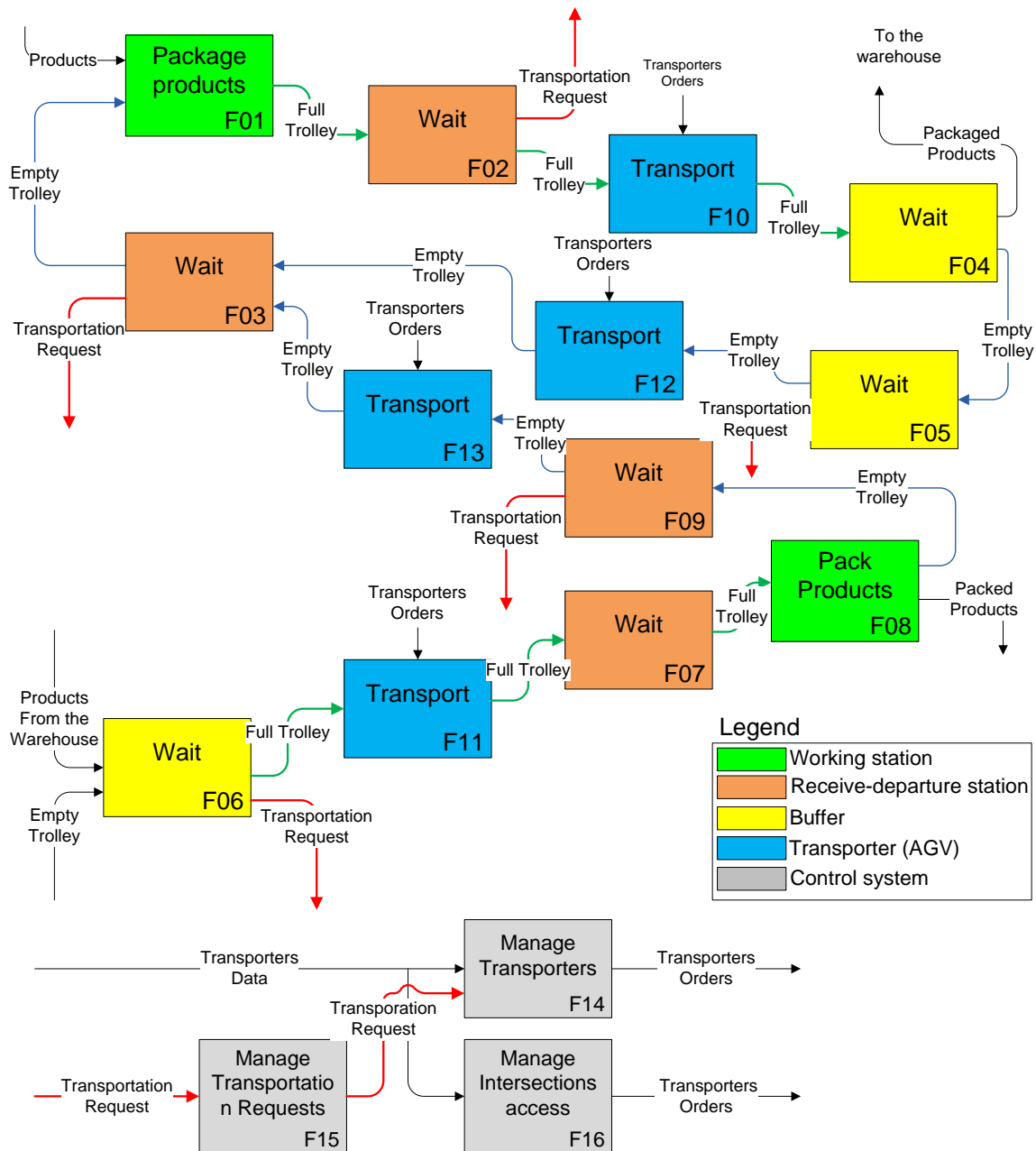


Figure 41 : Flux entre les fonctions du système et mise en évidence des modules identifiés.

6. Conclusion

Les RMS sont les systèmes de production les plus intéressants pour les entreprises manufacturières pour rester compétitive et pour survivre dans le marché actuel qu'est caractérisé par un très haut niveau de compétitivité. En raison de leur structure changeante (physique et logique) à différents niveaux de composition du système, un RMS est considéré comme un système complexe et sa conception n'est pas une tâche facile. Par conséquent, la plupart des méthodes et techniques, utilisées pour concevoir les systèmes complexes, pourraient être réutilisées pour concevoir un RMS.

Certaines modifications, adaptations et améliorations doivent être apportées à ces méthodes et techniques avant de les appliquer pour la conception d'un RMS.

La modularité est l'une des techniques utilisées pour gérer la complexité d'un système le long de son cycle de vie (phase de conception, phase de production, phase d'utilisation et même phase de démantèlement et de retrait de service). La modularité présente de nombreux avantages, mais présente également certains inconvénients, comme la difficulté de concevoir les interfaces, la perte de performances globales en raison de ces interfaces ajoutées entre les modules (Höltkä et al. 2005). On a présenté une méthode basée sur l'outil DSM (Design Structured Matrix) qui est un outil déjà utilisé pour concevoir des produits. Nous avons, montré que cette méthode pourrait être utilisée pour concevoir un système de production modulaire ; elle pourrait être appliquée sur l'ensemble du système ainsi que sur ses sous-systèmes. De plus, des algorithmes pourraient être développés pour aider à réorganiser la matrice DSM ; cela facilitera le regroupement des fonctions du système, puis la déduction de ses modules.

On a mentionné dans le chapitre 1, dans la conclusion, qu'une méthodologie générique qui couvre la conception d'un RMS est manquante. Le chapitre suivant est consacré à la description de cette méthodologie.

Chapitre 4 : Une méthodologie générique pour la conception des systèmes manufacturiers reconfigurables

1. Introduction

Actuellement, le marché est caractérisé par un haut niveau de compétitivité et des variations très fréquentes et soudaines. L'examen critique des systèmes de production dédiés (Dedicated Manufacturing Lines DML) et les systèmes de production flexibles (Flexibles manufacturing Systems FMS), montre que ces systèmes ne peuvent pas satisfaire les exigences imposées par le marché ; ces exigences sont principalement : le coût et la qualité des produits et la réactivité du système manufacturier aux changements du marché. Par conséquent, pour rester compétitives, les entreprises doivent concevoir des systèmes de production qui assurent des produits peu coûteux et de haute qualité avec une réactivité adéquate aux changements du marché. Le système manufacturier reconfigurable RMS est ce nouveau paradigme ; il est censé être assez réactif pour faire face aux changements soudains du marché tout en maintenant une haute qualité des produits à faibles coûts.

Il est évident qu'il est difficile d'établir une méthode globale, standard et détaillée qui répond à tous les problèmes de conception des RMSs en raison de leur complexité. La plupart des méthodes proposées dans la littérature (voir chapitre 1) ne traitent pas le problème de conception d'un RMS dans son ensemble, mais elles se limitent à une partie du problème. Donc, le principal défi d'un RMS est sa conception. Pour résoudre ce problème, l'ingénierie des systèmes IS, qui est une nouvelle discipline pour développer des systèmes complexes, pourrait être appliquée à la conception des RMSs. Cette discipline regroupe les meilleures pratiques des industriels dans le développement des systèmes complexes réussis. Ces pratiques sont généralement : les processus, les méthodes ou les méthodologies qui sont les résultats d'une longue expérience dans le développement de systèmes complexes.

L'objectif de ce chapitre est de présenter une méthode d'intervention globale pour la conception des RMSs, qui va de la définition du besoin jusqu'à la validation du déploiement du système. Cette méthode s'appuie fortement sur l'approche l'IS telle qu'elle est définie par la norme ISO/IEC/IEEE-15288 (ISO/IEC/IEEE15288 2015). Elle définit les besoins des parties prenantes puis organise les différentes activités jusqu'à la validation finale du système. Cette méthode d'intervention englobe la définition du contexte et du besoin, la définition et la spécification des exigences, la conception de la solution (architecture logique, architecture physique et choix technologique), la vérification et la validation du système. Si cette méthodologie est fortement ancrée sur les principes méthodologiques de l'IS, elle met en avant les évolutions nécessaires de certains concepts de l'IS pour prendre en compte la spécificité des systèmes reconfigurables.

2. Canevas de la méthodologie de conception des RMSs

On a conclu précédemment (voir chapitre 2) que le modèle en « V » est le mieux adapté pour le développement des systèmes techniques. On propose ici une version différente du cycle en « V » adaptée au développement des systèmes reconfigurables. Le système dans sa globalité ainsi que ses sous-systèmes seront développés suivant le cycle en « V ». Pour gérer la variation des exigences et minimiser les risques associés aux ambiguïtés (définition non claire des exigences, des exigences non fixes, etc.), on doit passer suffisamment de temps pour la définition du système avant de passer à sa réalisation. Il faut par contre souligner que dans le cadre des systèmes reconfigurables, il faudrait compléter ce cycle en « V » pour y inclure les phases de reconfiguration, qui sont des phases normales de la vie du système. Il faut donc, dès la définition initiale de ce cycle en « V » prévoir les retours dans la phase de conception qui seront nécessaires pour la reconfiguration. Cette évolution du cycle en « V » est un apport important du canevas méthodologique proposé dans ce chapitre.

Les différentes étapes du canevas de la méthodologie sont l'application de quelques processus de la norme ISO/IEC/IEEE-15288, principalement, les processus techniques. L'approche IS est une approche générale pour développer les systèmes techniques. La norme ISO/IEC/IEEE 15288 qui est la plus mature des normes de l'IS dicte ce qu'il faut faire pour faciliter le développement de ces systèmes, mais elle ne précise pas comment le faire. C'est le rôle des méthodes et des outils qui seront utilisés pour accomplir chaque activité ou tâche des différents processus de la norme. Donc les processus détaillés dans la norme ISO/IEC/IEEE 15 288 sont une abstraction de ce qu'il faut faire pour développer tout type de système technique. En appliquant les méthodes et les outils appropriés, on peut diriger ces processus pour développer un type spécifique de systèmes techniques. On va utiliser quelques processus techniques de cette norme avec l'application des méthodes et des outils appropriés pour mettre en place une méthodologie qui facilite le développement des systèmes manufacturiers reconfigurables. La Figure 42 résume la démarche suivie pour l'implémentation de cette norme.

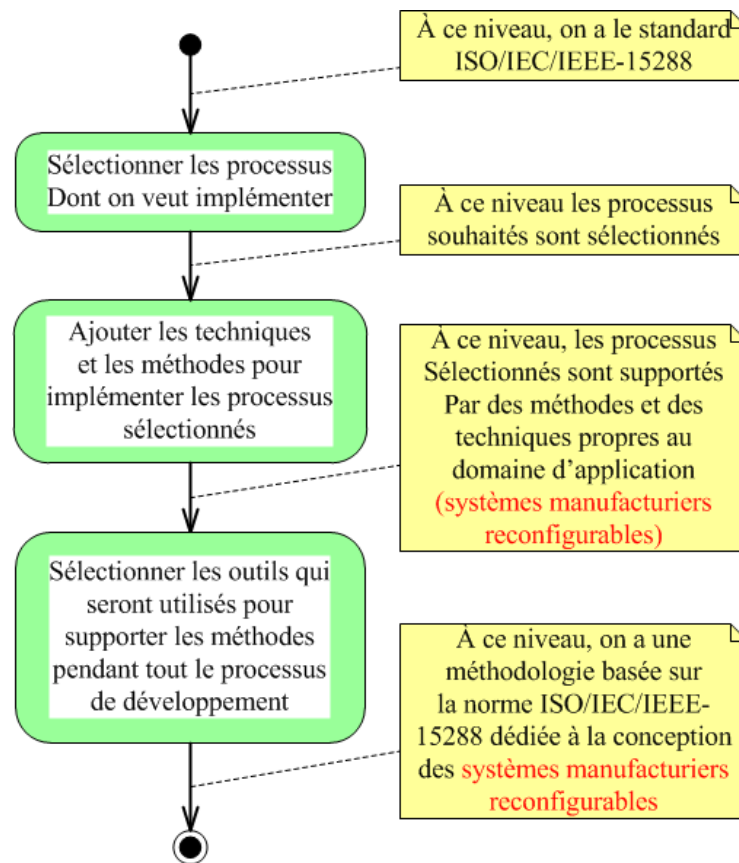


Figure 42 : démarche globale d'implémentation de la norme ISO/IEC/IEEE-15288.

La Figure 43 donne une vision globale du canevas de la méthodologie. Dans cette méthodologie, pour concevoir un RMS, nous commençons par la définition des besoins des parties prenantes (PPs) et nous finissons par la validation du système dans son environnement opérationnel. Pour le moment, la méthodologie n'adopte pas tous les processus techniques proposés dans la norme ISO/IEC/IEEE-15288. Les processus implémentés sont ceux présentés dans la Figure 43 à l'exception du processus d'implémentation et de réalisation. Les processus sont exécutés dans l'ordre indiqué par le chemin vert, mais la méthodologie est agile, donc les retours aux processus précédents pour effectuer les corrections nécessaires sont toujours possibles. L'expérience a révélé qu'il fallait consacrer plus de temps à la définition des besoins des PPs et aux processus d'analyse des exigences du système avant d'aller plus loin dans le développement du système.

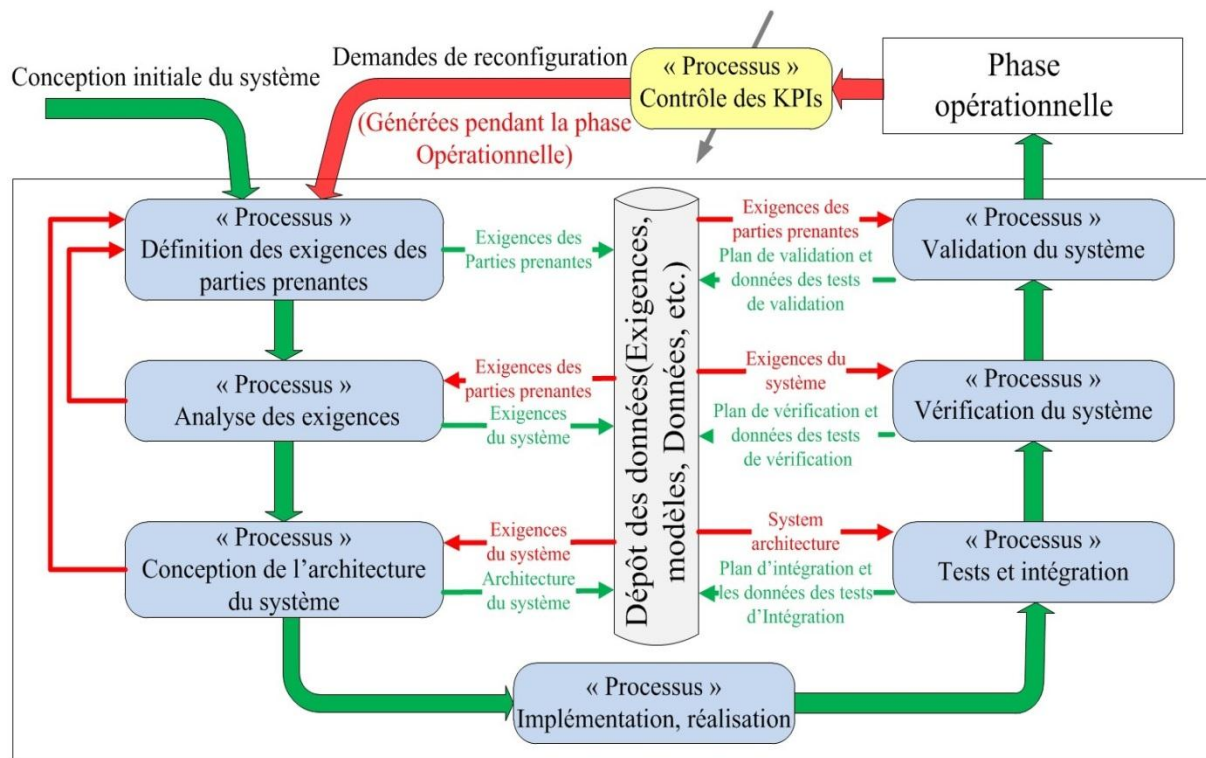


Figure 43 : Canevas de la méthodologie proposée pour la conception d'un RMS.

Nous avons mentionné que la méthodologie proposée suit une version modifiée du cycle en « V ». Lorsque le RMS est déployé pour la première fois, il sera dans sa configuration initiale ; nous l'appelons la configuration de déploiement. Au cours de sa phase opérationnelle, le système devrait être reconfiguré pour s'adapter à son contexte de production. Ainsi, chaque fois que la reconfiguration du système est nécessaire, une demande de reconfiguration est générée. Cette demande déclenchera la reprise de la méthodologie pour concevoir une nouvelle configuration du système qui répond aux exigences du nouveau contexte.

Notons que nous proposons deux ajouts par rapport au schéma classique du cycle en « V » (voir Figure 43). D'une part, nous proposons d'intégrer au cycle de vie du système le processus « contrôle des KPIs (Key Performance Indicators) » représenté en jaune. Au cours du fonctionnement du système de production reconfigurable, une observation des indicateurs de performance du système permet de détecter la nécessité d'une reconfiguration. D'autre part, nous proposons une flèche reliant ce processus « Vie du système » au processus « Définition des exigences des PP », ces exigences étant préalables au processus de reconfiguration. Ce processus de reconfiguration reprend les étapes de la conception initiale pour prendre en compte les nouvelles exigences du système. Bien sûr, il ne faut pas reconcevoir la totalité du système. Nous voulons ainsi montrer que pour un système reconfigurable, cette reconception est une étape normale du cycle de vie du système.

2.1. Reconfiguration du système sans évolution

Dans le cas où la reconfiguration du système, avec une solution déjà conçue (modules physiques et modules logiques), pourrait répondre aux exigences du nouveau contexte de production, seules la redéfinition ou la sélection de la configuration du système sont nécessaires. Par conséquent, la méthodologie sera reprise à partir du processus de conception de l'architecture du système, seulement la configuration du système sera considérée. Cela inclut la conception de l'architecture du système (la disposition des différents modules dans l'atelier) et la configuration des modules du système. Lors de la sélection de la nouvelle configuration du système, il est possible d'avoir plusieurs configurations qui répondent aux exigences du nouveau contexte de production, dans ce cas, une phase d'évaluation et de comparaison pourrait être nécessaire. Cette phase permet de comparer les différentes configurations possibles du système en utilisant un ou plusieurs critères. Par exemple, le coût de la production, le temps de cycle, l'effort et le coût nécessaires pour la reconfiguration du système, etc. peuvent être utilisés comme critères de comparaison. À la fin de cette phase de comparaison, la meilleure configuration selon les critères choisis sera sélectionnée pour être déployée.

La simulation est un outil très puissant qui peut aider à la conception des systèmes complexes. Dans le cas des RMSs, la simulation peut être utilisée pour évaluer, comparer et sélectionner les futures configurations du système. Le chapitre 5 décrit en détail comment la simulation peut être utilisée pour aider à la conception d'un RMS et comment elle a été exploitée afin d'aider à la sélection des futures configurations du système. Une fois la configuration future du système est sélectionnée ; elle devrait être déployée et validée. En conclusion, seuls les processus : « la conception de l'architecture du système » et « la validation du système » seront considérés dans ce cas (voir la Figure 44).

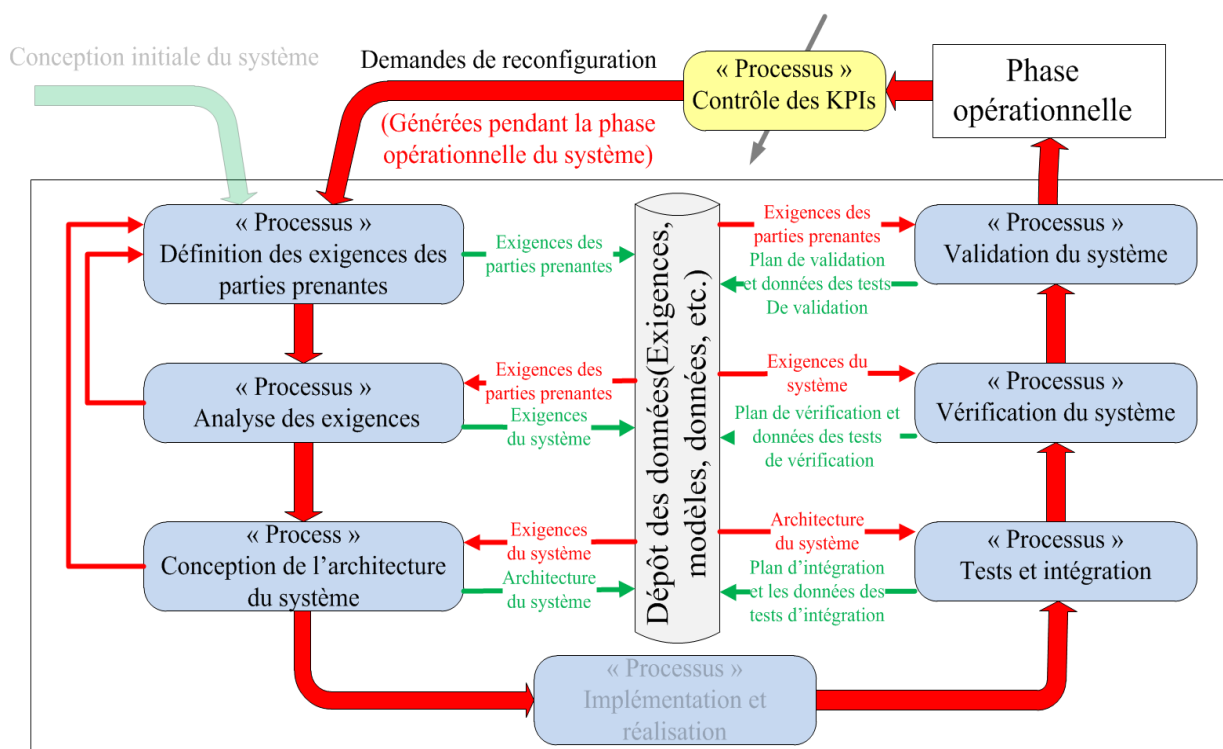


Figure 45 : la reconfiguration du système avec évolution.

2.3. L'échange des données

Le développement des systèmes complexes comme les systèmes de production, y compris les RMSs, implique des personnes et des équipes de différents domaines. La communication entre tous ces acteurs est vitale. Par conséquent, le partage des données du projet (exigences, spécifications, modèles, tests de vérification, tests de validation, rapports de test, etc.) est une solution efficace pour assurer la communication entre ces acteurs. Sur le schéma global de la méthodologie Figure 43, les données à échanger entre les différents processus sont stockées dans un dépôt commun des données, les flèches entrantes/sortantes du dépôt montrent l'échange des données entre les différents processus de la méthodologie. L'exécution de chaque processus nécessitera un ensemble de données et générera en conséquence un autre ensemble de données qui seront stockées dans le même dépôt pour être utilisées par d'autres processus.

Les exigences des PP générées pendant la définition des besoins sont stockées dans le dépôt des données. Ces exigences seront utilisées pendant l'analyse des exigences pour en déduire les exigences système qui seront utilisées à leur tour dans d'autres processus. Les exigences des PP seront aussi utilisées comme un référentiel par rapport auquel le système sera validé dans son environnement opérationnel. Les exigences système seront utilisées par le processus de conception de l'architecture pour concevoir le système. Elles constituent aussi le référentiel par rapport auquel le système sera vérifié (s'il a été bien construit selon les spécifications). Les données de conception du système

(l'architecture du système) qui sont générées pendant le processus de conception et qui sont stockées dans le dépôt des données sous différentes formes, mais principalement des plans et des modèles seront utilisés pour construire le système, et pour l'assembler. Elles représentent aussi un référentiel de vérification du système. Les données de vérification et de validation générées pendant les processus correspondants sont stockées dans le dépôt, ces données seront analysées pour identifier les aléas, ou de prouver la conformité du système aux spécifications ainsi qu'aux besoins des PPs.

2.4. Principe de décomposition

Dans le cas du développement des systèmes complexes, la décomposition est une technique très utile pour gérer la complexité de ces systèmes. Par exemple, si nous voulons développer un système complexe connaissant les principaux sous-systèmes qui le composent, le développement de l'ensemble du système peut être réalisé plus facilement en développant ses sous-systèmes et les interactions entre eux. Si, par exemple, on veut développer un système transitique à base d'AGVs, le développement de ce système dans sa globalité serait une tâche difficile vu sa complexité. Mais, si on décompose le système pour développer l'AGV indépendamment du reste du système, le système de contrôle de l'ensemble des AGVs sera aussi développé indépendamment du reste du système, ainsi que le système de repérage et de guidage des AGVs (les bandes au sol, et les balises sur les murs par exemple). Donc, on voit bien que la décomposition donne une certaine maîtrise de la complexité du système et facilite son développement. Il est par contre nécessaire d'assurer la cohérence entre les développements des différentes parties du système.

Notons enfin que la décomposition va aussi permettre de faciliter les phases de reconfiguration du système. Elle permet en effet de structurer le système en modules dont l'agencement pourra être modifié simplement lors d'une phase de reconfiguration. Si nous reprenons l'exemple du système transitique, si un module « Station de livraison de produits » a été défini, il sera aisé de modifier l'architecture du circuit en ajoutant ou en enlevant de tels modules. Ceci nous amène à développer le concept de modularité (voir chapitre 2 pour plus de détails sur le principe de la modularité).

Les systèmes de production pourraient être décomposés en quatre sous-systèmes (voir Figure 46) :

- Le sous-système à valeur ajoutée, c'est la partie responsable de l'ajout de valeur à la matière première (système d'usinage, système d'assemblage, etc.).
- Le sous-système de stockage.
- Le sous-système logistique ou de manutention.
- Le sous-système de contrôle.

Lors du développement des systèmes manufacturiers, nous commençons généralement par le développement de son sous-système à valeur ajoutée. Le sous-système de stockage et le sous-système

de maintenance sont développés pour supporter le système à valeur ajoutée, les fonctions de ces deux sous-systèmes sont considérées comme des fonctions de support. Le système de contrôle est développé pour piloter et superviser tous ces sous-systèmes. Cette décomposition du système manufacturier en sous-systèmes ainsi que les contraintes de précédence de conception de ces sous-systèmes peuvent être appliquées aux RMSs. Donc, un système manufacturier reconfigurable peut être développé en développant ses sous-systèmes qui devraient également être reconfigurables. Les fonctionnalités d'un RMS et les principes de conception devraient être appliqués pour développer ses sous-systèmes.

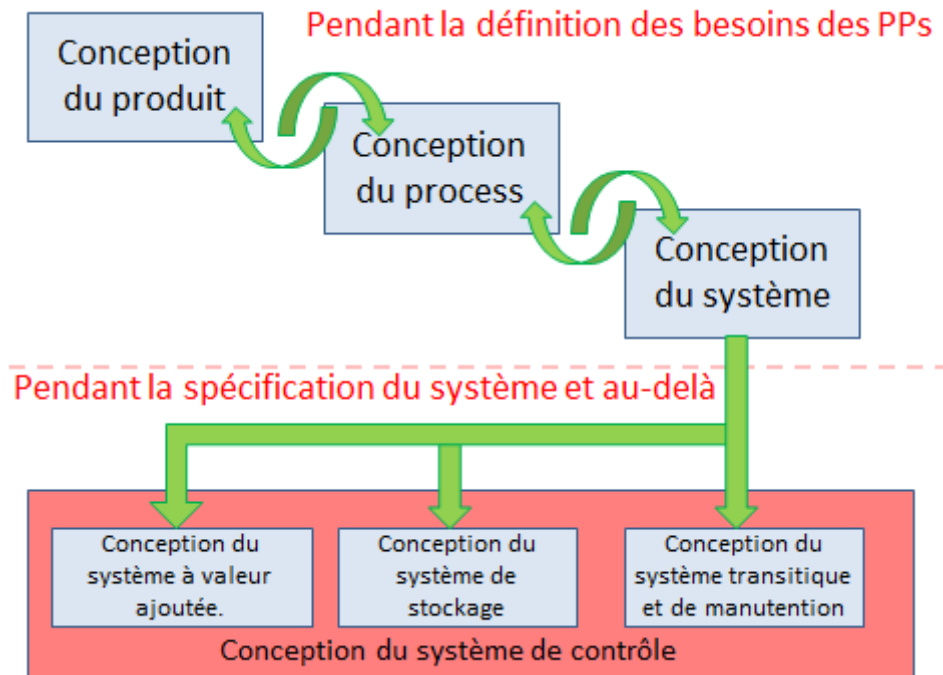


Figure 46 : démarche générale de conception d'un système manufacturier.

3. Les processus et les activités de la méthodologie

Après avoir donné quelques définitions concernant les RMSs, on détaille dans cette section les processus de la méthodologie proposée (Figure 43) pour supporter la conception des RMSs. La Figure 47 liste les activités de chaque processus de la méthodologie. Sur cette figure, on a fait le lien entre les activités à exécuter et les diagrammes SysML qui peuvent être utilisés pour aider à la réalisation de chaque activité. En fait, on a montré les types de diagrammes SysML qui peuvent être à utiliser pendant chaque activité, par exemple :

- Pour identifier les services requis au système, on peut utiliser le diagramme de cas d'utilisation.
- Pour définir les interactions entre le système et ses utilisateurs, on peut utiliser le diagramme de séquence.

- Pour définir chaque fonction que le système doit faire, on peut utiliser le diagramme d'activité et le diagramme de séquence.
- Pour définir l'architecture du système, on peut utiliser le diagramme de blocs pour montrer la composition du système, de ses sous-systèmes, et de ses sous-sous-systèmes, etc. On peut aussi utiliser le diagramme de définition interne du bloc pour définir les interfaces entre les différents constituants du système (composants, sous-systèmes, etc.).
- ...etc.

Dans la méthodologie proposée, on n'est pas obligé d'utiliser le langage SysML pour exécuter les différentes activités. On peut tout à fait faire appel aux méthodes classiques utilisées dans le développement des systèmes techniques ou on peut utiliser des méthodes propres au domaine d'application (les systèmes reconfigurables) de la méthodologie, mais vu qu'on essaie de mettre en place une méthodologie de développement dirigée par les modèles, il est conseillé d'utiliser le type de diagramme indiqué pour chaque activité (voir Figure 47).

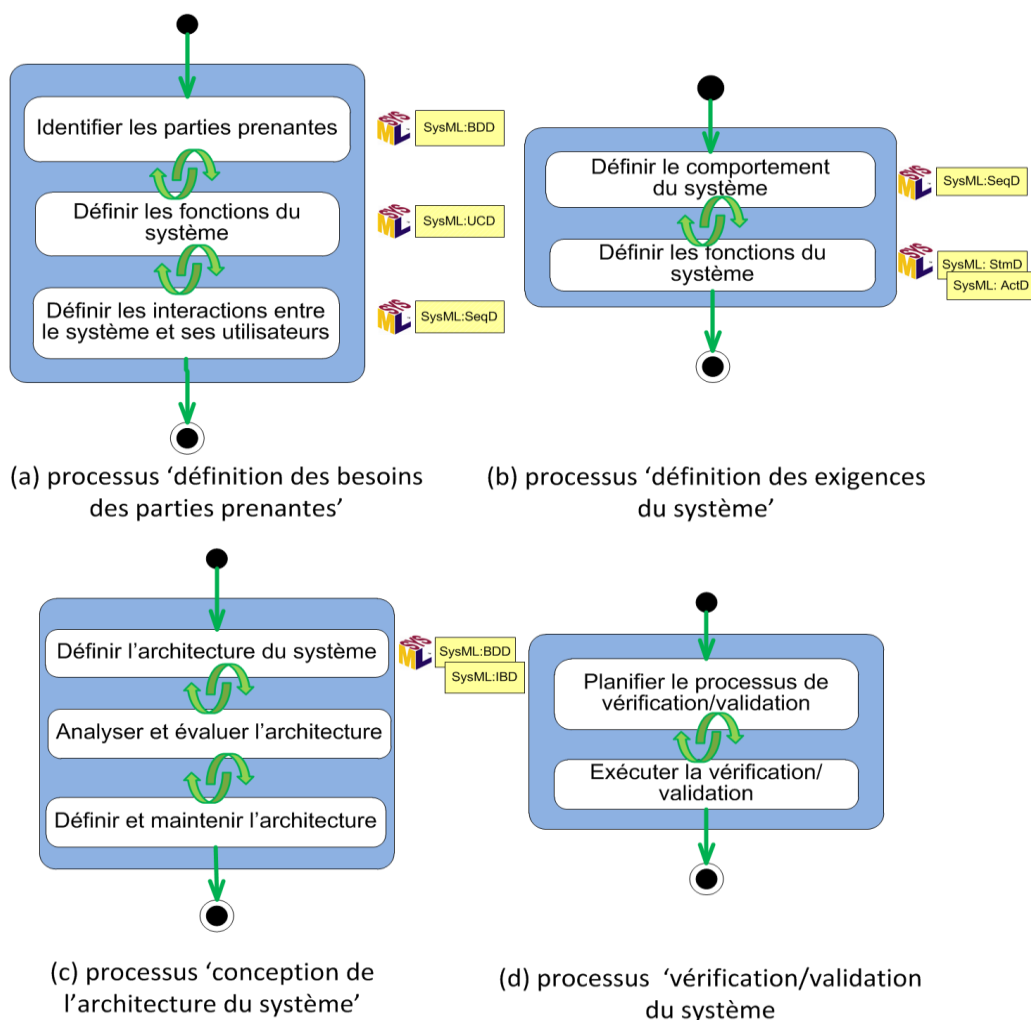


Figure 47 : les processus et les activités de la méthodologie de conception des RMS.

3.1. Définition des exigences des parties prenantes

L'objectif de ce processus est de définir les exigences du système qui peut fournir les services demandés par les utilisateurs et autres PPs dans un environnement bien défini. Il identifie les acteurs impliqués dans le système le long de son cycle de vie et leurs besoins. Il analyse et transforme ces besoins en un ensemble commun d'exigences qui expriment les interactions prévues entre le système et son environnement opérationnel, ces exigences constitueront le référentiel par rapport auquel chaque service opérationnel sera validé. L'exécution réussie de ce processus générera les résultats suivants :

- Le contexte d'utilisation des services est défini, les concepts opérationnels sont spécifiés, les caractéristiques requises sont identifiées.
- Le processus de fabrication de chaque produit est défini.
- Les familles de produits qui seront fabriqués par le système sont définies.
- Les DRs ainsi que le besoin des PPs en termes de reconfiguration du système sont définis.
- Les contraintes sur la solution sont définies.
- Les exigences des PPs sont définies.
- Les exigences des PPs pour la validation du système sont identifiées.
- La traçabilité est établie entre les exigences des PPs et leurs besoins.

3.1.1. Conception autour d'une famille de produits et changement de capacité

Un RMS est conçu autour d'une famille de produits avec suffisamment de flexibilité pour fabriquer tous ces produits. Le passage d'un produit à l'autre nécessite la reconfiguration du système. Les modifications à faire pour passer d'un produit à l'autre au sein de cette famille doivent être spécifiées pendant la phase du développement du système. C'est pour cela que dès la phase de définition des besoins des PPs, cette famille de produits autour de laquelle le système sera conçu doit être clairement définie. C'est à partir de cette famille de produits que la plupart des exigences de reconfiguration seront dérivées. Aussi, le système reconfigurable doit être conçu avec la possibilité de changer sa capacité de production facilement et graduellement. La conception d'un RMS sera facile si ces produits sont regroupés en familles selon leurs caractéristiques communes. Les différentes méthodes et techniques développées pour regrouper ces produits en familles peuvent être utilisées pendant ce processus (voir chapitre 1).

3.1.2. Exigences et contraintes de reconfiguration

Dans le cas de la conception d'un RMS, la question qui se pose est : comment les exigences et les contraintes de reconfiguration peuvent-elles être intégrées dans le processus de développement ? La définition de tous les DRs fait partie de la réponse à cette question. La modification du produit peut être considérée comme un DR. La modification du volume de production ou les pannes du système

pourraient aussi être considérées comme des DRs. Il n'existe aucun ensemble de DRs, standard et universel, commun à tous les RMSs, pour chaque cas, un ensemble de DRs devrait être défini lors du développement du système, une fois le système déployé et pendant sa phase opérationnelle, d'autres DRs pourraient être identifiés et ajoutés à la liste.

3.1.3. Contraintes de précedence de conception

Comme nous l'avons déjà dit auparavant (voir section 2.4), le sous-système à valeur ajoutée devrait être conçu en premier, après avoir regroupé les produits en familles. Les opérations du processus de fabrication devraient être attribuées aux futures cellules, machines ou stations du système. Le sous-système de stockage doit être défini pour répondre aux différents besoins du sous-système à valeur ajoutée. Par conséquent, les différentes zones de stockage qui seront nécessaires devraient être identifiées. Les fonctions du système de manutention ou du système logistique seront définies en fonction des flux matériels et informationnels entre les différents composants du sous-système à valeur ajoutée et ceux du sous-système de stockage.

3.1.4. Les activités et les tâches à réaliser

La Figure 47 (a) donne une vision globale des activités à réaliser du processus « définition des besoins des PPs ». Chaque activité du processus se décompose en plusieurs tâches qui doivent être exécutées avec les méthodes et les outils appropriés.

3.1.4.1. Identifier les parties prenantes du système

La première activité à effectuer est de définir les acteurs ou les PPs qui ont un intérêt dans le système le long de son cycle de vie. Cela inclut les : utilisateurs, opérateurs, développeurs, producteurs, opérateurs de maintenance, acquéreurs, fournisseurs, organisations de régulation (sécurité par exemple), etc. Pour un système manufacturier, l'acteur principal est l'utilisateur final qui utilisera ce système pour fabriquer les produits souhaités. Une PP peut être une personne physique (Opérateur, développeur, conducteur, etc.) ou une personne morale (organisation, un autre système, etc.). On peut utiliser le diagramme de contexte pour identifier et mettre en évidence les PPs du système. La Figure 48 est un exemple d'un diagramme de contexte d'un AGV, ce diagramme montre l'ensemble des PPs qui sont en interaction directe avec l'AGV dans son environnement opérationnel.

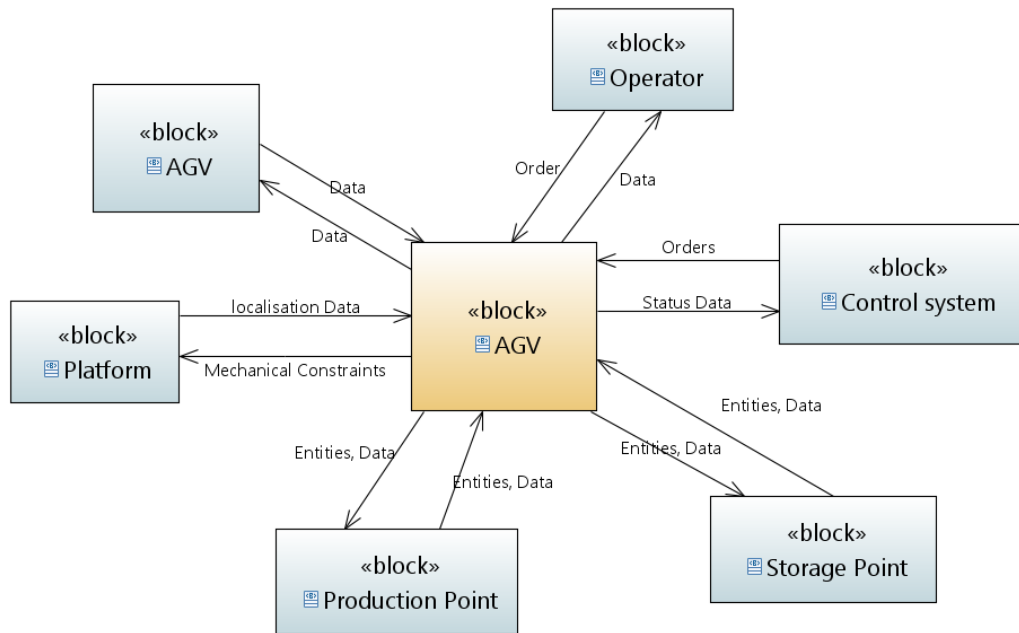


Figure 48 : exemple, diagramme de contexte d'un AGV.

3.1.4.2. Identifier les fonctions du système

Cette activité consiste à définir toutes les fonctions ou les services que le système doit fournir. Pour un système de production, les fonctions requises sont principalement les fonctions à valeur ajoutée comme les fonctions d'usinage, d'assemblage, ou toute autre fonction qui ajoute de la valeur à un produit le long de son processus de sa fabrication. Ces fonctions pourraient être identifiées à partir du processus de fabrication de chaque produit qui sera fabriqué par le système. L'ensemble des produits à réaliser ainsi que leurs processus de fabrication détaillés doivent être développés pendant cette activité. Pour identifier les services requis au système, on peut utiliser le diagramme des cas d'utilisation ou l'analyse fonctionnelle externe. La Figure 49 est un exemple de l'utilisation d'un «diagramme des cas d'utilisation» pour identifier et mettre en évidence l'ensemble des services demandés à un AGV.

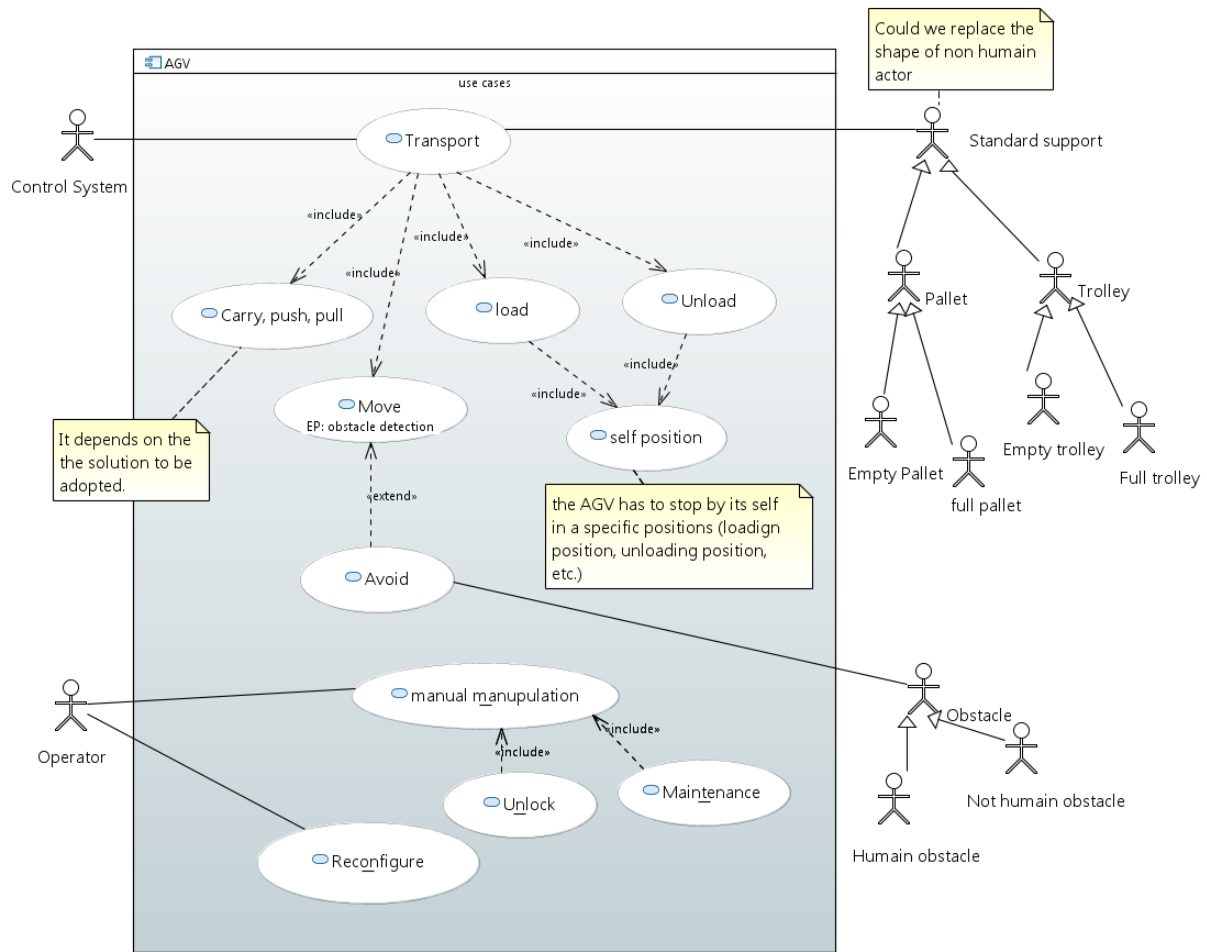


Figure 49 : exemple, diagramme cas d'utilisation d'un AGV.

Une fois les fonctions ou les services fournis par le système sont identifiés, on déduit les exigences des PPs. Elles sont exprimées sous forme textuelle ou graphique, elles se concentrent sur l'objectif du système et sur son comportement. Une simple énumération, généralement textuelle suffit, le but est d'identifier les besoins initiaux des PPs sans spécification. Après, on passe à la définition des contraintes imposées sur la solution future (le système), cela peut provenir d'une solution définie par les PPs, des décisions d'implémentation déjà fixées, obligations d'utiliser des moyens, des ressources, des agréments, des décisions de gestion, ou des décisions techniques.

3.1.4.3. Définir les interactions entre les utilisateurs et le système.

La troisième activité dans ce processus est de définir les interactions entre le système et ses utilisateurs. Dans le cas des systèmes de production, les opérateurs sont en interaction directe avec le système. Ils interagissent avec différentes ressources du système pour : manipuler les machines ou les stations de travail, gérer les stocks, conduire les moyens de transport, paramétrer le système de contrôle, etc. toutes ces interactions devraient être définies au cours de cette activité.

Chaque cas d'utilisation identifié auparavant est détaillé en un ou plusieurs scénarios opérationnels. Ces derniers sont utilisés pour analyser le fonctionnement du système dans un

environnement prévu afin d'identifier les exigences qui n'ont pas été formellement spécifiées par aucune des PPs. Pour détailler les scénarios opérationnels, on utilise généralement le diagramme de séquence, ce diagramme fait partie des diagrammes SysML, il permet de montrer les interactions entre les utilisateurs du système et le système. Ce diagramme peut être complété par le diagramme machine états pour détailler le fonctionnement interne du système.

Après la spécification des services rendus par le système, ses interactions avec les utilisateurs, son fonctionnement interne plus ou moins détaillé, il est nécessaire d'identifier les risques de sécurité, si cela est justifié, spécifier les fonctions de sécurité requises et les exigences de sécurité. Il faut identifier les fonctions qui pourraient impacter la sécurité des opérateurs du système.

Une fois toutes les exigences des PPs identifiées, il faut les analyser, cela inclut l'identification et la priorisation des exigences en conflit, manquantes, incomplètes, ambiguës, incohérentes, invérifiables, etc. Classiquement, l'analyse des exigences se fait à l'aide des outils simples comme les outils de traitement de texte ou les tableurs. Avec l'augmentation de la complexité des systèmes, ces outils sont devenus obsolètes vu le nombre important d'exigences à analyser. Mais, il existe des outils qui permettent une analyse rapide et efficace des exigences, ces outils sont communément appelés outils de gestion des exigences qui ont pour rôle la gestion (la collecte, l'analyse, l'échange, la validation, etc.) de l'ensemble des exigences d'un système le long de son cycle de vie. L'ensemble de ces exigences doit être validé par les différentes PPs pour pouvoir passer au prochain processus « l'analyse des exigences système ».

Après la définition et la validation de l'ensemble des exigences des PPs, les liens de traçabilité doivent être créés entre les PPs et leurs exigences. Ces exigences vont être aussi liées aux exigences système, aux tests de validation, et à d'autres entités (exigences, PPs, composante du système, test de vérification, test de validation, etc.) du cycle de développement d'un système. Les liens de traçabilité entre ces différentes entités facilitent la gestion du développement du système ainsi que la maîtrise de sa complexité. Pour pouvoir créer et gérer ces liens de traçabilité, il faut un outil spécial puisque les outils classiques (les outils de traitement de texte ou les tableurs) ne permettent pas d'atteindre ces objectifs. Les outils de gestion des exigences permettent de créer, personnaliser, et gérer les liens de traçabilité qui peuvent avoir lieu entre les différentes entités du cycle de développement du système.

3.1.5. Les liens de traçabilité

Les liens de traçabilités possibles dans le processus « définition des exigences des PPs » sont :

3.1.5.1. Entre les PPs et les exigences

Les PPs expriment des exigences, voici quelques règles de traçabilité :

- Une PP peut exprimer une ou plusieurs exigences.

- Une exigence peut être exprimée par une ou plusieurs PPs.
- Une exigence doit être exprimée par au moins une PP, si ce n'est pas le cas, cette exigence n'aura aucune utilité et elle doit être supprimée.

Le Tableau 5 est un exemple d'une matrice de traçabilité entre les PPs et les leurs exigences, ce genre d'outils est très utilisé en IS pour l'analyse et aussi pour gérer les différents liens de traçabilité. Par exemple, grâce au tableau suivant, on peut mettre en évidence, et d'une manière lisible, tous les liens de traçabilité qui existent entre les PPs et leurs exigences. On peut aussi écarter toutes les exigences qui n'ont pas été exprimées par aucune des PPs.

		Identifiants des parties prenantes					
		PP_01	PP_02	PP_03	PP_04	PP_05	PP_N
Identifiants exigences	Exg_01	X			X	X	
	Exg_02	X		X			
	Exg_03		X	X	X		
	Exg_04		X	X		X	X
	Exg_05			X	X		
	Exg_06		X			X	
	Exg_07	X			X	X	X
	Exg_08	X	X	X			X
	Exg_09		X			X	X
	Exg_10	X			X		
	Exg_11			X			
	Exg_N	X	X				X

Tableau 5 : exemple d'une matrice de traçabilité entre les PPs et les exigences.

3.1.5.2. Entre les exigences des PPs

Il y a plusieurs liens de traçabilité entre les exigences. Par exemple, une exigence peut être dérivée d'une autre exigence, une exigence peut être raffinée (rajouter des spécifications propres à l'exigence) par une ou plusieurs exigences, etc. Voici quelques règles de traçabilité entre les exigences.

- Une exigence ne peut pas être dérivée d'elle-même,
- Une exigence ne peut pas être raffinée par elle-même,

La Figure 50 montre les liens de traçabilité possibles entre les exigences.

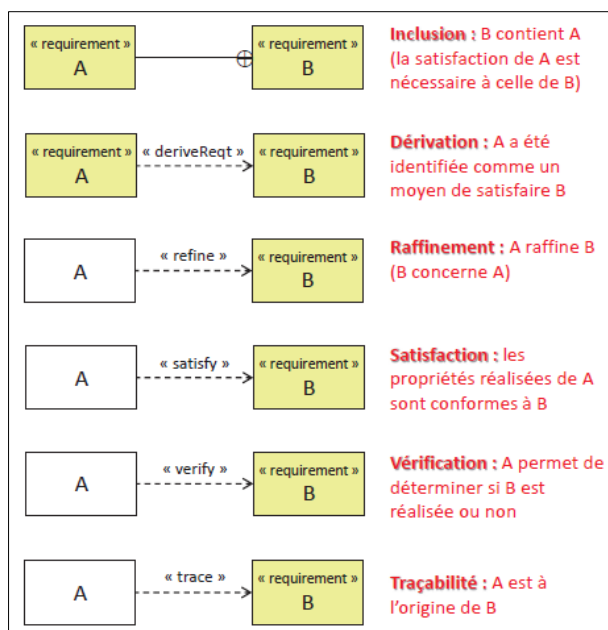


Figure 50 : les liens de traçabilité possibles entre exigences (Gendre & Virely 2013)

Le Tableau 6 est un exemple d'une matrice de traçabilité entre les exigences (sur ce tableau, la lettre « D » signifie dérivée, et la lettre « R » signifie raffinée).

		Identifiant exigences															
		Exg_01	Exg_02	Exg_03	Exg_04	Exg_05	Exg_06	Exg_07	Exg_08	Exg_09	Exg_10	Exg_11	Exg_12	Exg_13	Exg_14	Exg_N	
Identifiants exigences	Exg_01													D	D		
	Exg_02	D				D		R				D	D				
	Exg_03	D				D	D		R	R				R	R		
	Exg_04		D				D				R						
	Exg_05		D							D							
	Exg_06			D						D	R		R	R			
	Exg_07	R		D		R											
	Exg_08			D	R	R							D	R			
	Exg_09																
	Exg_10	R	R													D	D
	Exg_11			D			D								R		
	Exg_12		R				D										
	Exg_13				D	D											
	Exg_14									D							
	Exg_N				D	D			D								

Tableau 6 : exemple d'une matrice de traçabilité entre les exigences.

3.2. Analyse du système

L'objectif de ce processus est de transformer les services demandés au système qui sont représentés sous forme d'exigences des PP en exigences techniques du système qui peut fournir ces services. Ce processus construit une représentation du futur système qui répondra aux exigences des PP et qui n'impliquera aucune implémentation spécifique tant les contraintes le permettent. Le

processus crée un ensemble d'exigences système mesurables qui précisent quelles caractéristiques, attributs et exigences fonctionnelles et de performance que le système doit avoir, afin de satisfaire les exigences des PPs. L'exécution réussie de ce processus générera les résultats suivants :

- Les caractéristiques requises, les attributs, les exigences fonctionnelles, et les performances du système sont spécifiés.
- Les contraintes qui affecteront la conception de l'architecture du système sont spécifiées.
- La traçabilité est établie entre les exigences du système et les exigences des PPs.
- Les exigences pour la vérification du système sont identifiées.

3.2.1. Les activités et les tâches à réaliser

La Figure 47 (b) donne une vision globale des activités à réaliser du processus « Analyse des exigences ». Chaque activité se décompose en plusieurs tâches qui doivent être exécutées avec les méthodes et les outils appropriés.

3.2.1.1. Définir le comportement du système

La première tâche de cette activité est de définir le comportement du système, cela inclut l'analyse et la description des interactions requises entre le système et son environnement opérationnel. Le diagramme de séquence peut être utilisé pour montrer les interactions du système avec son environnement. Puis, il faut spécifier les exigences et les fonctions du système. Aussi, il faut analyser et définir les contraintes d'implémentation ou de réalisation nécessaires qui sont introduites par les exigences des PPs ou qui sont des limites incontournables de la solution. Les contraintes de sécurité, de santé, d'environnement, de durabilité, etc. en tenant compte des normes et de la législation doivent être aussi analysées et spécifiées. Il faut aussi définir les techniques et les paramètres pour mesurer la qualité des indicateurs de performances KPIs ; ce qui revient à définir les paramètres de performance critiques associés à chaque mesure d'efficacité identifiée dans les exigences des PPs.

3.2.1.2. Définir les fonctions internes du système.

Définir comment le système et ses opérateurs vont réaliser chaque fonction externe afin d'en déduire ses fonctions internes. Par exemple, les diagrammes SysML (diagramme machine à états, diagramme d'activités) peuvent être utilisés conjointement pour détailler le fonctionnement interne du système.

L'ensemble des exigences du système doit être analysé afin de valider l'unicité, la complétude, la non-ambiguïté, la cohérence, la possibilité d'implémentation de chaque exigence. Cette analyse permettrait d'identifier le manque, les conflits entre exigences afin de les résoudre. À la fin, il est nécessaire de valider avec les PPs que l'ensemble des exigences spécifiées du système reflète bien leurs besoins et exigences, et qu'il est suffisant pour les processus suivants (processus de conception

de l'architecture, processus d'implémentation et de réalisation, etc.). Puis, il faut assurer la traçabilité mutuelle entre les exigences système et les exigences des PPs. c.-à-d. que chaque exigence d'une PP est atteignable ou raffinée par une ou plusieurs exigences système.

3.2.2. Les liens de traçabilités

Les liens de traçabilité possibles dans le processus « Analyse des exigences » sont entre les exigences des PPs et les exigences système ou entre les exigences système elles-mêmes. Les exigences sont les exigences générées pendant le processus « Analyse des exigences ». Une exigence exprimée par une PP (partie prenante) pourrait être raffinée par une ou plusieurs exigences système. Une ou plusieurs exigences système pourraient être dérivées d'une exigence exprimée par une PP. Voici quelques règles de traçabilité entre les exigences.

- Une exigence ne peut pas être dérivée d'elle-même,
- Une exigence ne peut pas être raffinée par elle-même,

La Figure 50 montre les liens de traçabilité possibles entre les exigences. Le Tableau 6 est un exemple d'une matrice de traçabilité entre les exigences (sur ce tableau, la lettre « D » signifie dérivée, et la lettre « R » signifie raffinée).

3.3. La conception de l'architecture du système

L'objectif de ce processus est de générer des alternatives de l'architecture du système, de sélectionner une ou plusieurs alternatives qui encadrent les préoccupations des PPs et répondent aux exigences du système et à exprimer cela dans un ensemble de vues cohérentes. L'itération du processus de définition de l'architecture avec le processus de définition des exigences du système et avec d'autres processus est souvent utilisée afin qu'il y ait une compréhension négociée du problème à résoudre et qu'une solution satisfaisante soit identifiée (ISO/IEC/IEEE15288 2015). L'exécution réussie de ce processus générera les résultats suivants :

- L'architecture du système est définie.
- Les modules du système sont identifiés et conçus.
- Les configurations et les processus de reconfiguration de chaque module sont définis.
- Les exigences d'interfaces (internes et externes) sont définies.
- Les liens entre les DRs du système et ses configurations sont définis.
- Les bases pour l'intégration et la vérification des éléments du système sont définies.
- La traçabilité de la conception de l'architecture aux exigences système est établie.

3.3.1. La conception autour d'une famille de produits

Les configurations du système doivent être liées à ses contextes. Par exemple, chaque produit doit être lié à une configuration du système qui permet sa fabrication. Ces paramètres doivent être décrits

en détail (quelles sont les ressources à utiliser, quels sont leurs paramètres, quels sont les outils à utiliser, etc.). Tous ces liens devraient être établis au cours de ce processus de conception. Passer d'une configuration à l'autre nécessite la reconfiguration du système. Chaque processus de reconfiguration devrait être suffisamment détaillé pour être facilement réalisé par les opérateurs du système. Par conséquent, toutes les modifications nécessaires doivent être décrites ainsi que les moyens (outils, documents, etc.) qui doivent être utilisés. Le Tableau 7 est un exemple qui montre les liens entre les changements de produits et les différents processus de reconfiguration.

	Produit A	Produit B	Produit C	Produit D
Produit A		Reconfig A-B	Reconfig A-C	Reconfig A-D
Produit B	Reconfig B-A		Reconfig B-C	Reconfig B-D
Produit C	Reconfig C-A	Reconfig C-B		Reconfig C-D
Produit D	Reconfig D-A	Reconfig D-B	Reconfig D-C	

Tableau 7 : Liens entre le changement de produits et les reconfigurations du système.

3.3.2. La modularité

Un RMS possède certaines caractéristiques ou fonctionnalités clés qui lui permettent un haut niveau de réactivité aux changements. Ces fonctionnalités sont : la personnalisation, la convertibilité, l'évolutivité, la modularité, l'intégration et la diagnosticabilité (Koren & Shpitalni 2010). Ces caractéristiques devraient être intégrées dans le système dès les premières phases du processus de conception et dans l'intégralité du système afin d'assurer un degré élevé de reconfigurabilité. Donc, au cours du processus de définition de l'architecture du système, la modularité, une caractéristique importante d'un RMS, doit être prise en compte. L'objectif est de réaliser une architecture modulaire où les modules de base, physiques/logiques, sont clairement définis ainsi que les interfaces entre ces modules.

Donc il est important de disposer des méthodes ou des techniques qui facilitent le développement des architectures modulaires pour les systèmes manufacturiers. Une méthode de conception de systèmes modulaires a été proposée dans (Lameche et al. 2017a). Cette méthode peut être intégrée dans la méthodologie globale proposée pour la conception des RMSs (voir chapitre 3). L'objectif de la méthode proposée est de concevoir un ensemble de modules standards qui pourraient être utilisés pour former différentes configurations du système pour s'adapter facilement et de manière rentable au contexte changeant du système.

Quelle que soit la méthode ou la technique utilisée pour obtenir une architecture modulaire ; certains principes doivent être pris en compte lors de la conception d'un système modulaire. Ces principes sont :

- Allocation des fonctions : pendant le processus de développement, une fois que les fonctions du système sont définies, elles doivent être attribuées aux modules du système. Les fonctions ne doivent pas être partagées entre les modules du système.

- **Séparer les modules** : pour définir clairement les modules et les séparer, nous pouvons poser ces questions : l'architecture identifie-t-elle clairement les modules ? Est-ce que chaque module a un but clairement défini ? Les modules sont-ils subdivisés autant que possible ? Les modules ne reproduisent pas les mêmes fonctionnalités ?
- **Fournir des interfaces simples** : les interfaces simples réduisent le nombre d'interactions qui doivent être prises en considération lors de la vérification du fonctionnement du système. Les interfaces simples facilitent également la réutilisation des composants dans des circonstances différentes.

3.3.3. La simulation

La simulation est une étape essentielle de la méthodologie proposée, elle peut être utilisée pour évaluer et comparer les différentes solutions pour un problème particulier pendant la phase de conception du système. Elle peut également être utilisée pendant la phase opérationnelle du système pour soutenir la sélection des futures configurations du système. Lorsque la reconfiguration du système est nécessaire et que la configuration cible du système n'est pas définie, la simulation pourrait être utilisée pour évaluer et comparer les configurations possibles du système pour sélectionner la meilleure (voir chapitre 5).

Dans (Lameche et al. 2016), en parallèle avec le développement du système, des modules virtuels ont été développés, chacun de ces modules représente le futur module réel correspondant du système. Le développement de ces modules virtuels a énormément contribué à concevoir les réels modules du système. En outre, les modules virtuels développés ont été regroupés pour former une bibliothèque virtuelle qui a été utilisée plus tard pendant la phase opérationnelle du système pour créer des modèles virtuels pour différentes configurations du système. Cela a aidé à comparer ces configurations et à sélectionner la meilleure pour chaque contexte de production.

3.3.4. Les activités et les tâches à réaliser

La Figure 47 (c) donne une vision globale des activités à réaliser du processus « conception de l'architecture du système ». Chaque activité du processus se décompose en plusieurs tâches qui doivent être exécutées avec les méthodes et les outils appropriés.

3.3.4.1. Définir l'architecture du système

Pour concevoir un système, souvent on commence par son architecture fonctionnelle, et puis on passe à son architecture physique. Les exigences définies dans les processus précédents doivent être attribuées aux différents éléments fonctionnels du système et ces fonctions seront affectées aux éléments physiques du système. Par conséquent, les liens de traçabilité doivent être définis entre les exigences et les éléments d'architecture du système. Les données et les modèles de l'architecture du système constitueront la base pour établir le plan d'intégration (assemblage et test) du système.

Donc, en premier, il faut identifier et définir les flux entre les fonctions internes du système afin d'établir l'architecture fonctionnelle de ce dernier. Ces flux permettent d'identifier les interfaces entre les modules du système, et entre le système et les systèmes externes. Ces interfaces doivent être définies. Ensuite, il faut allouer les fonctions de l'architecture fonctionnelle aux éléments de l'architecture physique du système. Le choix de la technologie à utiliser pour chaque fonction du système et les interfaces entre elles sera fait en dernier. Le diagramme de définition interne du bloc peut être utilisé pour mettre en évidence les interfaces du système.

3.3.4.2. Analyser et évaluer l'architecture

Analyser l'architecture résultante pour établir les critères (caractéristiques physiques, de performance, de comportement, de durabilité, etc.) de conception de chaque élément. Évaluer les solutions alternatives de conception, les modéliser à un niveau de détails qui permet de les comparer par rapport aux spécifications exprimées par les exigences système, et par rapport aux performances, coûts, temps, et risques exprimés dans les exigences des PPs. Pour cela, on doit évaluer l'efficacité de la solution choisie en termes de coût, de risques pour une conception faisable, efficace, stable, et optimisée. Vérifier la disponibilité des éléments, physiques et logiciels, choisis. Cela concerne, entre autres, l'évaluation de la conception des éléments qui ne sont pas disponibles et l'estimation des coûts, des plannings, et des risques techniques afin de déterminer si un élément doit être développé, réutilisé, adapté ou sous-traité.

3.3.4.3. Documenter et maintenir l'architecture

Dans cette tâche, on enregistre le partitionnement structural et fonctionnel, les interfaces et les décisions et les conclusions de conception du système. La conception de l'architecture permet la révision en cas de changement au cours du cycle de vie du système ainsi que de fournir des informations pour toute autre réutilisation de l'architecture et c'est aussi la source d'information pour définir les tests d'intégration.

Les données de conception du système (l'architecture du système) qui sont générées pendant le processus de conception et qui sont stockées dans le dépôt des données sous différentes formes, mais principalement sous forme de plans et de modèles seront utilisées pour construire les éléments du système et les assembler. Elles représentent aussi un référentiel de vérification du système. Il faut établir et maintenir la traçabilité mutuelle entre la conception spécifiée et les exigences système.

3.3.5. Liens de traçabilité

Les liens de traçabilité possibles dans le processus « Conception de l'architecture » sont entre les exigences et les fonctions du système. Une ou plusieurs exigences pourraient être allouées à une fonction du système. Une fonction du système pourrait implémenter une ou plusieurs exigences. Voici quelques règles de traçabilité entre les exigences et les fonctions du système.

- Une exigence ne peut pas être allouée à plusieurs fonctions du système.
- Une fonction doit avoir au moins une exigence qui lui est allouée, sinon elle n’aura aucune utilité.

Le tableau suivant donne un exemple d’une matrice de traçabilité entre les exigences et les fonctions d’un système (la lettre «A» sur le tableau signifie Allocation), par exemple l’exigence «Exg_01» est allouée à la fonction «Fct_02» :

		Identifiants exigences														
		Exg_01	Exg_02	Exg_03	Exg_04	Exg_05	Exg_06	Exg_07	Exg_08	Exg_09	Exg_10	Exg_11	Exg_12	Exg_13	Exg_14	Exg_N
Identifiants fonctions système	Fct_01			A	A											
	Fct_02	A	A													
	Fct_03					A										
	Fct_04						A									
	Fct_05							A								
	Fct_06								A	A						
	Fct_07										A					
	Fct_08															A
	Fct_09														A	
	Fct_10													A		
	Fct_11											A	A			

Tableau 8 : exemple d’une matrice de traçabilité entre les exigences et les fonctions système.

3.3.5.1. Enter les fonctions du système et les composants du système

Une ou plusieurs fonctions du système pourraient être allouées à un élément ou composant du système. Un élément ou composant du système pourrait implémenter une ou plusieurs fonctions du système. Voici quelques règles de traçabilité à respecter :

- Une fonction ne peut pas être allouée à plusieurs éléments du système.
- Un composant ou un élément du système doit avoir au moins une fonction qui lui est allouée, sinon il n’aura aucune utilité.

Le tableau suivant donne un exemple d’une matrice de traçabilité entre les fonctions d’un système et ses composants (la lettre «A» sur le tableau signifie Allocation). Par exemple, les fonctions «Fct_01» et «Fct_02» sont allouées au composant «Cmp_02».

		Identifiants Fonctions système									
		Fct_01	Fct_02	Fct_03	Fct_04	Fct_05	Fct_07	Fct_08	Fct_09	Fct_10	Fct_11
Identifiants fonctions système	Cmp_01			A	A						
	Cmp_02	A	A								
	Cmp_03					A					
	Cmp_04						A				
	Cmp_05							A			
	Cmp_06								A	A	
	Cmp_07										A
	Cmp_08										
	Cmp_09										
	Cmp_10										
	Cmp_11										

Tableau 9 : exemple d'une matrice de traçabilité entre les fonctions et les composants d'un système.

3.4. La vérification du système

Après la réalisation du système, il doit être vérifié et validé avant de commencer la production.

3.4.1. Processus de vérification

Dans la norme ISO/IEC/IEEE-15288, le processus de vérification est décrit comme le processus qui fournit une preuve objective que le système ou ses éléments/modules remplissent leurs exigences et leurs caractéristiques spécifiées. Le but de ce processus est d'identifier les anomalies (erreurs ou défauts) dans le système en utilisant des méthodes, des techniques, des normes ou des règles appropriées. Le processus de vérification fournit aussi les informations nécessaires pour déterminer la résolution des anomalies identifiées. Il sert à vérifier si le système a été correctement implémenté, c'est-à-dire que le système réalisé respecte l'architecture conçue et les spécifications imposées.

En ce qui concerne les RMSs, en plus des tests de vérification standards (tests de sécurité, essais opérationnels, tests de robustesse, etc.), des tests de reconfigurabilité doivent être effectués sur le système ainsi que sur ses sous-systèmes et ses modules. Chaque partie du système doit être vérifiée dans ses différentes configurations. Dans chaque configuration du système, du sous-système ou du module, des tests de vérification standards doivent être effectués. En outre, les processus de reconfiguration doivent être vérifiés pour chaque partie du système ainsi que pour l'ensemble du système. Les résultats des tests de vérification seront analysés pour identifier les non-conformités du système avec les exigences système. L'exécution réussie de ce processus générera les résultats suivants :

- Une stratégie de vérification est définie et les contraintes de vérification sont identifiées.
- Les données fournissant des informations pour les actions correctives sont enregistrées.

- Les preuves que le système réalisé répond aux exigences système et aux exigences de conception de l'architecture sont fournies.

3.4.2. Les tâches et les activités à réaliser

La Figure 47 (d) donne une vision globale des activités à réaliser du processus de vérification du système. Chaque activité du processus se décompose en plusieurs tâches qui doivent être exécutées avec les méthodes et les outils appropriés.

3.4.2.1. Planifier la vérification

La vérification du système doit être planifiée avant d'être exécutée. Il faut s'assurer que les systèmes qui permettent la vérification sont disponibles et aussi les moyens, les équipements, et les opérateurs associés, et que les opérateurs sont préparés pour conduire la vérification. La vérification du système peut inclure ses opérateurs humains.

3.4.2.2. Exécuter la vérification

Conduire la vérification pour démontrer la conformité du système à ses exigences : les non-conformités identifient l'existence d'erreurs aléatoires ou d'erreurs de conception, et des actions de correction sont initiées en conséquence. Toutes données concernant les tests de vérification (les rapports de tests, les non-conformités, etc.) doivent être disponibles pour les différents acteurs du système comme les développeurs par exemple. Ces données sont analysées pour identifier et isoler les parties du système qui génèrent les non-conformités afin de diagnostiquer les sources de la non-conformité. La nature et l'objectif des actions de vérification (révision, audit, comparaison, test statique, test dynamique, démonstration, ou une combinaison, etc.) dépendent du modèle, du prototype, ou le système actuel, et les risques perçus.

3.4.3. Liens de traçabilité dans le processus « Vérification du système »

Les liens de traçabilité possibles dans le processus « Vérification du système » sont :

3.4.3.1. Entre les exigences et les tests de vérification

Une ou plusieurs exigences pourraient être vérifiées par un ou plusieurs tests de vérification. Un ou plusieurs tests de vérification pourraient vérifier une ou plusieurs exigences. Les exigences qui sont liées aux tests de vérifications sont souvent les exigences de spécification, voici quelques règles de traçabilité à respecter :

- Une exigence doit être vérifiée par au moins un test de vérification.
- Un test de vérification doit au moins vérifier une exigence, sinon il n'aura aucune utilité.

Le tableau suivant donne un exemple d'une matrice de traçabilité entre les exigences et les tests de vérification (les lettres « VR » sur le tableau signifient vérifier, et les lettres « TVr » signifient test de

vérification), par exemple le test de vérification «TVr_01» permet de vérifier les exigences «Exg_03», «Exg_04», «Exg_07» et «Exg_10».

		Identifiants exigences														
		Exg_01	Exg_02	Exg_03	Exg_04	Exg_05	Exg_06	Exg_07	Exg_08	Exg_09	Exg_10	Exg_11	Exg_12	Exg_13	Exg_14	Exg_N
Identifiants des tests de	TVr_01			VR	VR			VR			VR					
	TVr_02	VR	VR										VR			
	TVr_03					VR										
	TVr_04			VR			VR	VR				VR				
	TVr_05							VR								
	TVr_06								VR	VA					VR	
	TVr_07	VR			VR						VA					
	TVr_08															VA
	TVr_09			VR						VR					VA	
	TVr_10		VR				VR							VA		
	TVr_11								VR			VA	VA			

Tableau 10 : exemple d’une matrice de traçabilité entre les exigences et les tests de vérification.

3.4.4. Processus de validation

Si tous les tests de vérification sont passés, le système sera déployé et le processus de validation peut commencer. Le but de ce processus est de fournir une preuve objective que le système en cours d’utilisation satisfait son but prévu dans son environnement opérationnel prévu (ISO/IEC/IEEE15288 2015). Sur la base des besoins des PPs, un plan de validation est élaboré. La différence entre le processus de vérification et le processus de validation est que le premier détermine si le système est bien construit, mais le second détermine si le bon système est construit. L’implémentation réussie de ce processus devrait générer les résultats suivants :

- Une stratégie de validation est définie.
- La disponibilité des services requis par les PPs est confirmée.
- Les données de validation sont fournies.
- Les données capables de fournir des informations pour des actions correctives sont enregistrées.

3.4.5. Les tâches et activités à réaliser

La Figure 47 (d) donne une vision globale des activités à réaliser du processus « validation du système ». Chaque activité du processus se décompose en plusieurs tâches qui doivent être exécutées avec les méthodes et les outils appropriés.

3.4.5.1. Planifier la validation

Au début, en se basant sur les exigences des PPs, il faut préparer un plan de validation avant de l’exécuter. Il définit les étapes de validation, c.-à-d. différentes étapes opérationnelles, scénarios et

missions qui progressivement donnent plus de confiance dans la conformité du système. Les méthodes et les techniques nécessaires pour implémenter la validation doivent être spécifiées d'une manière claire. Les conditions, et les critères de conformité doivent être aussi définis.

La nature et la portée des actions de validation dépendent du : modèle, prototype, ou le système en cours de validation, des risques (nouveau, sécurité, techniques), les contraintes organisationnelles, et les exigences des PPs.

3.4.5.2. Exécuter la validation

Tout d'abord, il faut s'assurer que les opérateurs, les systèmes qui facilitent la validation et les moyens associés sont disponibles pour conduire la validation du système. Puis, conduire la validation pour démontrer la conformité des services fournis par le système c.-à-d., le système répond bien aux besoins des PPs. Isoler les parties du système qui engendrent des non-conformités. Analyser et enregistrer les données de validation selon les critères définis dans le plan de validation. Les données de validation sont analysées pour détecter les défauts, les erreurs de conception, etc. Le diagnostic des défauts est réalisé à un niveau de détail cohérent avec le coût des actions de corrections, en incluant les actions de revalidation pour la correction des défauts. Les méthodes de validation incluent le test, l'inspection, l'analyse et la démonstration. Ces méthodes sont déterminées par les risques perçus, la sécurité et la criticité de l'élément considéré (Haskins et al. 2007).

3.4.6. Liens de traçabilité dans le processus « validation du système »

Les liens de traçabilité possibles dans le processus « Validation du système » sont :

3.4.6.1. Entre les exigences et les tests de validation

Une ou plusieurs exigences pourraient être validées par un ou plusieurs tests de validation. Un ou plusieurs tests de validation pourraient valider une ou plusieurs exigences. Les exigences qui sont liées aux tests de validation sont souvent les exigences fonctionnelles exprimées par les PPs acquéreurs du système ou par les PPs qui sont en interaction directe avec le système dans son environnement opérationnel, comme on l'a dit auparavant, le but est de valider le système dans son environnement opérationnel, c.-à-d. valider le fait que le système fournit les services demandés dans son environnement opérationnel avec les performances et les contraintes imposées. Voici quelques règles de traçabilité à respecter :

- Une exigence doit être validée par au moins un test de validation.
- Un test de validation doit au moins valider une exigence, sinon il n'aura aucune utilité.

Le tableau suivant donne un exemple d'une matrice de traçabilité entre les exigences et les tests de validation (les lettres "VL" sur le tableau signifient valider, et les lettres "TVI" signifie test de validation) :

		Identifiants exigences														
		Exg_01	Exg_02	Exg_03	Exg_04	Exg_05	Exg_06	Exg_07	Exg_08	Exg_09	Exg_10	Exg_11	Exg_12	Exg_13	Exg_14	Exg_N
Identifiants des tests de	TV1_01			VL	VL			VL			VL					
	TV1_02	VL	VL										VL			
	TV1_03					VL										
	TV1_04			VL			VR		VL				VL			
	TV1_05							VL								
	TV1_06								VL	VL					VL	
	TV1_07	VL			VR						VL					
	TV1_08															VL
	TV1_09			VL						VL					VL	
	TV1_10		VL				VR							VL		
	TV1_11								VL				VL			

Tableau 11 : exemple d'une matrice de traçabilité entre les exigences et les tests de validation.

4. Conclusion

En raison de leur structure changeante (physique et logique) à différents niveaux de composition, les RMSs sont des systèmes complexes et leur conception n'est pas une tâche facile. Par conséquent, la plupart des méthodes et techniques, utilisées pour concevoir les systèmes complexes, pourraient être appliquées aux RMSs avec certaines modifications, adaptations et améliorations. Ainsi, sur la base de cette idée, une méthodologie globale basée sur la norme ISO/IEC/IEEE-15288 pour aider à concevoir les RMSs a été proposée dans ce chapitre. Cette méthodologie aide les développeurs dans leurs travaux de la définition des besoins des PPs jusqu'au déploiement et la validation du système qui répond à ces besoins. Elle met en avant les évolutions nécessaires de certains concepts de l'ingénierie des systèmes pour prendre en compte la spécificité des systèmes reconfigurables.

La méthodologie proposée implémente plusieurs processus techniques de la norme qui sont adaptés au domaine des systèmes reconfigurables. Elle adopte une version modifiée du cycle en « V ». À chaque fois que la reconfiguration du système est nécessaire, une demande de reconfiguration est générée. Cette demande déclenchera la reprise de la méthodologie pour concevoir une nouvelle configuration du système qui répond aux exigences du nouveau contexte. La méthodologie nécessite l'intégration d'autres activités et tâches spécifiques au domaine des RMSs. En outre, ils manquent des techniques, des méthodes et des outils pour supporter les différentes activités déjà implémentées. L'objectif ultime est de développer une méthodologie basée sur principe de l'ingénierie des systèmes où les différentes tâches et activités sont réalisées en utilisant principalement des modèles.

La simulation a été brièvement présentée comme un outil puissant qui peut aider au développement des RMSs, dans le chapitre suivant, on détaille comment cet outil a été utilisé pour supporter la conception et la gestion des systèmes reconfigurables.

Chapitre 05 : la simulation pour la conception et la gestion des Systèmes Manufacturiers Reconfigurables

1. Introduction

Un système manufacturier reconfigurable RMS est conçu dès le départ pour des modifications rapides de sa structure, ainsi que de ses composants matériels et logiciels (physiques et logiques), afin d'ajuster rapidement sa capacité et sa fonctionnalité de production autour d'une famille de produits en réponse aux changements soudains du marché, ou aux changements intrinsèques du système (c.-à-d. les pannes). Donc un RMS est un système réactif dont la capacité de production est réglable afin de faire face aux fluctuations de la demande et dont la fonctionnalité est adaptable à de nouveaux produits (Koren 2005) (Koren & Shpitalni 2010) (Mehrabi et al. 2002), (Mehrabi et al. 2000a).

Le choix d'une configuration adaptée pour faire face aux changements et aux incertitudes de l'environnement de production est difficile. La simulation peut aider à déterminer la meilleure stratégie pour atteindre l'objectif assigné par l'augmentation brusque de la demande, le changement de produit ou une panne d'un composant du système. En effet, la simulation est l'activation d'un modèle, représentant le système réel, et qui est capable de reproduire son fonctionnement dans le temps, afin de connaître son comportement dynamique et de prédire son comportement futur. Elle est généralement, utilisée pour évaluer et comparer des scénarios définis au préalable et permet également de prédire les conséquences de modification d'un ou plusieurs composants du système ainsi que le choix d'une stratégie de pilotage plutôt qu'une autre.

La simulation de flux est très utilisée pour la conception des systèmes dédiés où les lignes de production et les machines sont dédiées à produire ou à fabriquer des produits spécifiques avec une haute cadence, et les systèmes flexibles où les lignes de production et les machines sont capables de produire une large variété de produits à une faible cadence. Mais pour le nouveau concept de système manufacturier reconfigurable, elle est peu utilisée. La définition de son rôle dans la conception et le fonctionnement de ce système n'ont pas été étudiés dans la littérature. Le but de ce chapitre est de mettre en évidence au travers d'un cas d'étude réel, l'apport de la simulation de flux dans la conception et la gestion des RMSs.

Dans ce chapitre, on montera comment la simulation peut être utilisée pour supporter la conception des RMSs, on proposera une méthodologie qui guidera tout projet la simulation d'un RMS. Un cas d'étude, sur lequel la méthodologie proposée est testée et validée, est présenté.

2. La simulation des flux

2.1. Définition

La simulation intègre à la fois la construction d'un modèle et l'utilisation expérimentale de ce modèle pour étudier un problème. Le modèle consiste en une représentation d'un système réel, capable de reproduire son fonctionnement. La simulation des flux est principalement utilisée pour étudier les flux physiques (pièces, matières, outils, etc.) et informationnels (Ordres de Fabrications, Kanban, etc.) dans l'atelier et les disponibilités des ressources (opérateurs, machines, convoyeurs, etc.). Elle ne permet pas de trouver directement et de façon optimale des solutions à des problèmes de production. On l'utilise généralement pour évaluer et comparer des scénarios possibles. Ses capacités d'imitation et de prédiction permettent d'obtenir des renseignements sur les conséquences de changements ou de modifications dans l'atelier (au niveau physique ou décisionnel), avant que ceux-ci ne soient effectués.

La simulation est un processus qui consiste à : concevoir un modèle du système étudié, mener des expérimentations sur ce modèle, interpréter les observations fournies par le déroulement du modèle et formuler des décisions relatives au système. Le but peut être de comprendre le comportement dynamique du système, de comparer des configurations, d'évaluer différentes stratégies de pilotage, d'évaluer et d'optimiser les performances du système (Lameche et al. 2016).

2.2. Les points forts de la simulation

On peut s'appuyer sur la simulation pour traiter des problèmes d'ordonnancement, pour aider à la maintenance des systèmes de production, pour choisir des indicateurs ou pour mesurer les effets des pannes sur le fonctionnement global du système. La simulation peut aussi fournir des estimations réalistes du comportement attendu du système. Elle peut aussi aider à évaluer les effets des actions suivantes : (ajouter, déplacer ou supprimer des machines, modifier les flux, modifier les durées de processus et/ou de montage, introduire de nouveaux produits ou supprimer des produits, modifier les systèmes de manutention, modifier l'ordonnancement et la répartition des tâches, ajouter ou supprimer du personnel ...etc.).

2.3. Les limites de la simulation

- Une fois le modèle du système est programmé et validé, la simulation fonctionne comme une boîte noire en fonction d'un scénario de fonctionnement. Elle ne fait donc que reproduire le comportement du système modélisé. Il est indispensable de vérifier la validité des données utilisées par la simulation pour éviter d'obtenir des résultats qui ne correspondent pas à ceux obtenus sur le système réel. Cela est particulièrement critique quand le système simulé est d'un niveau de complexité ne permettant pas une vérification analytique des résultats fournis par la simulation.

- La qualité des résultats de la simulation dépend exclusivement de la qualité des règles de fonctionnement et des données introduites lors de la modélisation.
- Les outils de simulation permettent toujours d'obtenir un résultat, mais n'apportent rien quant à sa validité par rapport au système réel que l'on simule.
- La simulation ne peut pas résoudre des problèmes, mais seulement fournir des indications à partir desquelles des solutions peuvent être déduites. Elle ne peut pas optimiser les performances du système.

3. Rôle de la simulation dans la conception des RMSs

Le potentiel de la simulation est très vaste, car elle est applicable à tous les flux de l'industrie et même des services, à tous les niveaux hiérarchiques et à toutes les phases du cycle de vie d'un système de production. Le plus souvent, le logiciel de simulation est utilisé en phase de conception ou d'amélioration des ateliers pour dimensionner les capacités des stocks et des files d'attente, pour tester des règles de fonctionnement, pour identifier les goulots d'étranglement, pour mesurer l'influence des perturbations, etc. La simulation de flux peut également être utilisée en phase d'exploitation, en complément d'outils de planification ou d'ordonnancement pour, par exemple, estimer des délais. Elle permet aussi de déterminer les valeurs optimales des paramètres de pilotage à appliquer au système de production avant de lancer l'exécution en couplant au modèle de simulation un algorithme d'optimisation (Najid & Kouiss 2007).

Pendant la phase de conception, la simulation peut aider à identifier et définir les modules de base qui composent un système manufacturier et en particulier un RMS. Un RMS est conçu pour être reconfigurable, c.-à-d. en cas de changement de contexte, le système doit changer de configuration pour répondre au mieux aux nouvelles exigences. Mais avant de passer à la nouvelle configuration, cette dernière doit être vérifiée si elle répond aux exigences du nouveau contexte de production avant de la déployer réellement sur le terrain. On pourrait avoir plusieurs configurations qui répondent toutes aux exigences du nouveau contexte de production, dans ce cas, la simulation peut nous aider aussi à sélectionner la meilleure configuration. Par conséquent, la simulation peut être utilisée pour tester et valider les différentes configurations du système, et de choisir celle qui répond au mieux aux critères de choix (satisfaction des exigences du nouveau contexte de production, temps et effort nécessaires pour la reconfiguration, les performances, etc.). Cela suppose d'avoir un modèle suffisamment fiable du système pour prendre des décisions correctes.

3.1. Positionnement de la simulation dans la méthodologie globale

Dans la méthodologie globale qu'on propose pour supporter la conception des RMSs (voir chapitre 04), la simulation constitue une étape essentielle. Elle peut être utilisée pendant la phase de

conception du système ainsi que pendant la phase opérationnelle. La Figure 51 illustre le positionnement de la simulation au niveau de la méthodologie proposée.

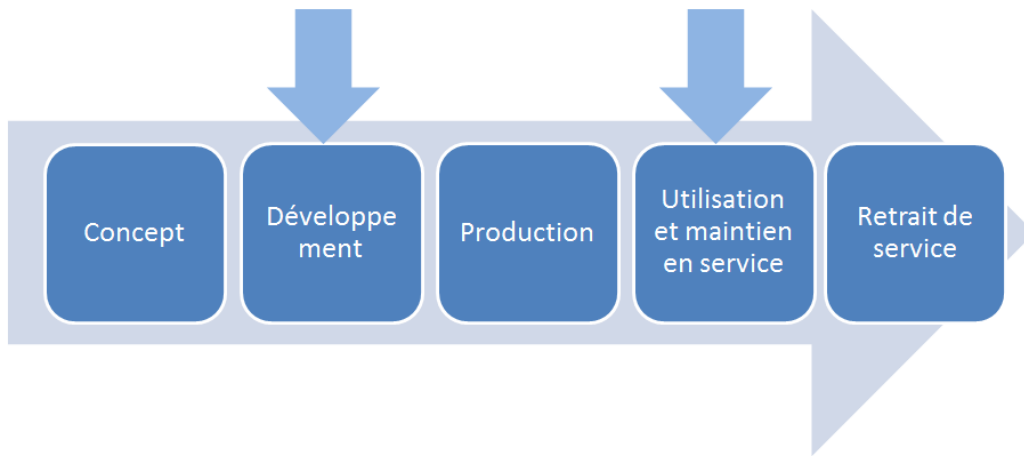


Figure 51 : positionnement de la simulation dans la méthodologie globale de conception d'un RMS.

3.1.1. Utilisation de la simulation pendant la phase de développement du système

La simulation peut être utilisée pendant la phase de développement du système pour évaluer et valider les différentes solutions possibles. Par exemple, évaluer les différentes stratégies de gestion (des stocks, des moyens de production, des moyens de transport, etc.). Elle peut aussi aider à identifier et à concevoir les éléments récurrents du système, c'est ce qu'on appelle les modules ou les éléments standards, à définir les interfaces entre ces modules, leurs paramètres, et leurs comportements.

3.1.2. Utilisation de la simulation pendant la phase opérationnelle du système

Dans la même méthodologie, il est supposé qu'une fois le système conçu, déployé, et opérationnel, la personne qui validera les configurations futures du système en cas de besoin de reconfiguration n'aura pas nécessairement des connaissances approfondies dans le domaine de la simulation des flux. Cette exigence a eu pour conséquence, la conception d'un outil simple d'utilisation pour simplifier la phase de test et de validation des différentes configurations possibles à l'aide de la simulation. L'idée est qu'en parallèle au développement des modules de base du système, on développe les modules virtuels qui leur correspondent, ces modules seront regroupés pour mettre à la disposition du configurateur une bibliothèque virtuelle qui lui facilite la construction des modèles pour les différentes configurations possibles du système afin de les simuler et de récupérer les données nécessaires pour la prise de décision sur le choix de la configuration future du système.

4. Méthodologie de conduite d'une simulation d'un RMS

Généralement, une étude de simulation de flux se déroule en quatre étapes : analyse du problème, modélisation du système et implémentation du modèle, conduite des expérimentations sur le

modèle, et rapporter les résultats et les conclusions. L'enchaînement de ces étapes est séquentiel, mais des re-bouclages ou des retours en arrière sont possibles afin de corriger ou de compléter l'élaboration du modèle en fonction des objectifs fixés (voir Figure 52).

Pour le cas particulier des systèmes reconfigurables, deux étapes doivent être rajoutées aux processus de simulation, ces étapes sont :

- L'identification, la définition et la conception des modules du système.
- Développement d'une bibliothèque virtuelle qui regroupe les modules (virtuels) de base qui composent le système.

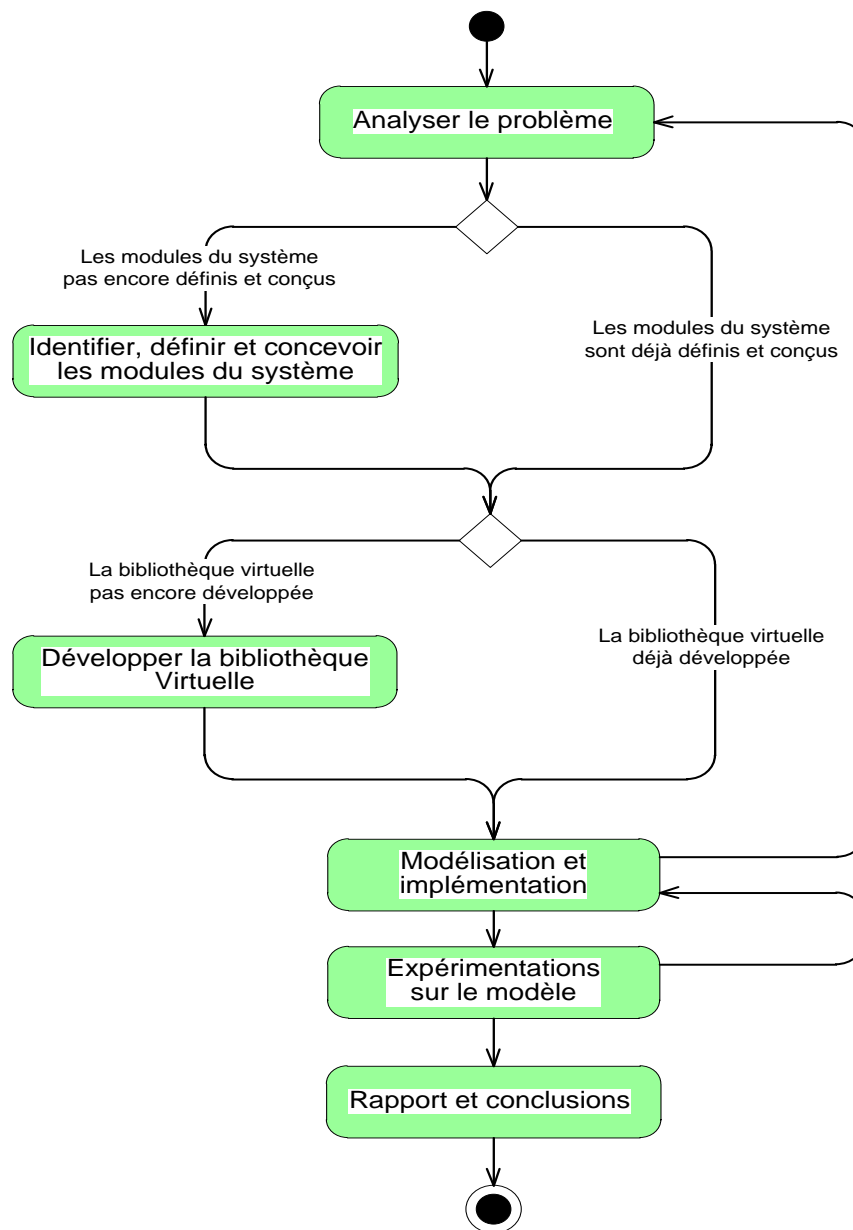


Figure 52 : la méthodologie globale pour la simulation des RMSs.

4.1. Analyse du problème

L'analyse du problème est une étape indispensable et d'une grande importance, puisque c'est dans cette étape que l'on doit définir précisément ce que l'on veut mettre en évidence avec la simulation, et quelle précision on attend. On détermine aussi les indicateurs de performance qui vont permettre de vérifier si l'on atteint les objectifs qu'on s'est fixés. Enfin, il faut pouvoir fournir des données numériques au modèle. Celles-ci sont relatives à tous les éléments utilisés dans la simulation, par exemple :

- Données sur les produits à fabriquer : gammes de fabrication (routage), loi d'arrivée dans le modèle (fréquence et taille de lot).
- Données sur les moyens de production : nombre et types de machines, lois de pannes, nombre et types de ressources complémentaires de production (outils, palettes, etc.).
- Données sur les systèmes de manutention : nombre et type de convoyeurs ou de chariots, capacité, vitesse de déplacement, etc.
- Données sur les stocks et les magasins : types et capacités, règles d'entrée et de sortie, etc.
- Données sur le personnel : effectifs, compétences et horaires.

À ces données numériques, il convient d'en ajouter d'autres qui s'expriment sous forme logique et qui caractérisent les règles de pilotage de l'atelier. Il est à noter que bien souvent, l'objectif de la simulation est de tester certaines règles pour déterminer les plus pertinentes. À titre d'exemple :

- Règles de lancement et de séquençement des produits en fabrication : à la commande ou en fonction des prévisions, par lots ou à l'unité, périodique ou aperiodique...
- Règles d'ordonnancement des produits sur les différentes machines de l'atelier.
- Règles d'affectation des ressources.

Enfin, en complément aux données numériques et logiques, on doit disposer de documents graphiques, à la fois pour avoir une représentation géométrique ou spatiale (exemple : le plan de l'usine ou de l'atelier) du système simulé, mais aussi pour avoir une représentation des flux. La représentation des flux permet de décomposer le processus en actions élémentaires telles que les opérations de stockages, de transferts et de contrôles.

4.1.1. Modèle géométrique du système

On rappelle que l'objectif du modèle géométrique ou graphique du système est de disposer d'un document sur lequel toutes les données nécessaires à la réalisation du modèle informatique de simulation sont réunies. L'une des premières étapes de la réalisation de ce modèle est l'identification des objets. Cette étape nécessite une analyse du système réel qui va permettre au concepteur du modèle de filtrer les éléments physiques réels pour passer aux objets du modèle graphique. Il est donc

important dans cette étape de décider quels sont les éléments qui doivent être modélisés, et avec quel niveau de précision (un exemple d'un modèle géométrique est donné dans la Figure 56).

4.1.2. Modèle informationnel du système

L'objectif du modèle informationnel est de réaliser une modélisation préalable en faisant apparaître les différentes données, de nature numérique ou logique, ainsi que la décomposition en objets de simulation : machines, convoyeurs, articles, stocks, etc. Ceci permet de créer un document de synthèse sur lequel seront rassemblées toutes les données utiles à la construction du modèle d'action. Le modèle informationnel est un outil de synthèse et de communication entre les différents acteurs d'une étude de simulation. Les objectifs d'un tel modèle sont multiples :

- Structurer et rassembler les données utiles à la construction du modèle informatique.
- Servir d'outil de communication entre les différents acteurs (opérateurs, techniciens de maintenance, ingénieurs, responsables...) à la fois pour la conception d'un système de production, mais aussi pour son exploitation.
- Permettre de représenter aussi bien les flux physiques qu'informationnels. – Identifier et définir les règles et lois utiles au pilotage du système.
- Servir de support à la réalisation du modèle informatique.

L'une des caractéristiques essentielles de ce modèle est de devoir être construite à partir d'objets graphiques symbolisant les éléments du système réel. Le modèle informationnel est très imagé par rapport au système réel. On retrouve, sur le même modèle, à la fois une modélisation du flux physique, du flux informationnel, des données utiles à la simulation (un exemple d'un modèle informationnel dans Figure 57).

4.2. Identification des modules de base du système

Pour faciliter la conception initiale du système, mais surtout la re-conception lors d'une reconfiguration, le concept de modularité est indispensable. La modularité permet de réduire l'espace de décision lors de la conception, donc d'accélérer la phase de conception. Elle limite les possibilités lors de la phase de conception en ayant fait des choix pré établis à travers les modules. Cette limitation peut être réduite en introduisant les paramètres aux modules. Pour que la modularité soit efficace, il faut bien définir les modules puisqu'ils seront la base de construction du système. Les interfaces entre les modules doivent être aussi bien définies (informationnelles, électriques, mécaniques, etc.).

Cette étape est nécessaire dans le cas où les modules du système ne sont pas encore définis. Le système doit être analysé et décomposé en plusieurs modules standards qui doivent être bien définis en termes d'interfaces (physiques et logiques), de comportement interne, et de paramètres. Voir le chapitre 3 pour plus de détails sur l'identification des modules de base d'un système, (voir le chapitre 3). Le Figure 53 illustre la modularité des systèmes manufacturiers.

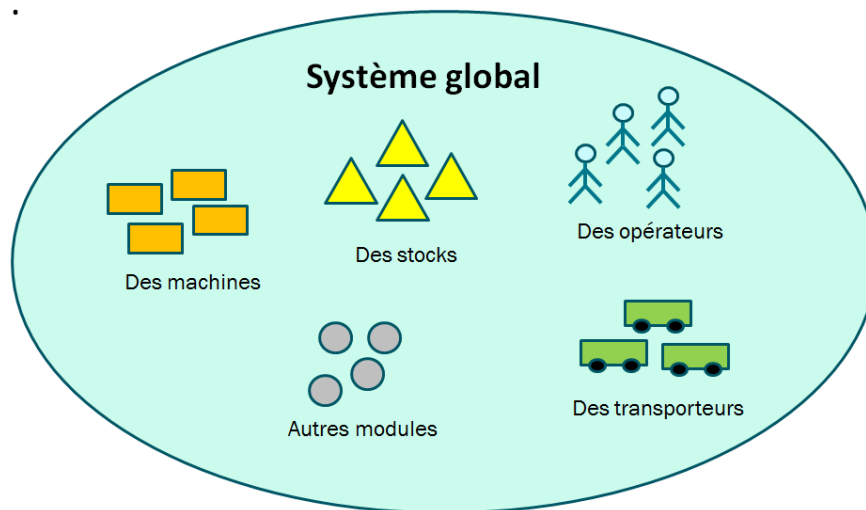


Figure 53 : concept de modularité dans les systèmes manufacturiers.

Dans la méthodologie globale proposée pour la conception des RMSs (voir chapitre 4), la simulation accompagne la définition et la conception du système. En parallèle de la définition des modules de base du système, les modèles de simulation de ces modules sont construits, simulés et analysés. Les résultats d'analyse servent à spécifier ces modules et les redéfinir si nécessaire. Au cours de ce processus, on a constaté que la simulation nous aide énormément dans l'identification des entrées et des sorties (les interfaces) de chaque module, les paramètres que l'utilisateur sera invité à saisir, et les différentes configurations possibles du module.

4.3. Développement de la bibliothèque virtuelle

Dans le cas où la bibliothèque virtuelle du système n'est pas encore développée. Cette étape sera nécessaire pour faciliter la suite du processus de simulation. Le principe est de développer un module pour chaque module du système (modules déjà définis dans l'étape précédente). Ces modules virtuels seront regroupés pour former ce qu'on appelle une bibliothèque virtuelle. Cette bibliothèque sera utilisée pour monter facilement différents modèles ou configurations du système. Il y a peu de logiciels de simulation qui permettent l'implémentation des bibliothèques virtuelles. Pour nos cas d'études (voir chapitre 6), on a utilisé le logiciel Arena de Rockwell pour le développement et l'implémentation de notre bibliothèque virtuelle.

Arena est un outil de simulation avancé qui offre un environnement interactif pour construire, animer, vérifier et analyser des modèles de simulation (Rockwell Automation 2012). Initialement lancé en 1993, Arena emploie une conception orientée-objet pour un développement entièrement graphique des modèles. Les analystes de simulation placent des objets graphiques, appelés modules, sur l'espace de travail afin de définir les composants du système tels que les machines, les opérateurs, les dispositifs de manutention, etc. ARENA est construit sur le langage de simulation SIMAN. Après

avoir créé graphiquement un modèle de simulation, Arena génère automatiquement le modèle SIMAN qui sera utilisé pour exécuter la simulation (Takus & Profozich 1997).

Les modules graphiques utilisés par les analystes de simulation pour créer des modèles sont fournis avec le logiciel ARENA. Ces modules peuvent également être conçus spécialement pour produire un environnement de modélisation adapté à un domaine d'application spécifique (logistique, usinage, gestion des stocks, semi-conducteur, textile..., etc.) (Takus & Profozich 1997). Arena permet la construction des modules de simulation qui peuvent être très simples, comme celui qui compte les clients lorsqu'ils quittent une banque, ou très complexes, comme celui qui capture toutes les activités sur un quai d'un chantier naval. Il permet aussi de combiner les modules développés pour former des modules plus complexes. Les modules développés peuvent être collectés dans des bibliothèques qui peuvent être utilisées pour supporter les différentes activités de simulation et peuvent être aussi partagées avec d'autres utilisateurs du logiciel Arena (Rockwell Automation 2012). La Figure 54 illustre le processus de création de bibliothèque virtuelle avec l'outil Arena.

Définition des modules du système

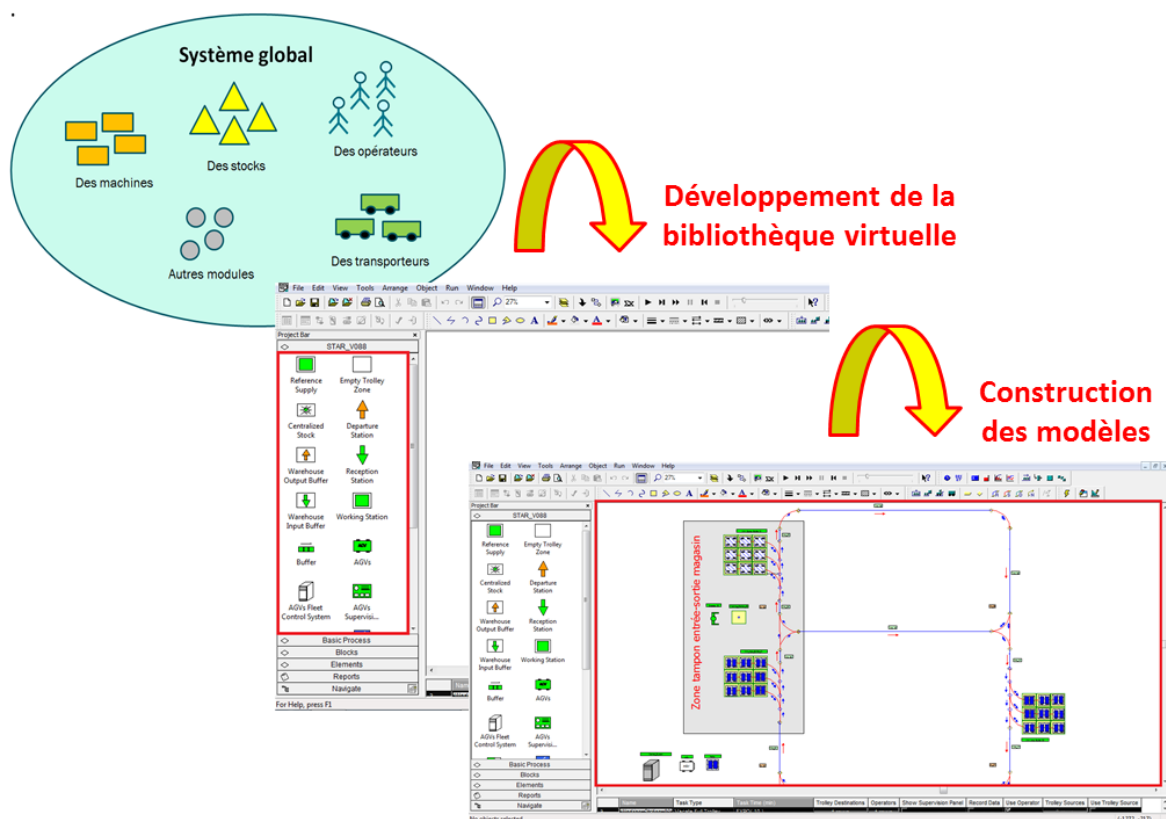


Figure 54 : processus création d'une bibliothèque virtuelle.

4.4. Modélisation et implémentation

La construction du modèle permet de coder le modèle dans un langage informatique approprié. Cette étape est de plus en plus facilitée par l'évolution des logiciels de simulation. Pour le cas

particulier des systèmes reconfigurables, la construction du modèle est facilitée par l'utilisation des bibliothèques virtuelles développées précédemment. Le concepteur du modèle n'a plus besoin d'être un expert en simulation des flux pour développer des modèles, les vérifier, conduire des tests sur ces modèles et en tirer des conclusions qui en seront exploitées pour la gestion du système simulé.

La phase de modélisation se déroule généralement en plusieurs étapes. On commence par construire un modèle global du système, puis on l'affine progressivement, en suivant éventuellement une méthodologie, en validant chaque étape avant de perfectionner le modèle. Le problème qui se pose lors de la modélisation est de savoir jusqu'à quel niveau de détails il faut aller pour que le modèle soit représentatif de la réalité. En ce sens, il est donc préférable de rechercher à affiner le modèle, plutôt que de se contenter d'un modèle trop général et imprécis.

Une fois le modèle réalisé, il faut vérifier si les règles logiques qui décrivent les flux sont bien programmées ou implémentées et correspondent à ce qui est demandé. On exécute donc des tests de simulation uniquement pour vérifier le bon fonctionnement du modèle. Si des différences apparaissent, il faut pouvoir "tracer" tous les événements du modèle et vérifier leur cohérence par rapport aux données. Cette étape doit se terminer par une validation qui consiste par exemple à comparer les résultats fournis par le modèle aux résultats du système réel si celui-ci existe. Les rapports statistiques peuvent aider à cette validation entre résultats simulés et résultats réels. La visualisation graphique et dynamique du modèle est un atout considérable pour en faciliter sa vérification et sa validation, puisque l'on peut suivre tous les changements d'état qui interviennent sur le modèle.

4.4.1. Classification des modèles de simulation

On a jugé qu'une classification générale des modèles de simulation est nécessaire. [Law & Kelton \(1991\)](#) proposent une classification de ces modèles, on distingue :

- **les modèles physiques des modèles logico-mathématiques** : les modèles physiques sont ceux dans lesquels le système réel est représenté par une réplique ou une maquette, à une échelle différente et éventuellement à l'aide de matériaux différents. Ils sont utilisés à des fins d'entraînement : simulateurs de vol, de conduite, maquettes de véhicules pour des essais aérodynamiques...etc. Les modèles logico-mathématiques ou symboliques sont définis par des relations logiques et quantitatives qui sont manipulées et changées pour voir comment le modèle du système réel réagit. Ils sont exécutés sur des ordinateurs. C'est exclusivement ce type de modèle qui sera utilisé dans la suite de ce travail.
- **Les modèles déterministes des modèles stochastiques** : cette distinction concerne la prise en compte d'aléas ou de variations aléatoires dans le modèle. Si le système est indépendant de l'influence de variables aléatoires ou imprévisibles, on utilise un modèle déterministe. Contrairement, si les aléas jouent un rôle significatif dans le comportement du système (exemple typique : les pannes), on utilise un modèle stochastique.

- **Les modèles statiques des modèles dynamiques** : cette classification distingue les modèles statiques, pour lesquels le temps n'intervient pas (exemple : modèle comptable permettant de calculer le bénéfice en fin d'année à l'aide d'un tableur), des modèles dynamiques, pour lesquels le comportement est une fonction du temps (exemple : système de maintenance dans une usine).
- **Les modèles dynamiques à évènements discrets (discontinus) des modèles dynamiques continus** : cette distinction concerne les modèles dynamiques seulement. Les modèles à évènements discrets (ou discontinus) dans lesquels les changements d'état ne surviennent que lors d'évènements tels le début ou la fin d'une opération, la mise en attente d'une pièce dans un stock, la libération d'une ressource... Dans une simulation à évènements discrets, les flux essentiels que l'on examine sont composés d'éléments isolables que l'on peut dénombrer et identifier individuellement. Ces éléments sont couramment appelés "Entités" ou "Articles". Les modèles continus, plus adaptés aux flux continus, utilisent des équations mathématiques pour prendre en compte les changements d'état qui s'effectuent de façon continue au cours du temps. Les valeurs des variables d'état sont recalculées régulièrement selon un pas d'horloge d'après ces équations.

Dans le domaine de la simulation des flux, les modèles à évènements discrets sont les plus adaptés. La Figure 55 résume la classification des modèles de simulations.

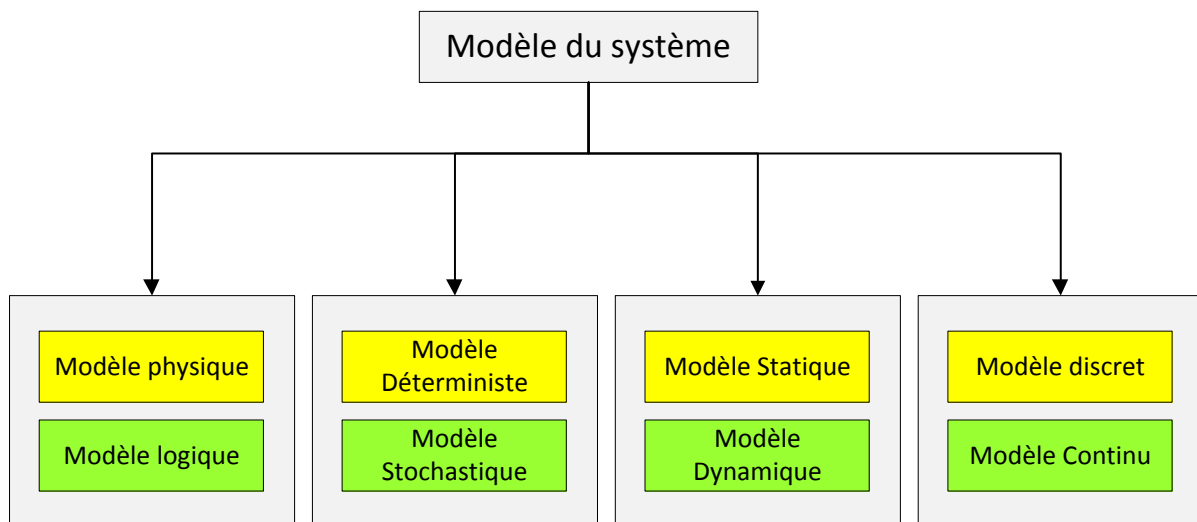


Figure 55 : classification des modèles d'un système.

4.5. Expérimentations sur le modèle

L'exploitation de la simulation est l'étape où l'on utilise le modèle comme support expérimental pour évaluer le comportement dynamique du système. Il faut bien sûr avoir défini les données sur lesquelles on va pouvoir agir (les paramètres de performances) pour atteindre les objectifs que l'on s'est fixés dans la 1ère étape (les indicateurs de performances). Un scénario ou une expérimentation se

caractérise donc par un jeu de données qui varie à chaque itération du processus expérimental. Il faut être capable d'interpréter les résultats fournis par la simulation. Cela suppose la maîtrise de notions de statistique telles que l'intervalle de confiance, la moyenne arithmétique ou temporelle, l'écart type, etc.

4.6. Rapport et conclusions

Cette dernière étape est importante vis-à-vis du demandeur de l'étude de simulation. En effet, celui-ci n'est pas forcément un expert dans ce domaine et il va falloir présenter les résultats de l'étude pour qu'ils soient compréhensibles. La majorité des progiciels proposent en standard des résultats de simulation sous la forme de rapports statistiques qui peuvent être complétés de graphiques pour faciliter l'analyse. Les indicateurs "standards" concernent essentiellement les quantités et les durées relatives aux flux d'entités, ainsi que les taux d'utilisation, de panne, d'arrêt des machines et ressources.

5. Cas d'application

Dans cette section, on présentera un cas d'étude qui permet d'illustrer les points abordés précédemment sur l'utilisation de la simulation dans la conception des RMSs.

5.1. Description

En premier lieu, on va décrire le comportement du système qui fait l'objet de notre cas étude :

- Les produits arrivent dans le système avec une cadence qui peut être modifiée afin de tester les différents scénarios possibles du changement du contexte de production (dans ce cas, on se limite au changement de la cadence). Ces produits sont reçus dans une zone qu'on appelle zone de conditionnement au niveau de laquelle les produits sont conditionnés et préparés pour être envoyés et stockés dans le magasin de stockage pour qu'ils soient prêts à être livrés à la demande du client.
- Juste après le conditionnement, les produits sont chargés dans des chariots, ces chariots sont mis à des emplacements bien définis pour être transportés vers le magasin de stockage.
- Le transport des chariots pleins de la zone de conditionnement vers le magasin de stockage est assuré par un ou plusieurs AGVs selon le besoin,
- Les produits ne sont pas mis directement dans le magasin de stockage, mais ils sont mis en attente dans une zone tampon en attendant d'être déchargés des chariots et mis dans leurs emplacements dans le magasin. Cette tâche est exécutée par des opérateurs puisqu'elle nécessite beaucoup de travail manuel.
- Les chariots vides sont mis par les opérateurs dans une autre zone tampon. Ces chariots sont transportés par les AGVs vers la zone de conditionnement pour être réutilisés, et le processus se répète.

5.2. Analyse du système

Dans un premier temps, nous avons utilisé un modèle géométrique (Figure 56) pour avoir une vision globale et claire du processus, ce modèle montre la disposition des objets (stock, machine, etc.) et les liens entre eux, il est accompagné d'un modèle de connaissance (Figure 57). Ce dernier est utilisé principalement pour faire apparaître les données ou les informations (taille d'un stock, le temps de cycle d'une machine, etc.) nécessaires à l'analyse du système. Cette façon de séparer le modèle géométrique qui représente la réalité du système, du modèle de connaissances ou informationnel qui fait apparaître les données nécessaires à l'analyse du système, facilite la compréhension, l'analyse du système, et par conséquent l'identification et la définition des modules de base qui le composent. Cette séparation de modèles facilite aussi le développement des modules virtuels à partir desquels des modèles de différentes configurations possibles du système peuvent être construits, testés et comparés.

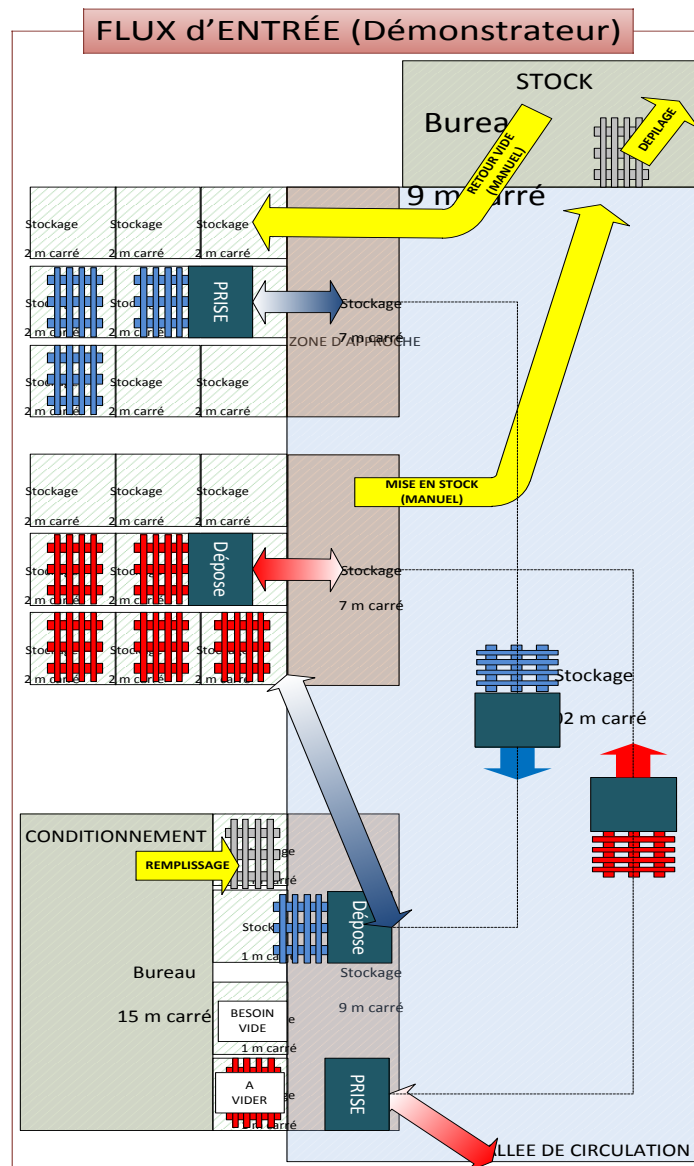


Figure 56 : modèle géométrique du système.

	AGVs, c.-à-d. cette zone ne sera pas en interaction avec les AGVs. Les opérateurs sont les seuls qui sont en interaction avec cette zone. Ils peuvent déposer ou prendre des chariots de cette zone.
AGV	L'AGV est le transporteur qui sera utilisé pour transporter les différents supports chargés et vides entre les différents points (zone de conditionnement, buffer entrée magasin, buffer sortie magasin, stations, etc.) dans le modèle.
Zone de chargement	Cette zone sert comme une station de chargement de la batterie de l'AGV.
La zone de repos des AGVs	Le module 'Zone de repos' est créé pour répondre à ce besoin, il sert comme une zone de repos pour les AGVs en attente de recevoir une nouvelle mission.
Opérateur	Ce module sert à modéliser un opérateur, ce dernier sera en charge d'effectuer toutes les tâches manuelles, par exemple : le chargement et le déchargement des supports, déplacement des supports, etc.
Intersection	Ce module sert à modéliser les intersections qui font partie du circuit utilisé par les AGVs pour se guider et se déplacer dans l'atelier.
Lien	Ce module sert à modéliser les liens qui constituent le circuit utilisé par les AGVs pour se guider et se déplacer dans l'atelier.
Générateur d'ordres	Ce module sert à générer des ordres comme les ordres de travail.
Système de contrôle de la flotte d'AGVs	Ce module sert à contrôler la flotte d'AGVs. Il reçoit les demandes de transport et d'approvisionnement et envoie les ordres de déplacement aux différents AGVs.

Tableau 12 : les modules identifiés du système.

5.4. Développement de la bibliothèque virtuelle

Pour chaque module identifié du système, un module virtuel doit être développé et qui sera utilisé dans les différents modèles de simulation du système. On rappelle ici que l'outil utilisé pour l'implémentation des modules listé dans le Tableau 12 est Arena (rajouter un renvoi vers la section Arena). On se limite ici au module 'Buffer entrée magasin' (voir Figure 58) pour montrer le développement la bibliothèque des modules virtuels.

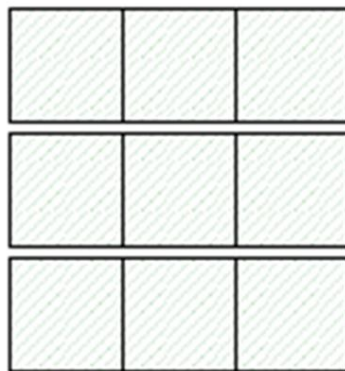


Figure 58 : forme géométrique du module identifié 'Buffer entrée magasin'.

La Figure 59 montre le module virtuel (vue ou modèle géométrique) qui correspond au module 'Buffer entrée magasin'.

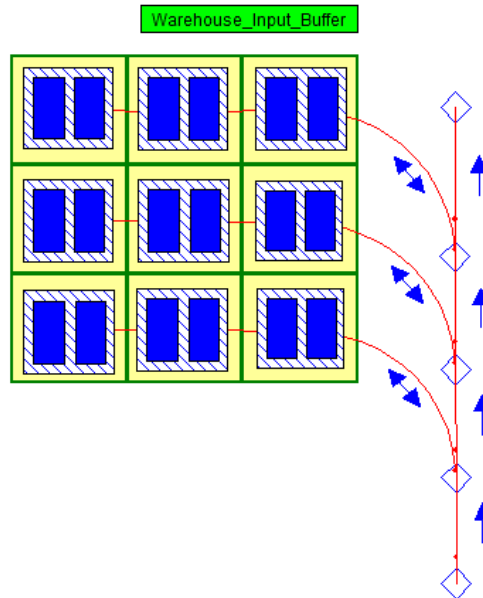


Figure 59 : modèle (vue géométrique) du module identifié.

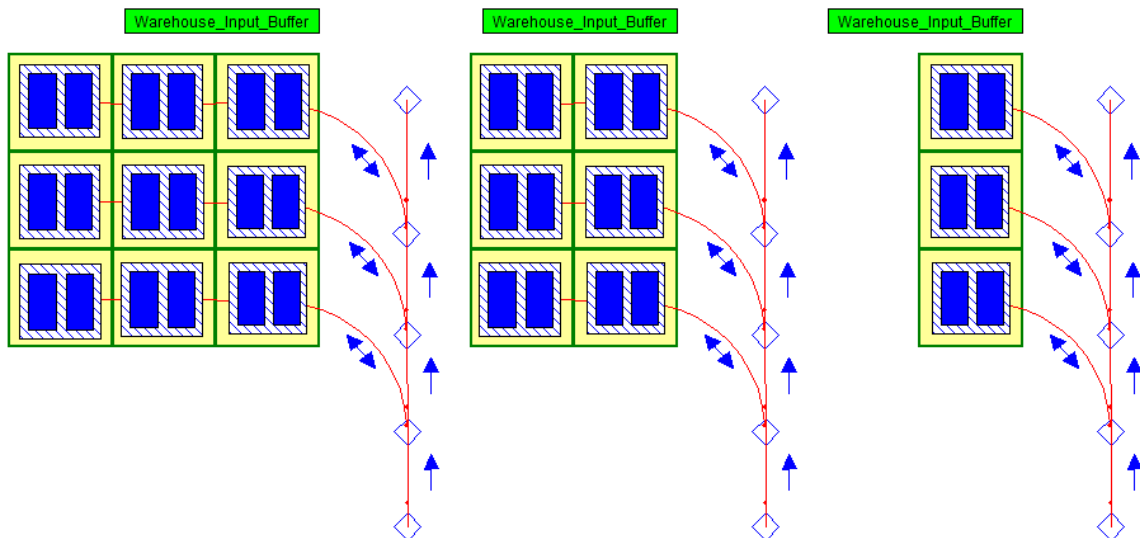


Figure 60 : les configurations possibles du module virtuel développé.

La Figure 61 montre l'interface graphique d'utilisateur développé pour le module virtuel 'Buffer entrée magasin'.

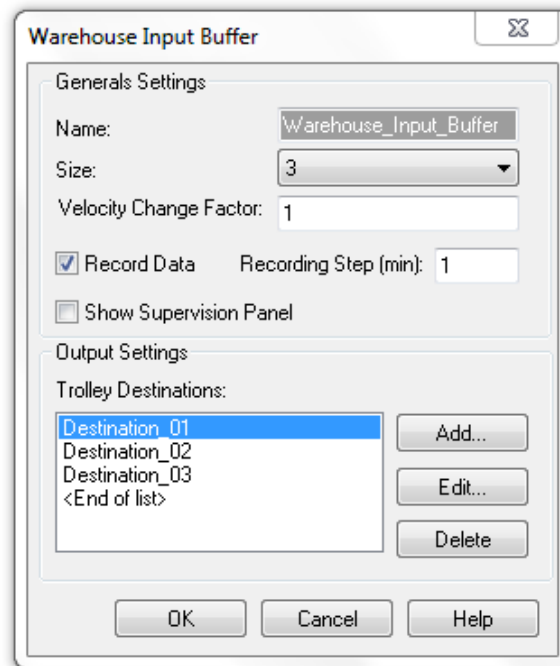


Figure 61 : interface graphique utilisateur du module développé.

Le Tableau 13 résume l'ensemble des paramètres qui figurent sur l'interface illustrée par la Figure 61.

Paramètre	Description
'Name'	Le nom du Buffer, par exemple 'Input Buffer 01'. Ce nom sera l'identifiant du module dans le modèle.
'Size'	La taille du Buffer, trois configurations possibles : 3 places, 6 places, et 9 places.
'Velocity Change Factor'	Le rapport de changement de vitesse du transporteur. Pour des raisons de sécurité, la vitesse du transporteur (AGV) sera réduite à l'approche du buffer. Si le rapport de réduction est de 0.5 par exemple, la vitesse de l'AGV sera réduite de moitié à l'approche du buffer.
'Record Data'	Activer ou non la sauvegarde des données de simulation du module. Ces données peuvent être utilisées pour évaluer les performances du système ou pour le diagnostic.
'Recording Step'	Le pas d'enregistrement des données à indiquer en minutes (1 min, 1/60 min par exemple).
'Show supervision Panel'	Afficher ou non le panneau de supervision du module pendant la simulation.
'Trolley Destinations'	La liste des destinations possibles du chariot à la sortie du buffer. On peut rajouter une destination à la liste, modifier ou supprimer une destination qui existe déjà dans la liste à l'aide des trois boutons 'Add', 'Edite' et 'Delete' respectivement. L'interface ci-dessous s'affiche à chaque clique sur l'un de ces trois boutons. Cette interface sert à indiquer le nom de la destination. (N.B : pour éviter les erreurs, la destination à indiquer doit être choisie parmi l'ensemble des destinations possibles dans le modèle).

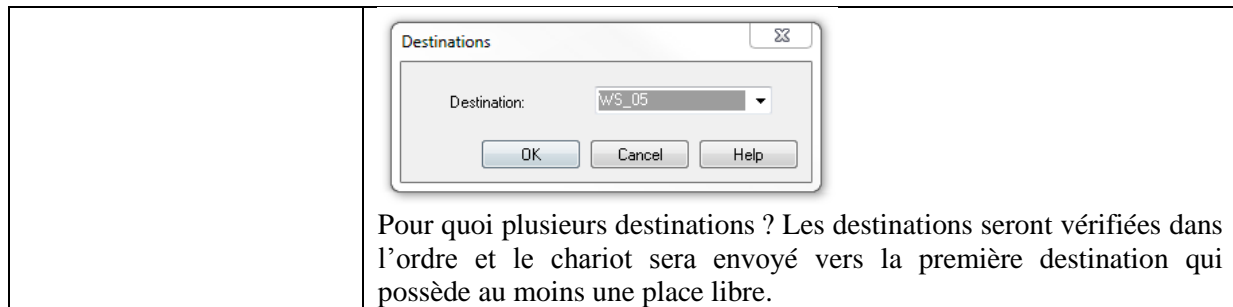


Tableau 13 : Description des paramètres de l'interface utilisateur représentée par la Figure 61.

Une fois, les modules sont bien définis, ainsi que leurs modèles de simulation individuels, ces derniers sont regroupés pour former ce qu'on appelle une bibliothèque virtuelle. Pour simuler le système dans sa globalité, il n'est plus nécessaire de perdre du temps à construire son modèle à partir des objets de base offerts par l'outil de simulation, mais on utilise directement les modules de la bibliothèque déjà développée. Cela facilite la construction du modèle du système même pour une personne non initiée à la simulation. Pour plus de détails sur les modules de la bibliothèque virtuelle, voir le Tableau 14.

5.5. Modélisation et implémentation

Généralement, la construction d'un modèle nécessite la maîtrise de l'outil utilisé. En cas de besoin, et pour plus de détails, on peut toujours s'orienter vers les différents documents d'aide intégrés par défaut dans le logiciel utilisé. Concernant l'utilisation de la bibliothèque, tous les logiciels d'aujourd'hui présentent une interface graphique qui facilite la construction des modèles. Pour rajouter un module dans le modèle ou dans l'espace de travail, cliquer sur ce module dans la bibliothèque et faire glisser le curseur jusqu'à l'endroit où on veut le placer dans le modèle puis lâcher le bouton. Répéter cette manipulation pour tous les modules qu'on veut mettre dans le modèle. Une fois que tous les modules sont placés dans le modèle, il faut les paramétrer en faisant attention aux données saisies parce que plusieurs erreurs sont générées à cause d'une mauvaise saisie des paramètres. Une fois le modèle est construit, lancer la simulation. Une phase de paramétrage nécessaire avant de lancer la simulation. On peut par exemple vouloir fixer le durée de simulation, les données dont on veut récupérer une fois la simulation est terminée, etc.

5.6. Expérimentation (scénarios de reconfiguration)

5.6.1. Configuration initiale

Pour la configuration initiale du système, on démarre avec une cadence normale (un chariot est préparé toutes les 10 minutes), cette cadence ne nécessite qu'une seule zone tampon (buffer_01) en entrée magasin pour contenir tous les chariots qui arrivent de la zone de conditionnement sans débordement. La Figure 62 montre le modèle de cette configuration. Ce modèle a été construit facilement à partir de la bibliothèque mise en place.

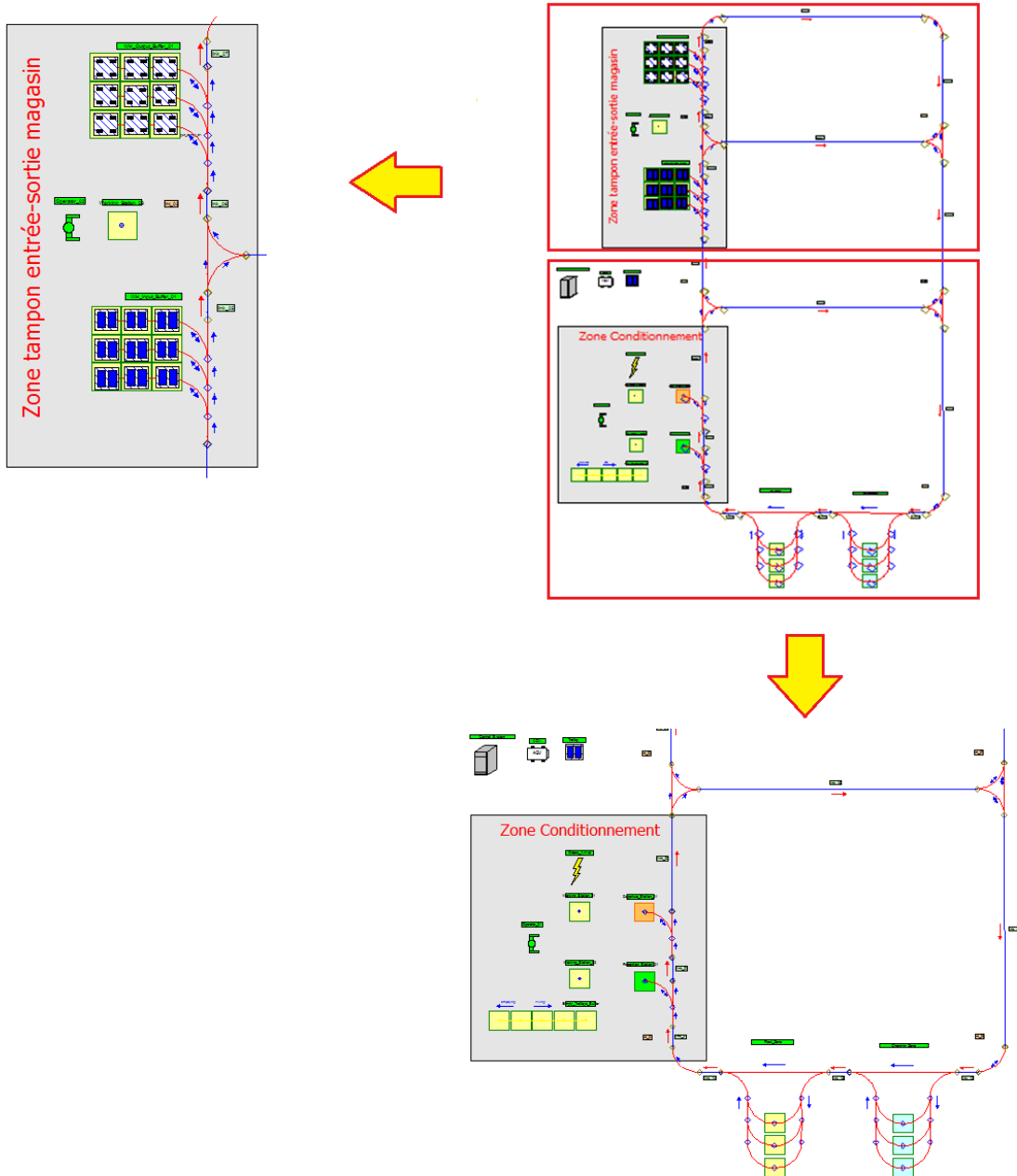


Figure 62 : la configuration initiale du système.

5.6.2. Nouvelle Configuration

Supposons que le contexte de production vient de changer et que la cadence de production vient d'augmenter à un point où une seule zone tampon en entrée magasin ne peut plus accueillir l'ensemble des chariots qui arrivent de la zone de conditionnement. Alors, on ajoute une nouvelle zone tampon en entrée magasin 'Buffer_02'. Si le 'Buffer_01' est rempli, les chariots ou palettes seront redirigés automatiquement vers le 'Buffer_02'. On ajoute en conséquence le circuit de l'AGV qui relie la zone de conditionnement au 'Buffer_02' (reconfiguration du circuit et des stocks). La Figure 63 montre la

nouvelle configuration du système en mettant en évidence le module rajouté (Buffer entrée magasin) au système.

On teste le fonctionnement de la nouvelle configuration et l'on valide si elle donne des résultats satisfaisants par rapport au nouveau contexte et éventuellement aux indicateurs de performances prédéfinis sinon on passe à une autre configuration.

Virtuellement, la reconfiguration du système (le modèle) est très facile et rapide grâce à la bibliothèque mise en place pendant la phase de définition et de conception du système, tout ce qu'on a fait est de rajouter une nouvelle zone tampon en entrée du magasin, la relier au circuit de l'AGV et paramétrer les éléments suivants :

- Système de pilotage : indiquer le nouveau module 'Buffer_02' en entrée magasin comme une deuxième destination en cas où le 'Buffer 01' en entrée magasin est rempli.
- Le 'Buffer_02' en entrée magasin : indiquer la taille et les autres paramètres (voir Figure 61 et Tableau 13).

Sur le terrain, cette reconfiguration revient à ajouter une nouvelle zone tampon en entrée magasin qui est supposée être un module standard déjà développé et facile à déployer. Ensuite, étendre le circuit de l'AGV pour pouvoir desservir cette zone ainsi qu'un paramétrage simple du système de contrôle pour indiquer l'ensemble des destinations et l'ordre utilisé pour les desservir ('Buffer' 01 puis 'Buffer 02'). La figure ci-dessous montre le modèle de la nouvelle configuration.

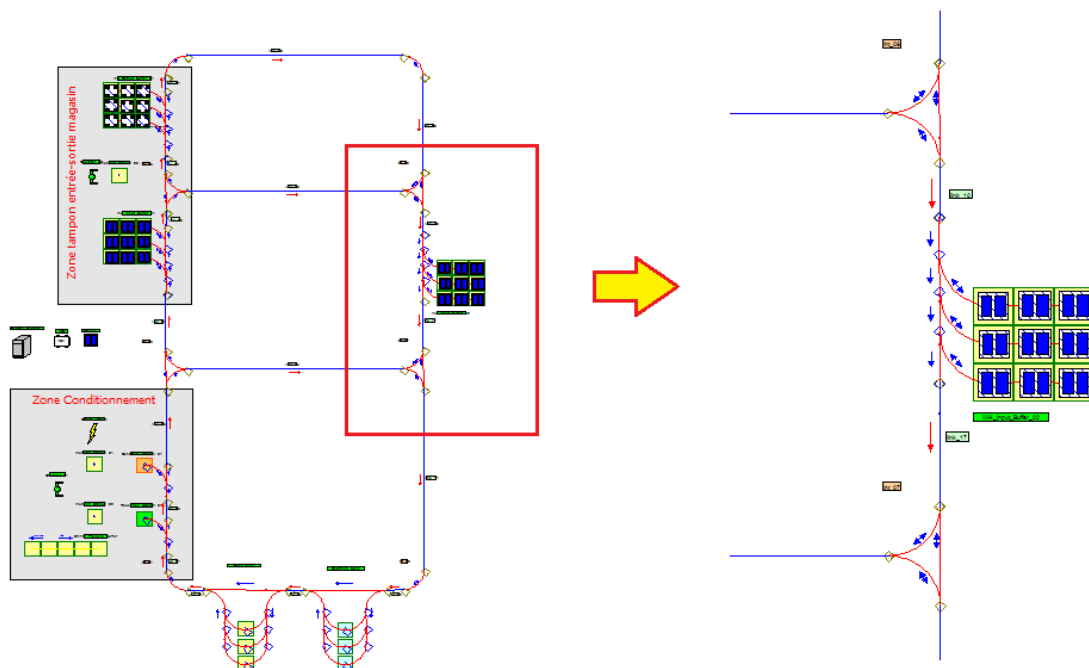


Figure 63 : la nouvelle configuration du système.

Si la cadence des produits à conditionner augmente davantage, il est probable qu'un seul AGV ne pourra pas supporter la nouvelle cadence de l'ensemble des flux matériels, donc une reconfiguration du système est nécessaire. Dans ce cas, on rajoute un AGV pour réduire la charge du travail sur le premier. Pratiquement, cela revient à rajouter un nouvel AGV dans le système, le déclarer dans le système de contrôle et lui affecter les paramètres nécessaires (la vitesse, les missions, le circuit de circulation, etc.). La reconfiguration réelle du système ne doit pas être trop différente de la reconfiguration virtuelle.

5.7. Rapport et conclusion (Quelques résultats de la simulation)

En guise d'exemple, on donne ici quelques résultats de la simulation générés par le logiciel Arena. La durée de simulation est 1 jour ou 1440 minutes.

5.7.1. Configuration initiale

La configuration initiale est décrite dans la section 5.6.1 de ce chapitre et la Figure 62 donne une vue globale de cette configuration.

Paramètres de la simulation

Au niveau de la zone de conditionnement :

- Le temps entre l'arrivée de deux produits à conditionner suit une loi exponentielle d'une moyenne de 10 min.
- Le temps nécessaire pour conditionner un produit suit une loi exponentielle d'une moyenne de 10 min.
- Le premier produit à conditionner arrive à $t=5$ min.

Au niveau de la zone tampon :

- Le temps nécessaire pour vider un chariot qui se trouve dans la zone tampon 'Buffer_01' suit une loi exponentielle d'une moyenne de 10 min.

Résultats de la simulation

La Figure 64 montre l'état de remplissage de la zone tampon en entrée du magasin 'Buffer_01', la capacité d'accueil maximale de cette zone est de 9 chariots. On voit bien que pendant toute la durée de la simulation, le nombre maximum de chariots présents simultanément est 5, donc on est loin d'atteindre la capacité maximale de la zone tampon. Voici quelques données utiles récupérées à la fin de la simulation :

- La moyenne du nombre des chariots présents simultanément dans le 'Buffer_01' = 0.43
- Le nombre minimum des chariots présents simultanément dans le 'Buffer_01' = 0
- Le nombre maximum des chariots présents simultanément dans le 'Buffer_01' = 5

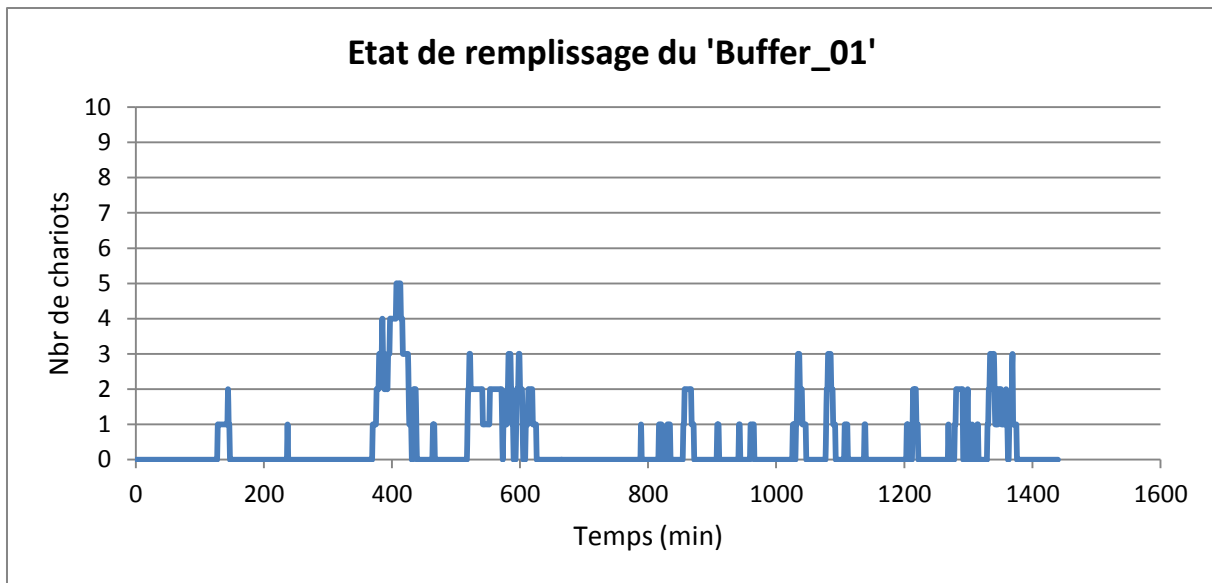


Figure 64 : état de remplissage du 'Buffer_01', configuration initiale.

Augmentation de la cadence

On suppose que le temps entre l'arrivée de deux produits successive à conditionner suit une loi exponentielle d'une moyenne 5 min (on a doublé la cadence d'arrivée des produits à conditionner).

Résultats de la simulation

La Figure 65 montre l'état de remplissage de la zone tampon en entrée magasin 'Buffer_01'. On voit bien que pendant toute la durée de la simulation, le nombre maximum de chariots présents simultanément est, à plusieurs reprises, égal à 9. Donc, dans le cas où aucune place n'est libre au niveau de la zone tampon 'Buffer_01', il sera impossible d'envoyer des chariots vers cette zone, cela bloquera toutes les opérations en amont et les activités au niveau de la zone de conditionnement s'arrêteront et les produits en attente d'être conditionnés s'accumuleront. Donc toute la chaîne logistique sera impactée et les performances du système seront médiocres. Pour remédier à cela, la reconfiguration du système est nécessaire, dans ce cas, la solution évidente est de rajouter une autre zone tampon 'Buffer_02' pour absorber le surplus du flux des chariots qui arrivent de la zone de conditionnement. Voici quelques données utiles récupérées à la fin de la simulation :

- La moyenne du nombre des chariots présents simultanément dans le 'Buffer_01' = 4.53
- Le nombre minimum des chariots présents simultanément dans le 'Buffer_01' = 0
- Le nombre maximum des chariots présents simultanément dans le 'Buffer_01' = 9

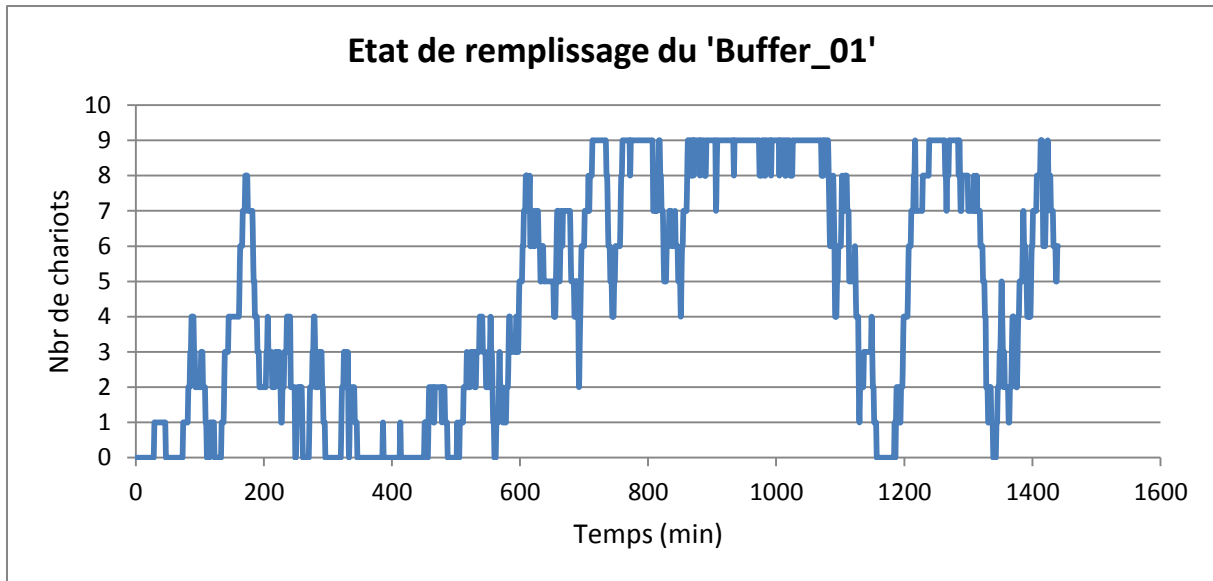


Figure 65 : état de remplissage du 'Buffer_02', configuration initiale, cadence augmentée.

5.7.2. Nouvelle configuration

Les figures suivantes (Figure 66, Figure 67) montrent respectivement l'état de remplissage de la zone tampon 'Buffer_01' et de la zone tampon 'Buffer_02' après la reconfiguration du système. Si on compare ces figures avec la figure précédente (Figure 65), on arrive facilement à la conclusion que la reconfiguration du système a résolu le problème de remplissage de la zone tampon 'Buffer_1' et par conséquent amélioré les performances du système.

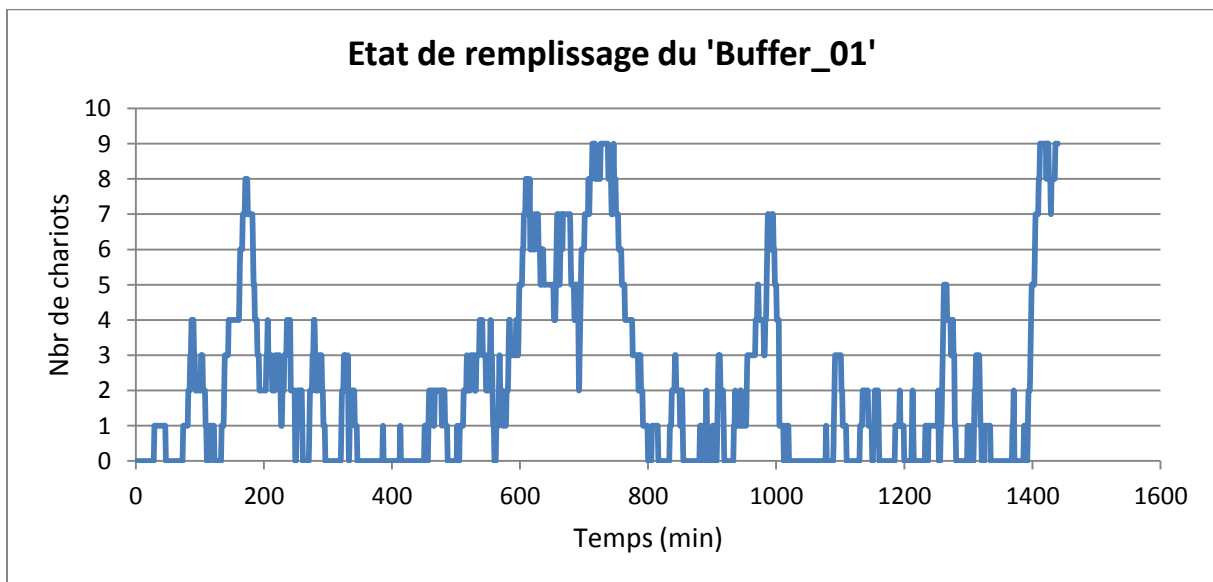


Figure 66 : état de remplissage du 'Buffer_01' après la reconfiguration du système.

Voici quelques données utiles récupérées à la fin de la simulation :

- La moyenne du nombre des chariots présents simultanément dans le 'Buffer_01' = 2.10

- Le nombre minimum des chariots présents simultanément dans le 'Buffer_01'= 0
- Le nombre maximum des chariots présents simultanément dans le 'Buffer_01'=9

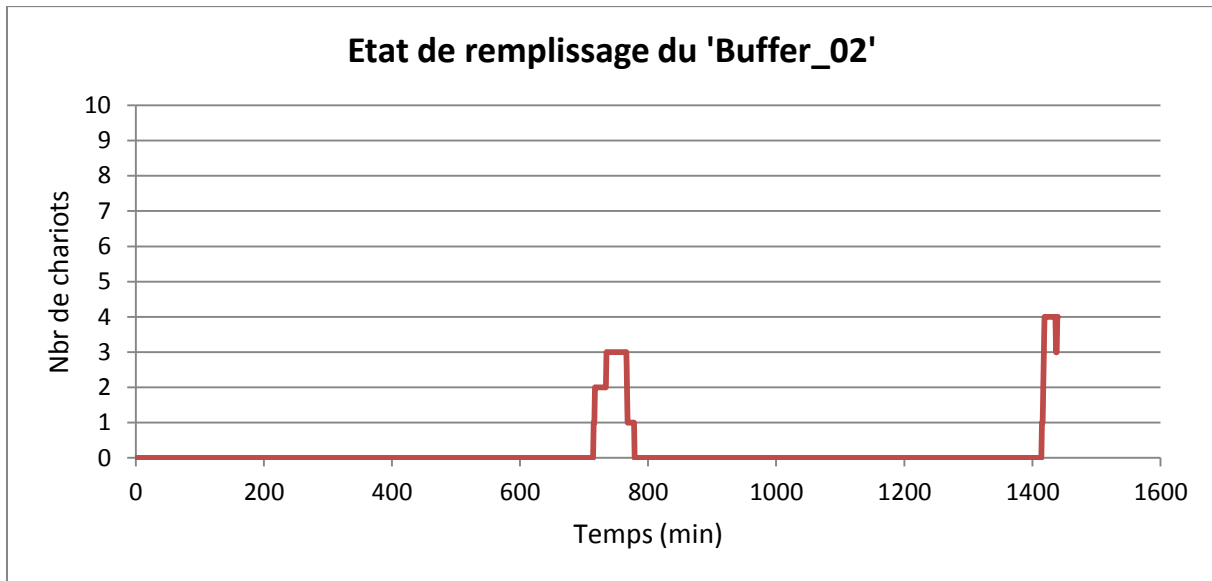


Figure 67 : état de remplissage du 'Buffer_02' après la reconfiguration du système.

Voici quelques données utiles récupérées à la fin de la simulation :

- La moyenne du nombre des chariots présents simultanément dans le 'Buffer_02'= 0.16
- Le nombre minimum des chariots présents simultanément dans le 'Buffer_02'= 0
- Le nombre maximum des chariots présents simultanément dans le 'Buffer_02'=4

6. Conclusion

Ce chapitre a permis de présenter la simulation de flux qu'est un outil utilisé pour analyser et optimiser le fonctionnement des systèmes manufacturiers. En effet, plutôt que de mener des expériences longues et coûteuses sur le système réel, la simulation permet de réaliser les mêmes expériences, mais de façon virtuelle, sur un ordinateur, sans aucun risque et avec un temps de réponse rapide.

On a aussi présenté le rôle de la simulation dans la conception des systèmes manufacturiers reconfigurables. On a montré l'importance de la considérer comme une étape incontournable pendant la définition, la spécification et la conception du système. Nous avons montré également comment elle peut aider pour choisir et valider une configuration quand la reconfiguration du système est nécessaire. Cette tâche sera aussi facilitée par la mise en place d'une bibliothèque de modules virtuels facile à utiliser pour la construction du modèle globale du système et ses différentes variantes (les configurations du système). Cela est utile une fois que le système est opérationnel parce que le passage d'une configuration à une autre nécessite forcément la validation de la configuration future du système

avant de la déployer réellement sur le terrain. Donc, à l'aide de la simulation et la bibliothèque mise en place, l'étape de choisir et de valider la configuration future du système sera réduite en termes de temps et d'effort. Les modèles des différentes configurations possibles seront facilement construits, testés, évalués, comparés et la configuration répondant le mieux aux besoins sera sélectionnée pour être déployée.

Pour la conduite d'une étude de simulation d'un RMS, on a présenté une méthodologie en 6 étapes (analyse du problème, identification des modules du système, développement de la bibliothèque virtuelle, construction des modèles, tests et expérimentations, et enfin rapports et conclusions). Un diagramme d'activités est présenté pour résumer la méthodologie et guider son application. On aussi montré à travers un exemple, qui est un cas d'étude sur lequel on a travaillé, comment la méthodologie proposée a été appliquée.

On prévoit d'aller plus dans l'utilisation de la simulation pour la conception et la gestion des RMSs. L'objectif est de pouvoir générer automatiquement les modèles de simulation à partir des modèles graphiques qui décrivent le système, cela facilitera la construction et la comparaison des différentes configurations possibles du système, et la personne chargée de tester et valider ces configurations n'aura même plus besoin d'avoir des connaissances approfondies dans la simulation de flux pour construire ces modèles. Parmi les objectifs finals est la mise en place d'un système de pilotage (MES) modulaire reprenant les objets de la bibliothèque de simulation et de le coupler avec le modèle de simulation pour plus d'assistance et d'aide à la décision lors des phases de production (voir figure ci-dessous).

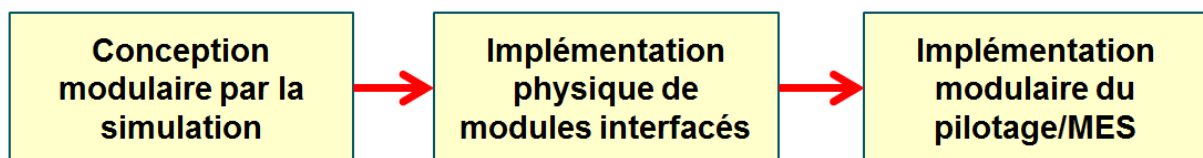


Figure 68 : Implémentation d'un système de pilotage modulaire.

Afin de mettre en évidence les points forts et limites de la méthodologie de conception des RMSs proposée dans le chapitre 4 ainsi que ceux de l'outil d'aide à la décision, pour la gestion et la reconfiguration des RMSs, présenté dans le présent chapitre, les deux cas d'étude sur lesquels on a appliqué la méthode et l'outil sont détaillés dans le chapitre suivant.

Chapitre 6 : Application sur des cas d'étude

1. Introduction

Dans ce chapitre, une description des deux cas d'étude sur lesquels la méthodologie proposée dans le chapitre 4 est donnée. L'outil d'aide à la décision basé sur la simulation de flux présenté dans le chapitre précédent est décrit plus en détail dans ce chapitre ainsi que son application sur l'un des cas d'étude (Faurecia).

2. Description des cas d'étude

On rappelle que cette thèse s'inscrit dans le cadre du projet STAR (Système Transitiq ue Agile et Robotisé), ce dernier se limite uniquement à la partie transitiq ue des systèmes.

2.1. Cas d'étude N°01: Faurecia, Usine Nogent-sur-Vernisson

Le cas d'étude Faurecia concerne l'usine de Nogent-sur-Vernisson. Ce site emboutit, assemble et peint des armatures métalliques de sièges automobile. Le site est composé de deux ateliers, la tôlerie et l'assemblage. Le premier reçoit des bobines d'acier afin de les emboutir dans l'une des 6 presses que compte l'atelier avant de les expédier, pour la majorité à l'atelier assemblage et pour le reste à d'autres entités du groupe ou à des clients externes. Le second atelier, réalise l'assemblage des pièces métalliques grâce à des cellules de soudure à bobine, électrique ou laser, ainsi que des visseuses et riveteuses. Cet atelier fait l'objet de l'étude menée pour le Projet STAR. Le service logistique vient alimenter ces deux unités et récupère les produits finis destinés aux clients. Certaines armatures sont peintes puis les sièges sont expédiés par camions vers les clients internes, les usines JAT (Juste À Temps), qui se chargeront d'assembler tous les éléments du siège.

Le site est en cours de restructuration. L'ensemble de l'activité métal sera réorganisée de manière effective à l'horizon 2018. Le but de cette transformation ou modification est de simplifier les flux (physiques) internes afin d'améliorer les performances du système de production. Dans le cadre du Projet STAR, il faut donc considérer le cas d'étude suivant cette future configuration de l'atelier d'assemblage. La Figure 69 donne une vue globale de l'usine de Faurecia à Nogent-Sur-Vernisson, on peut distinguer : En entrée de l'usine, le stock de masse, en sortie de l'usine, la zone TPA, entre les deux, les postes de production.

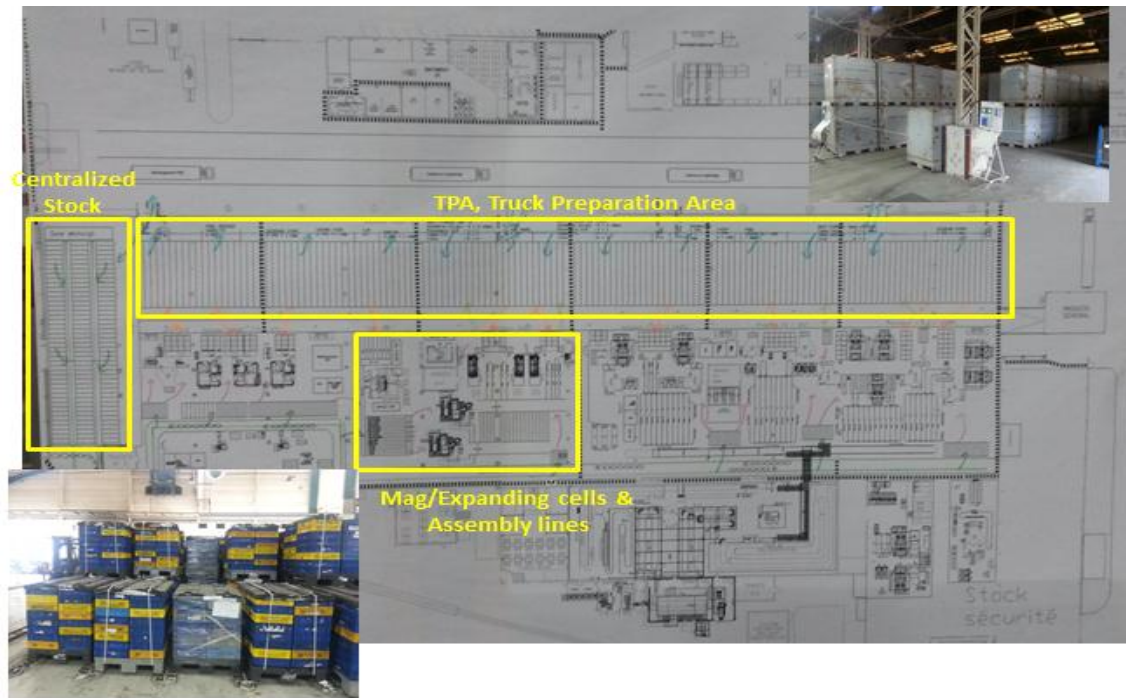


Figure 69 : Site de Faurecia Sièges Automobiles à Nogent-sur-Vernisson, horizon 2018.

On a décomposé le mouvement des produits dans l'usine en trois grandes étapes :

- Du stock de masse jusqu'à la zone de picking (Figure 70) : la réception des palettes, la mise en références des palettes, le dé-palettisation et mise en références des pièces.
- De la zone de picking jusqu'aux magasins décentralisés, un seul opérateur (Figure 70) : préparer la commande (picking), transporter les pièces, alimenter les magasins décentralisés.
- Du magasin décentralisé jusqu'à la zone TPA (Figure 71) : alimenter les postes pour la production (approvisionnement), production et préparation des palettes standards, transfert des palettes standards (contenant les produits finis) vers la zone TPA.



Figure 70 : flux (physique) : stock de masse → zone de picking, et Zone de picking → magasin décentralisé.

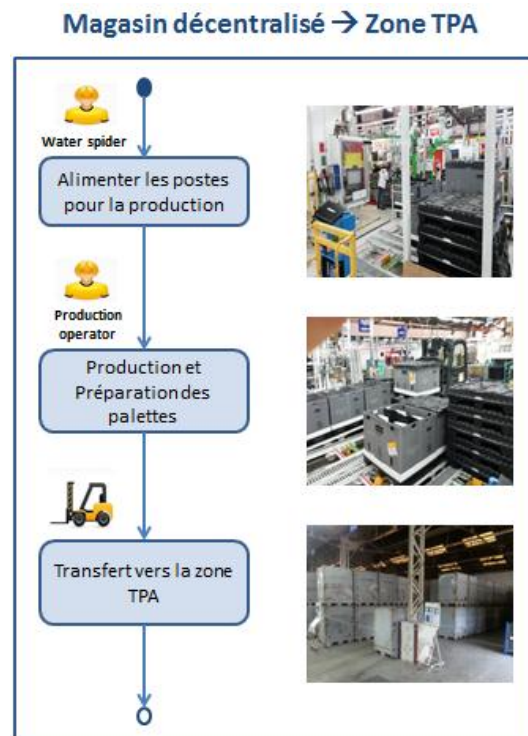


Figure 71 : flux (physique) : magasin décentralisé → zone TPA.

2.2. Cas d'étude N°02: Daher, Plate-forme logistique de Marignane

DAHER exploite une plate-forme pour différentes activités logistiques pour le compte d'Airbus Hélicoptères (réception, entreposage, préparation, colisage...) à Marignane. Ils cherchent à mettre en application une solution de chariot AGV pour des manutentions de charges internes au bâtiment ; plusieurs zones de déposes et de poses sont prévues dédiées à cette application. Le cas Daher faisant l'objet de l'étude dans le cadre du projet STAR est présenté dans la Figure 72.

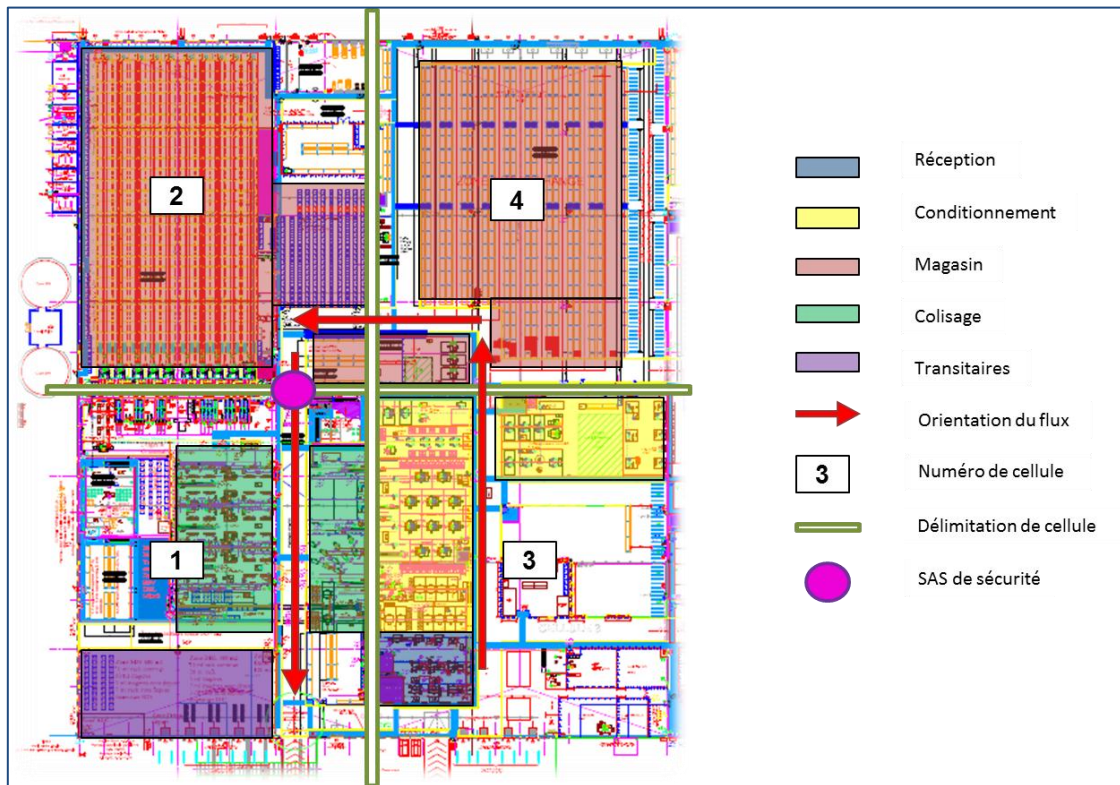


Figure 72 : Daher, plate-forme logistique Marignane, vue générale.

La Figure 73 donne un aperçu général des flux matériels dans la plate-forme Daher-Marignane. On distingue 3 zones principales qui échangent des flux matériels, ces flux sont généralement des chariots ou des palettes remplis de différents produits. Le processus du système logistique est constitué des étapes suivantes :

- Conditionnement : Les produits sont reçus de l'extérieur ; ils sont conditionnés dans la zone de conditionnement et envoyés vers la zone magasin pour être stockés dans l'entrepôt (le magasin).
- Stockage : Lorsque les produits arrivent à l'entrepôt, ils sont d'abord placés dans une zone d'attente appelée 'zone d'attente en entrée du magasin'. C'est une zone tampon pour les produits qui attendent d'être mis dans l'entrepôt.
- Déstockage : Lorsque les produits sont demandés par le client, ils sont recherchés dans l'entrepôt, et envoyés vers la zone de colisage pour être préparés et envoyés aux clients. Lorsque les produits

doivent être envoyés vers la zone de colisage, ils sont d'abord placés dans une zone d'attente appelée 'zone d'attente en sortie du magasin'. C'est comme une zone tampon pour les produits en attente d'être envoyés vers la zone de colisage.

- Colisage : Une fois la préparation des produits dans la zone de colisage terminée, ils seront expédiés vers le client.

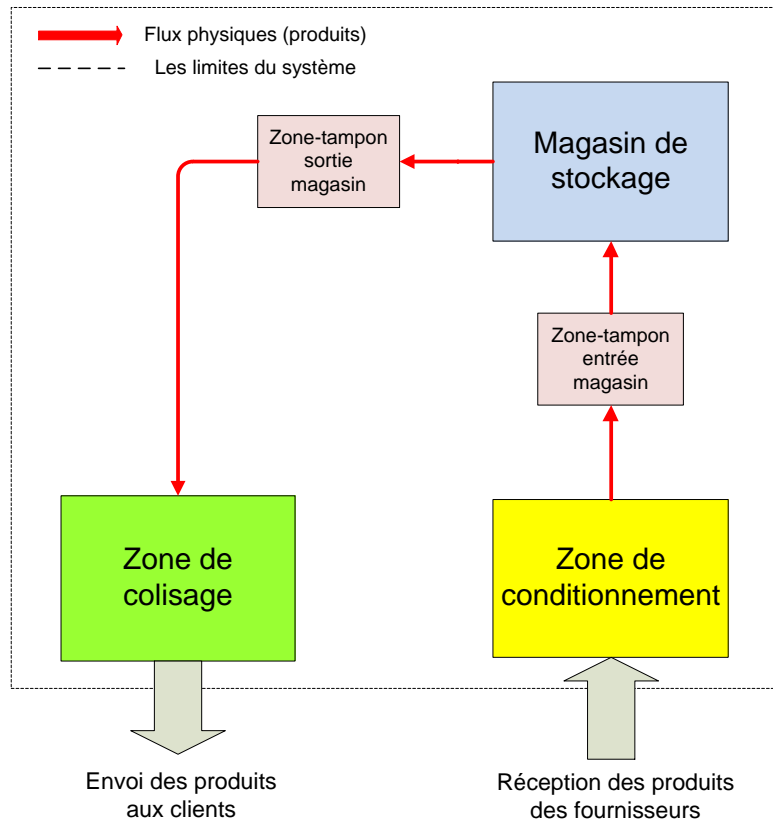


Figure 73 : Daher, plate-forme logistique Marignane, flux matériels

Dans les deux cas d'études (Faurecia et Daher), le transport des produits se fait actuellement à l'aide des engins (transpalettes, chariots élévateurs, petits trains, etc.) conduits par des opérateurs humains en utilisant différents supports ou conteneurs (chariots, palettes, bac, etc.). L'objectif des deux industriels (Faurecia et Daher) est d'automatiser cette fonction de transport afin de réduire les coûts de la production ou de service. Cela a conduit au lancement du projet STAR, dont l'objectif est de développer un système de transport automatisé et agile qui peut être déployé facilement sur les deux plates-formes (Faurecia et Daher) et éventuellement sur à d'autres plates-formes. Il doit aussi être reconfigurable et agile afin de faciliter son adaptation aux changements des plates-formes.

3. Application de la méthodologie

L'application de la méthodologie décrite dans le chapitre 4 a été supportée par deux outils logiciels :

- L'outil de gestion des exigences Polarion-ALM. La méthodologie a été implémentée sur cet outil, les grandes étapes ainsi que les différentes tâches, pour plus de détails (voir [Annexe 1](#)). Le rôle principal de Polarion-ALM est la gestion des exigences (collection, création des liens de traçabilité, analyse du changement, etc.), aussi, de guider les développeurs dans l'application de la méthodologie implémentée.
- L'outil Papyrus de modélisation des systèmes (logiciels UML, physiques SysML). Donc tous les digrammes SysML sont réalisés avec cet outil qui est un plug-in de l'environnement de développement intégré Eclipse.

3.1. Analyse du système à développer

Le but est de concevoir un système transport automatisé qui utilisera les AGVs comme moyen de transport. Donc, avant de commencer la conception de ce système, on sait qu'il sera composé de :

- Une flotte d'AGVs qui transporteront les conteneurs (chariot, palettes, ou autres) entre les différents points dans l'atelier.
- Le système de pilotage qui contrôlera l'ensemble ou la flotte d'AGVs.
- Le système de guidage/repérage des AGVs ou le support utilisé par les AGVs pour se localiser et se déplacer dans l'atelier, ça peut être (des bandes au sol, des étiquettes RFID au sol, des balises sur les murs, système GPS, etc.), les emplacements ou les points de prise et de dépose des chariots par l'AGV font partie du système de repérage.

Donc la conception du système globale équivaut à la conception de ces trois sous-systèmes (AGV, système de pilotage, et le système de repérage) sans oublier les interactions entre eux.

3.2. Processus : définition des besoins/exigences des parties prenantes

La Figure 74 montre les parties prenantes du système logistique. Parmi ces parties prenantes, on trouve : les opérateurs de production, de maintenance, les développeurs (Faurecia, Daher, BA-Systèmes, etc.), les fournisseurs (BA-Systèmes), etc. Pour le cas particulier des systèmes reconfigurables, on peut rajouter la catégorie des opérateurs de reconfiguration, c'est ceux qui s'occupe d'évaluer le système, d'en décider de reconfigurer le système ou non, d'en choisir la configuration future, de reconfigurer le système, et de valider la nouvelle configuration.

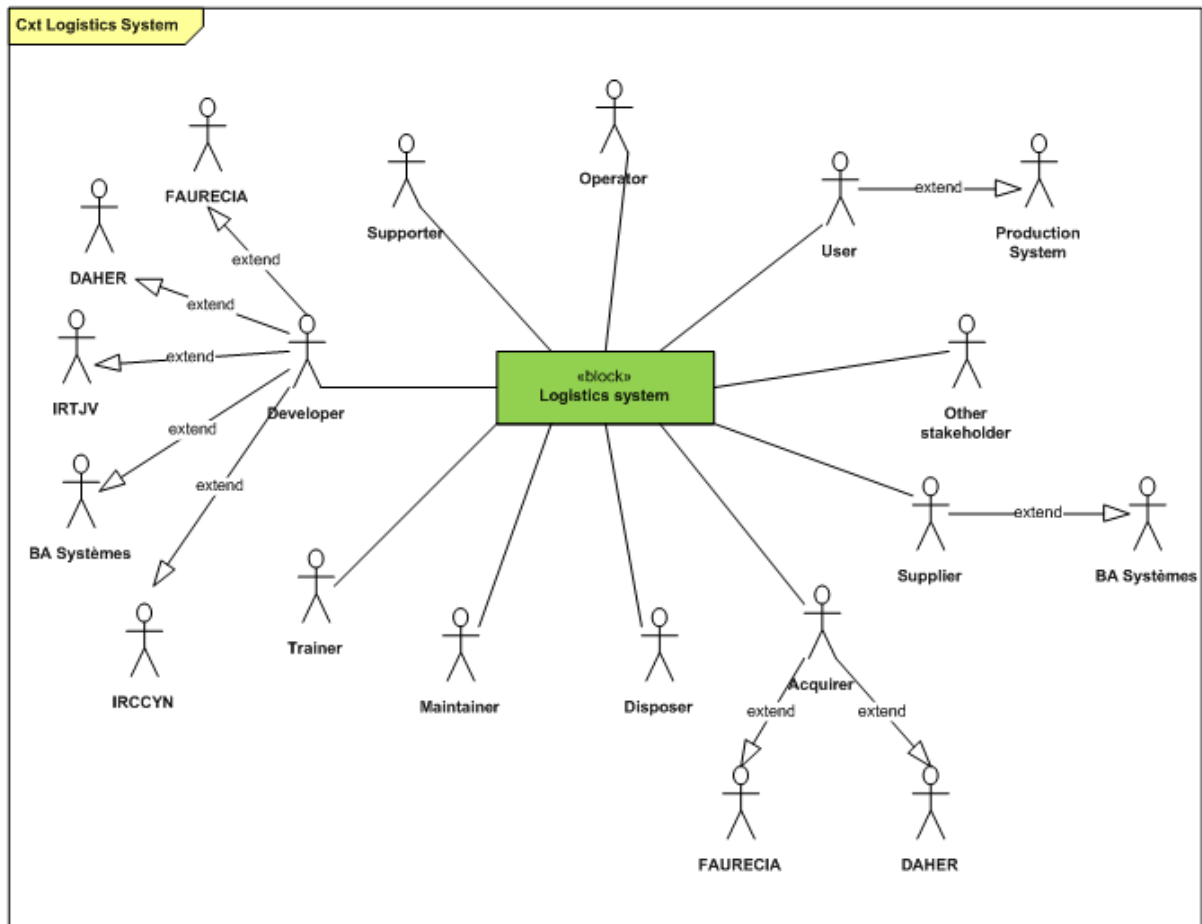


Figure 74 : les parties prenantes du système logistique.

Dans ces deux cas d'étude, il est plus facile de résoudre le problème de regroupement des produits en familles (voir chapitre 1, section 4.3). Pour absorber la variété entre les différents produits qui seront transportés par le système, on a pensé à utiliser un support standard qui pourra contenir tout type de produit (inclus dans les deux cas d'étude) ainsi que les palettes.

3.2.1. Sous-système : AGV

Les figures suivantes montrent, respectivement, un diagramme du contexte de l'AGV et un diagramme des cas d'utilisation de ce dernier. L'utilisation des diagrammes SysML facilite l'analyse du système et déduction des exigences.

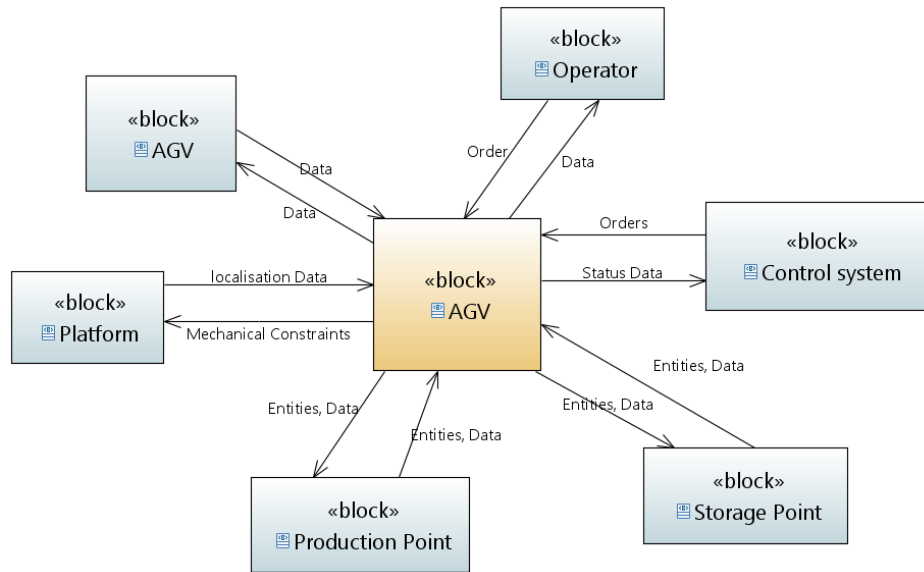


Figure 75 : AGV, diagramme du contexte.

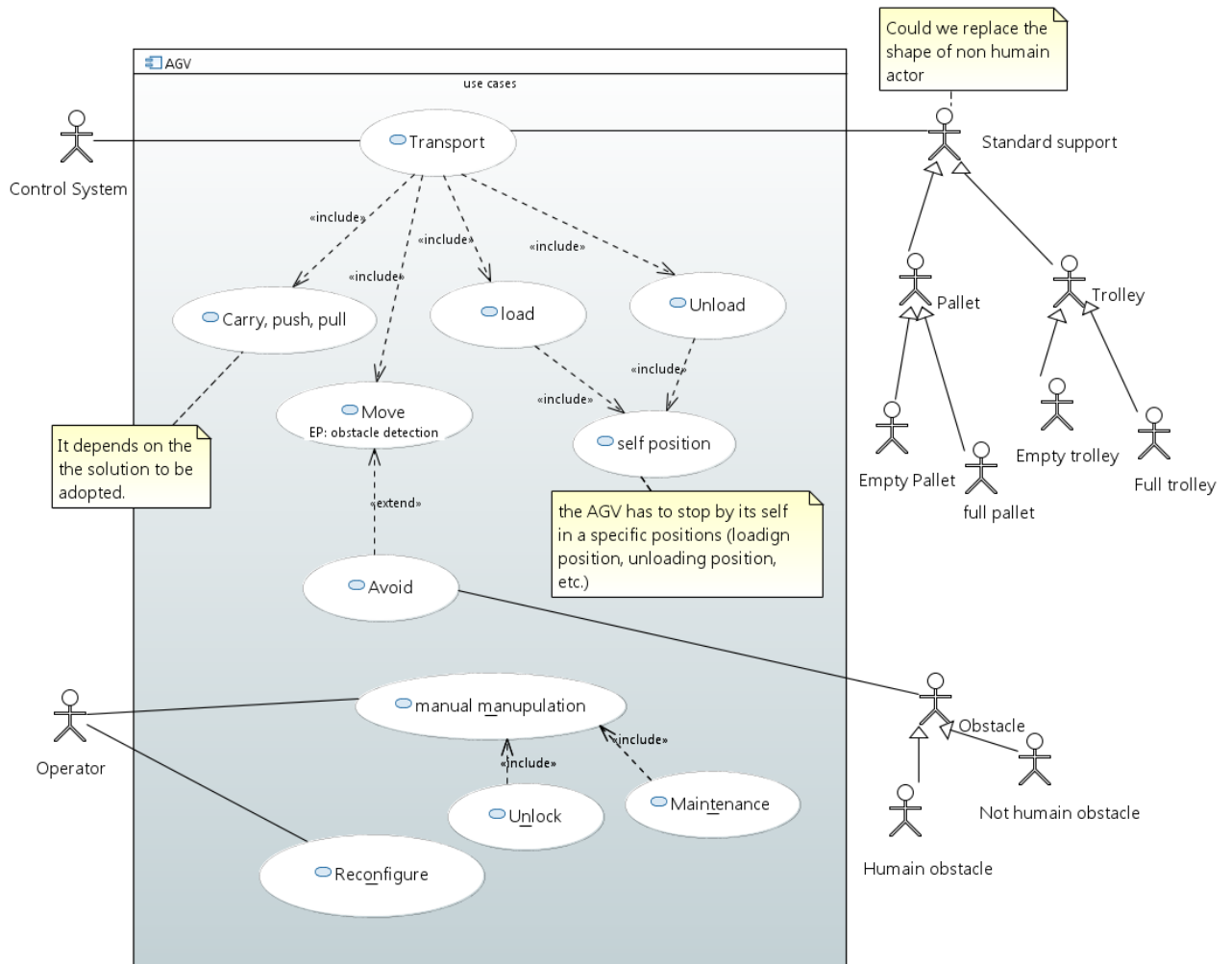


Figure 76 : AGV, diagramme des cas d'utilisation.

À partir de tous les diagrammes précédents, et à partir des plusieurs discussions avec les parties prenantes du système (principalement Faurecia et Daher), on a formulé plusieurs exigences. Le diagramme suivant est un exemple d'un diagramme d'exigences de l'AGV parmi l'ensemble des diagrammes qu'on a créés. Ces exigences sont intégrées, sous forme textuelle, dans Polarion pour faciliter leur gestion (vérification, analyse, traçabilité, etc.).

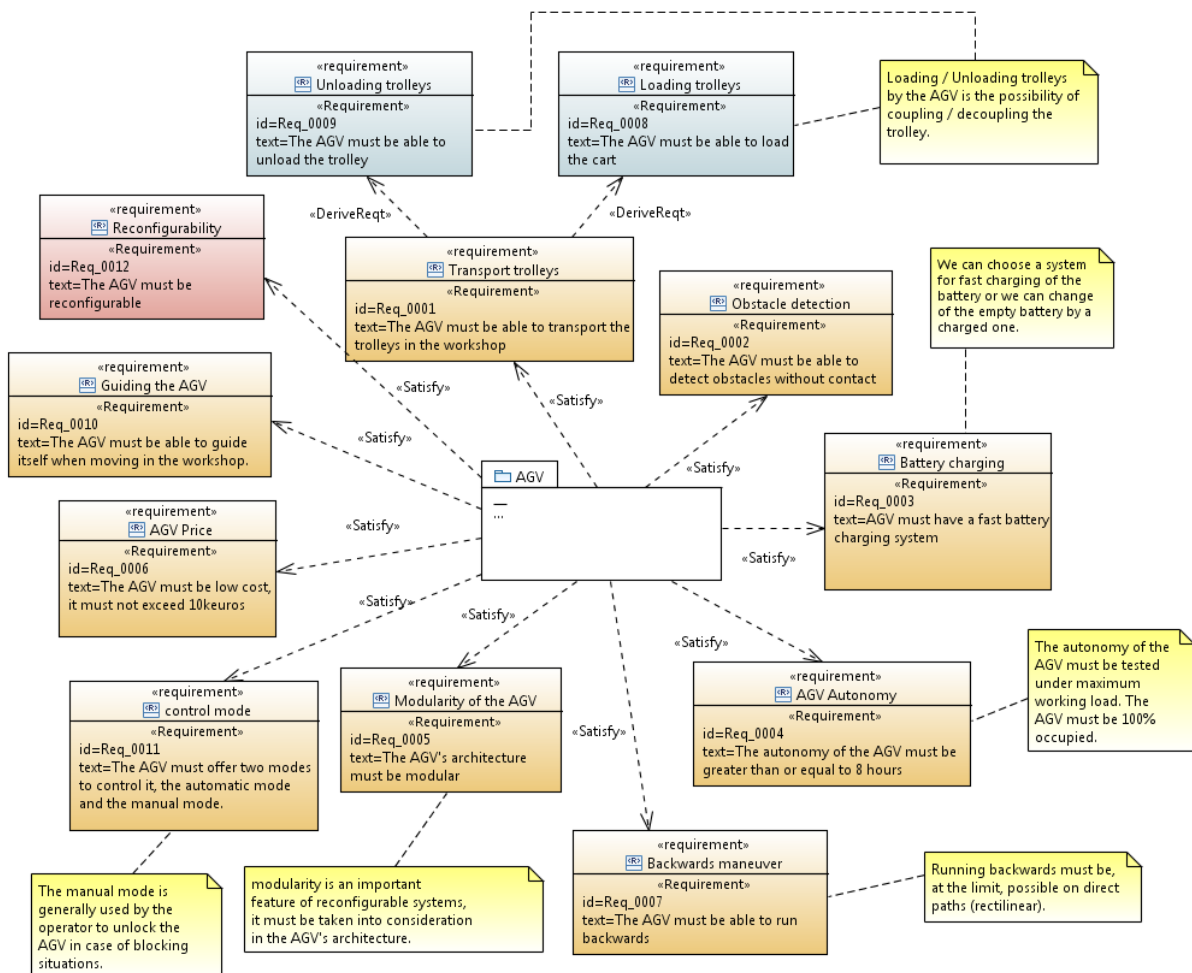


Figure 77 : AGV, exemple d'un diagramme d'exigences de l'AGV.

3.2.2. Sous-système : système de pilotage/contrôle

Les figures suivantes montrent, respectivement, un diagramme du contexte du système de pilotage et un diagramme des cas d'utilisation de ce dernier.

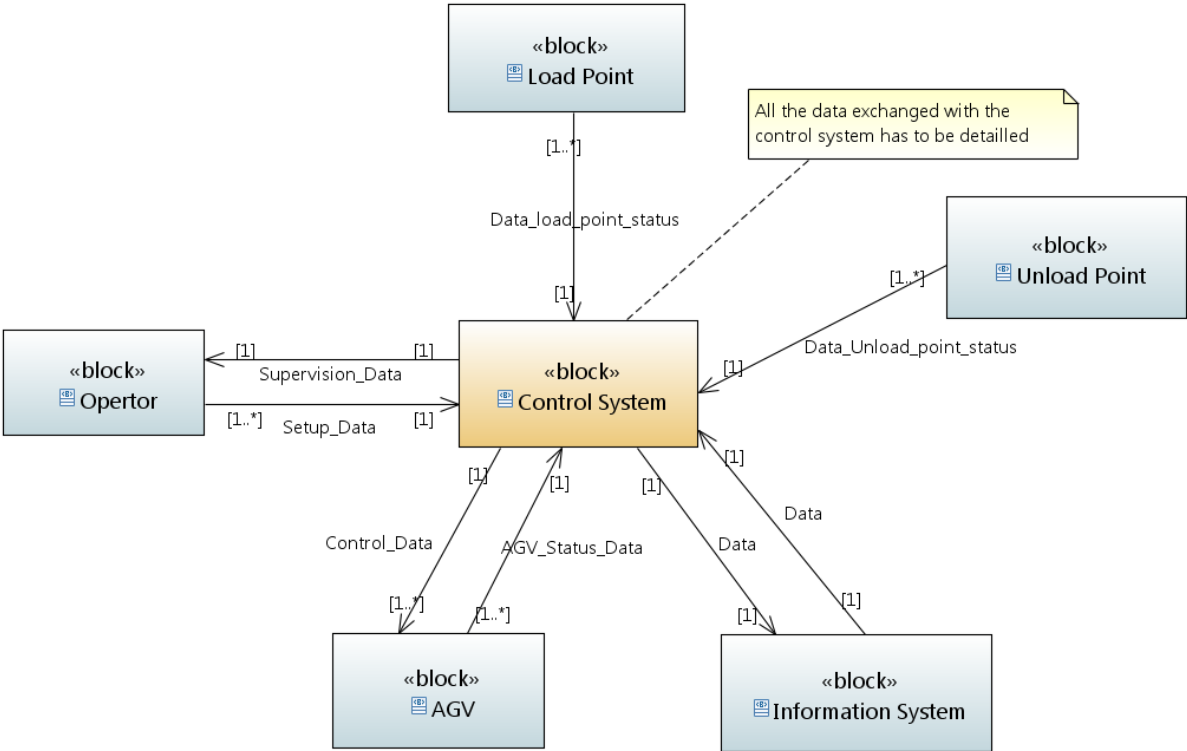


Figure 78 : Système de pilotage, diagramme de contexte du système de contrôle.

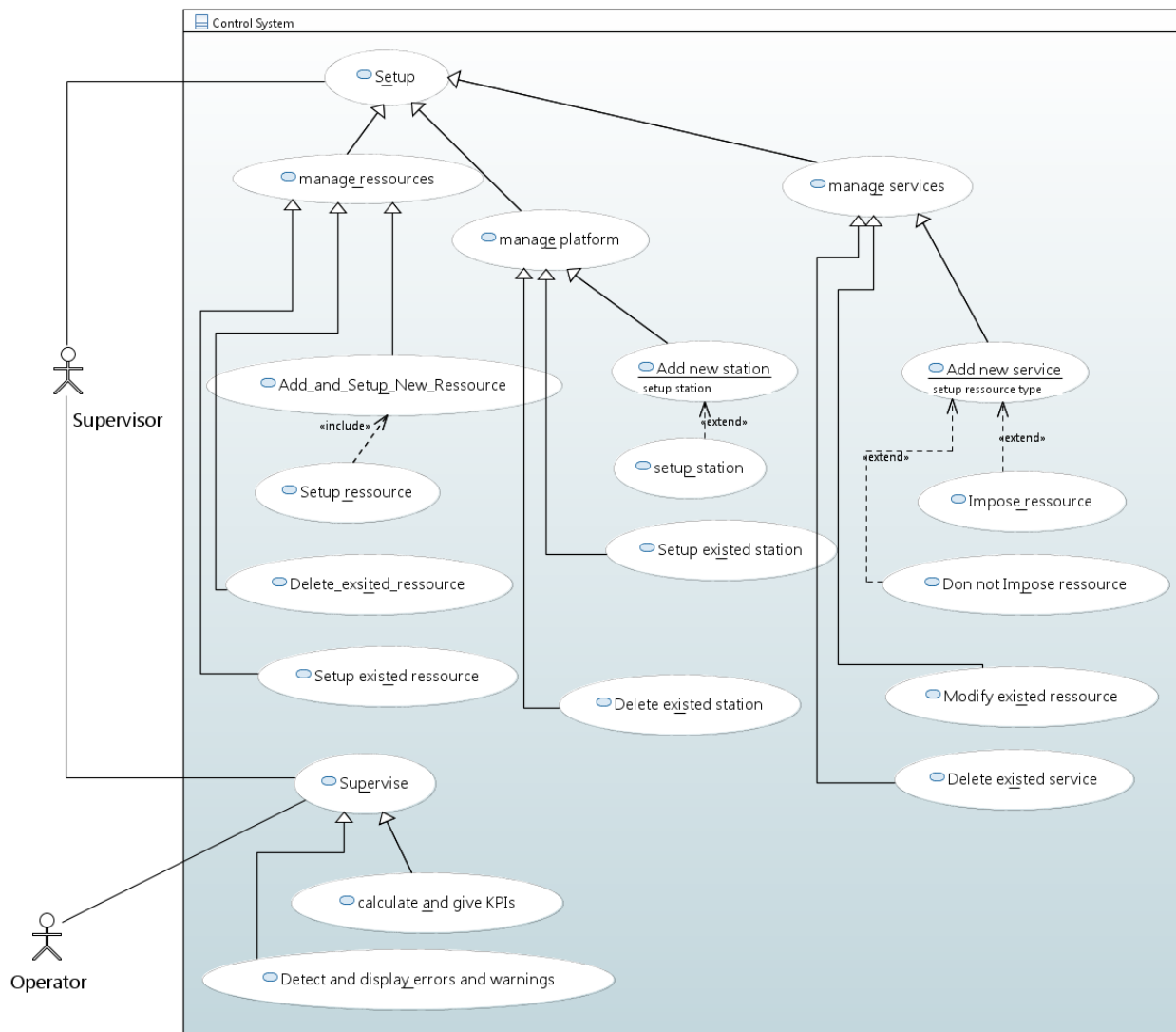


Figure 79 : système de pilotage, diagramme des cas d'utilisation.

A partir de tous les diagrammes précédents, et à partir des plusieurs discussions avec les parties prenantes du système, plusieurs exigences ont été formulées et des diagrammes d'exigences ont été créés. Ces exigences sont aussi intégrées, sous forme textuelle, dans le logiciel Polarion.

3.3. Processus spécification des exigences du système

D'autres diagrammes (diagrammes machine à états, diagrammes de séquences) ont été utilisés afin d'analyser plus en détail le système et de déduire plus d'exigences. Après l'analyse des diagrammes supplémentaires, plus d'exigences ont été rajoutées aux exigences qui ont été définies dans le processus précédent et plus de spécifications ont été apportées au système. Ces exigences et spécifications supplémentaires ont été aussi intégrées dans le logiciel Polarion.

3.3.1. Sous-système : AGV

Les figures suivantes montrent quelques exemples de diagrammes machine à état de l'AGV.

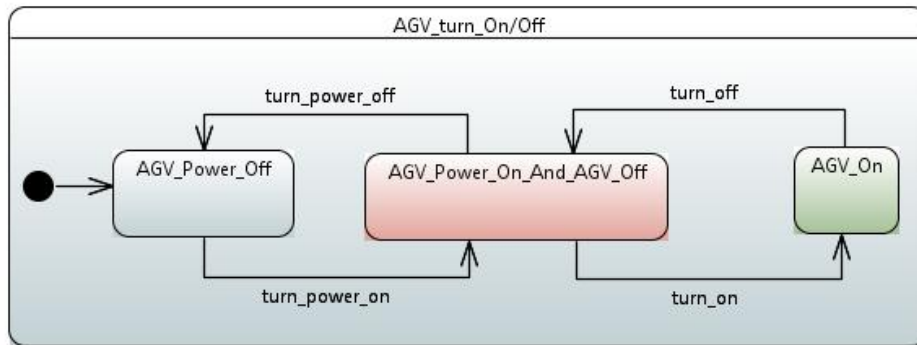


Figure 80 : AGV, diagramme machine à états, AGV On-Off.

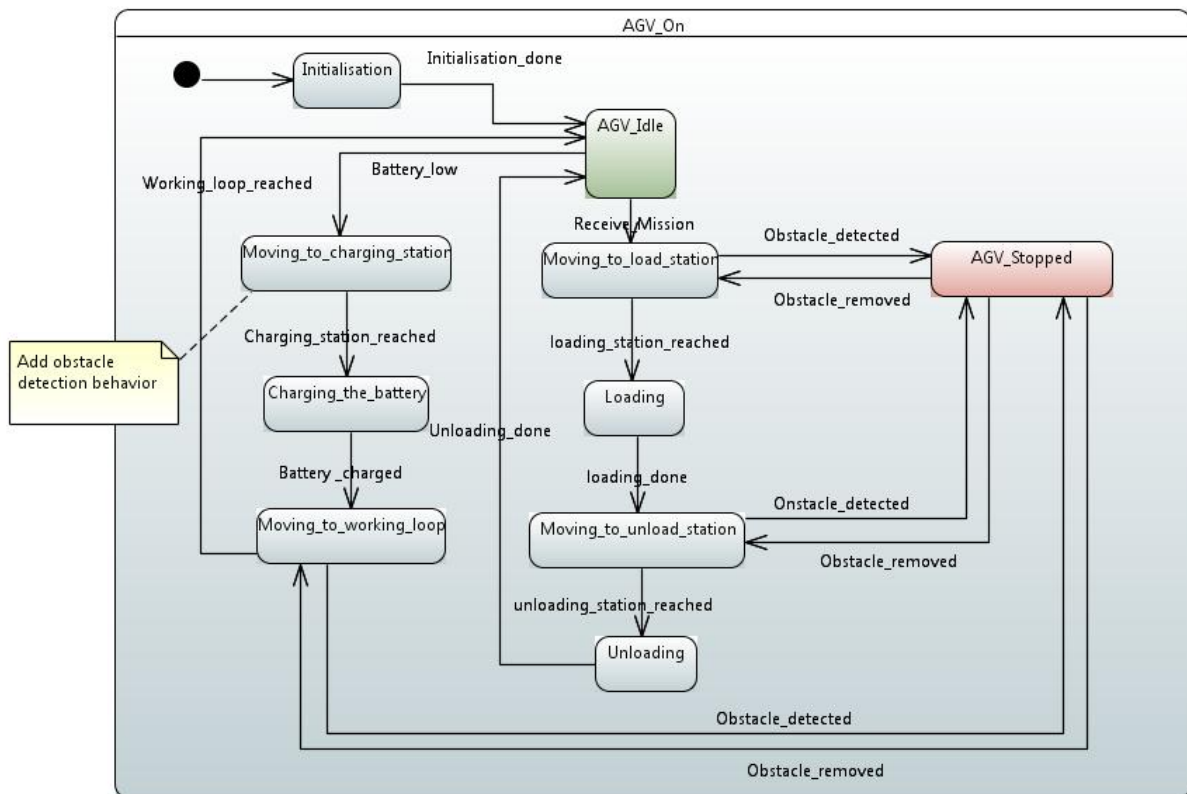


Figure 81 : AGV, diagramme machine à états, AGV On.

3.3.2. Sous-système : système de pilotage/contrôle

Les figures suivantes montrent le diagramme de séquence qui détaille les interactions entre le système de pilotage et l'AGV. Il montre :

- la séquence d'exécution d'une mission de transport par l'AGV.
- la séquence quand l'AGV sera envoyé pour recharger sa batterie quand c'est nécessaire.
- la séquence quand l'AGV sera envoyé vers un point de repos quand il y a plus de missions de transport à exécuter.

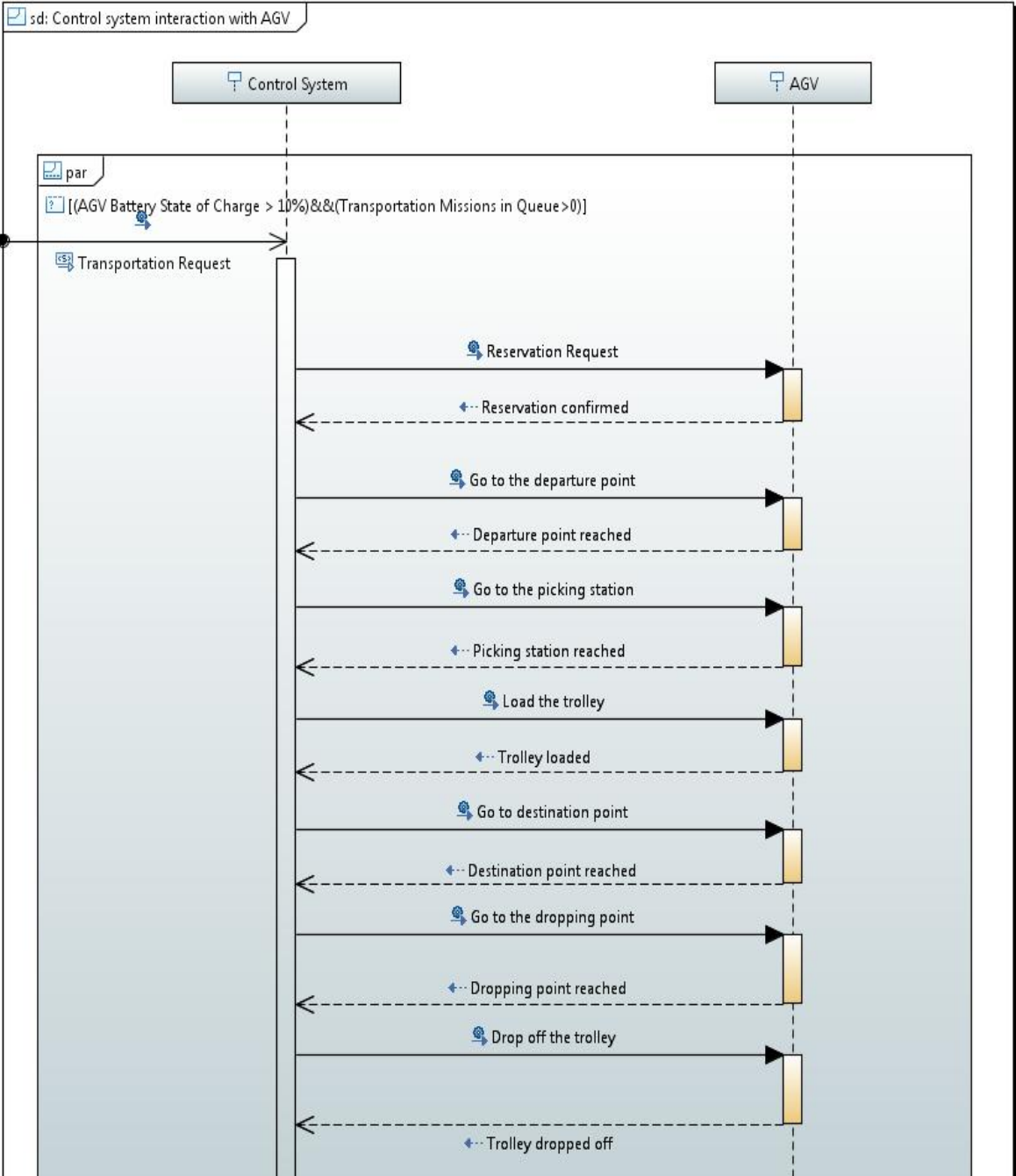


Figure 82 : Système de contrôle, interaction avec l'AGV.

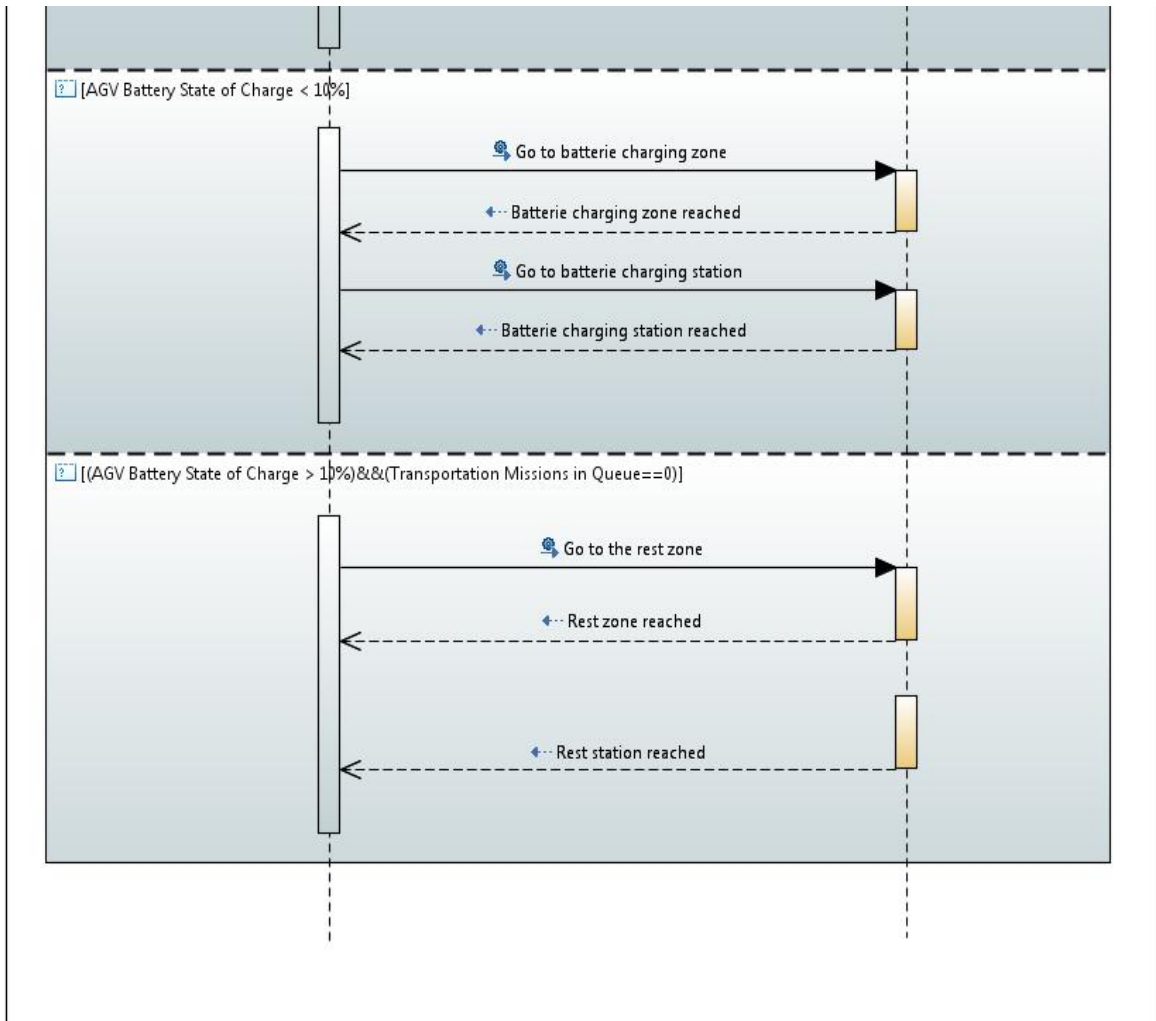


Figure 83 : Système de contrôle, interaction avec l'AGV (suite Figure 82).

Après l'analyse du système, on a identifié les déclencheurs de la reconfiguration suivants :

- ajout ou suppression d'un point de prise ou de dépose.
- augmentation du débit des flux matériels suite à une augmentation de la cadence de production par exemple.

3.4. Processus : conception de l'architecture, réalisation, et vérification

3.4.1. Sous-système : AGV

La conception, la réalisation et la vérification de l'AGV ont été réalisées par la société BA-Système. Nous nous sommes occupés de définir les besoins initiaux et spécifier une partie des exigences l'AGV.

3.4.2. Sous-Système : Système de contrôle

Concernant le système de pilotage, son développement (conception de l'architecture, implémentation et vérification) a été réalisé par un autre ingénieur sur le projet STAR. Nous nous

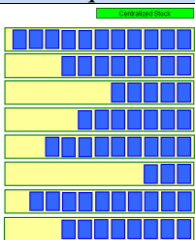


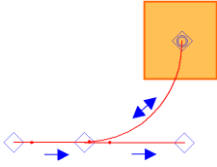
sommes occupés de la définition des exigences de ce système. L'ingénieur s'est basé sur une architecture multi-agents pour le développement du système de pilotage.

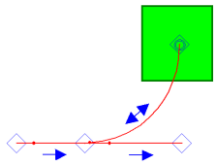
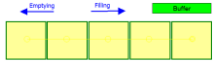
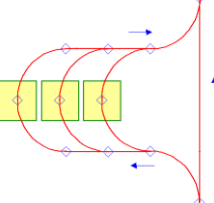
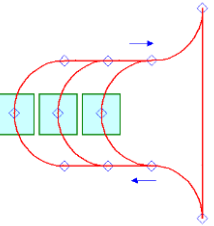
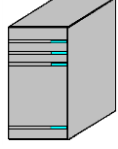

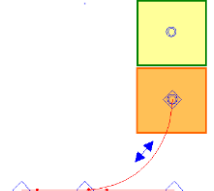
3.5. Processus validation du système

Pour la validation du système logistique complet (AGV + système de pilotage + système de repérage), un prototype a été déployé sur la plate-forme logistique de Daher à Marignane. Mais, suite à des problèmes techniques, le système n'a pas pu être validé. D'autres tests de validation du système vont être programmés dans un futur proche. On compte publier les résultats de ces tests dans nos futurs travaux.

4. La bibliothèque virtuelle

Parmi les objectifs de cette thèse, nous devons développer un outil de simulation de flux qui sera utilisé pour aider à la prise de décision concernant les configurations futures du système principalement. La description de la démarche suivie pour la création de cet outil est décrite dans le chapitre 5. En générale, l'outil a été développé avec le logiciel ARENA. Cet outil est une bibliothèque virtuelle qui rassemble tous les modules (virtuels) de base qu'on a pu identifier et qui constituent le système. La bibliothèque peut être intégrée facilement dans le logiciel ARENA. Le tableau suivant résume tous les modules de cette bibliothèque.

Nome	La forme dans la maquette	Description	KPIs
Stock de masse (propre à Faurecia)		Le stock de masse contient toutes les pièces nécessaires pour la production. Il sert à approvisionner tous les stocks décentralisés ou déportés dans l'usine. Dans ce stock, les pièces sont regroupées selon leurs références (voir à côté).	Pas de KPIs.
Station de travail		La station de travail sert à modéliser les postes de travail au niveau desquels les tâches, remplir un chariot vide, vider un chariot plein, transférer un chariot, seront exécutées.	<ul style="list-style-type: none"> • Temps d'attente dans la station • Temps d'exécution de la tâche
Opérateur		Le module opérateur sert à modéliser les opérateurs travaillant dans l'usine comme les préparateurs de commandes par exemple.	Taux d'occupation
Station de départ		Cette station sert à modéliser les endroits géographiques dans l'usine qui servent à maintenir les chariots provisoirement en attendant d'être transportés par un AGV ou tout autre moyen de transport.	Temps d'attente dans la station

<p>Station de réception</p>		<p>Cette station sert à modéliser les endroits géographiques dans l'usine qui servent à recevoir les chariots transportés par un AGV. Ces chariots doivent être transportés ailleurs afin de libérer la station pour recevoir les prochains chariots.</p>	<ul style="list-style-type: none"> • Temps d'attente dans la station • Temps d'approvisionnement
<p>Buffer</p>		<p>Ce buffer sert à modéliser les zones tampons utilisées pour stocker, généralement, les chariots vides, ou tout autre type d'entités.</p>	<p>Temps d'attente dans la station</p>
<p>Zone de repos des AGVs</p>		<p>Ce module sert à modéliser la zone géographique réservée aux AGVs pour mise en attente (repos) en attendant de nouvelles instructions.</p>	<p>Temps d'attente dans la zone</p>
<p>Zone de chargement de batterie de l'AGV</p>		<p>Ce module sert à modéliser la zone géographique réservée aux AGVs pour le chargement de leurs batteries.</p>	<p>Temps d'attente dans la zone</p>
<p>Système de pilotage</p>		<p>Ce module sert à modéliser le système de contrôle qui pilote la flotte d'AGVs, gère les demandes de transport et les demandes d'approvisionnement.</p>	<ul style="list-style-type: none"> • Temps d'exécution d'une mission de transport. • Temps d'exécution d'une demande d'approvisionnement.
<p>La flotte d'AGVs</p>		<p>Ce module sert à modéliser la flotte d'AGVs utilisés pour transporter les chariots (vides ou pleins) entre les différents endroits dans l'usine.</p>	<ul style="list-style-type: none"> • Distance parcourue à vide • Distance parcourue chargé avec un support vide ou chargé • Distance totale parcourue à vide ou chargé • Temps d'attente en état de blocage
<p>Un stock déporté (propre à Faurecia)</p>		<p>Ce module sert à modéliser les stocks déportés. Chaque stock est constitué de deux stations : une station d'approvisionnement, et une station de consommation.</p>	<ul style="list-style-type: none"> • Temps d'approvisionnement • Temps ne d'attente dans le stock.

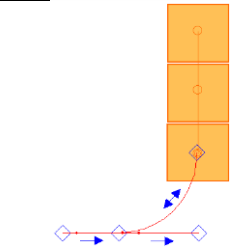
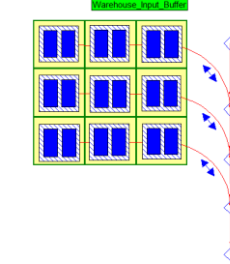
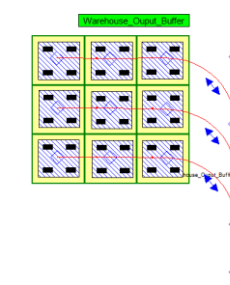
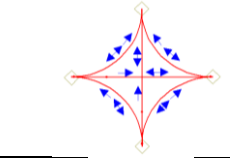
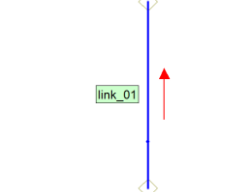
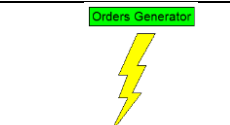
Buffer déporté (propre à Faurecia)		Ce module sert à modéliser un buffer déporté. Ce buffer est utilisé pour stocker les chariots vides qui attendent d'être transportés pour être réutilisés.	Temps ne d'attente dans le stock.
Buffer entrée magasin (propre à Daher)		Ce buffer retiendra provisoirement les chariots ou généralement les supports de transport chargés qui arriveront de la zone de conditionnement avant d'être traités et stockés dans le magasin.	Temps d'attente dans le buffer
Buffer sortie magasin (propre à Daher)		Ce buffer retiendra provisoirement les chariots ou généralement les supports de transport vides qui arriveront du 'buffer entrée magasin' après le vidage ou ailleurs dans le système avant de les envoyer vers la zone de conditionnement ou vers une autre destination pour être réutilisés.	Temps d'attente dans le buffer
Intersection		Ce module sert à modéliser les intersections qui font partie du circuit utilisé par les AGVs pour se guider et se déplacer dans l'atelier.	Pas de KPIs.
Lien		Ce module sert à modéliser les liens qui constituent le circuit utilisé par les AGVs pour se guider et se déplacer dans l'atelier.	Pas de KPIs.
Générateur d'ordres		Ce module sert à générer des ordres comme les ordres de travail.	Pas de KPIs.

Tableau 14 : les modules de la bibliothèque virtuelle.

Les KPIs listés dans le tableau ci-dessus sont très importants pour l'évaluation des différentes configurations du système logistique.

4.1. Maquette numérique

Grâce à la bibliothèque virtuelle, deux maquettes numériques complètes ont été développées pour les deux cas d'étude (Faurecia Daher). La construction de ces maquettes a été rapide et facile, et on a eu un retour positif de la part des industriels concernant l'utilisation de la bibliothèque.

4.1.1. Maquette numérique Faurecia

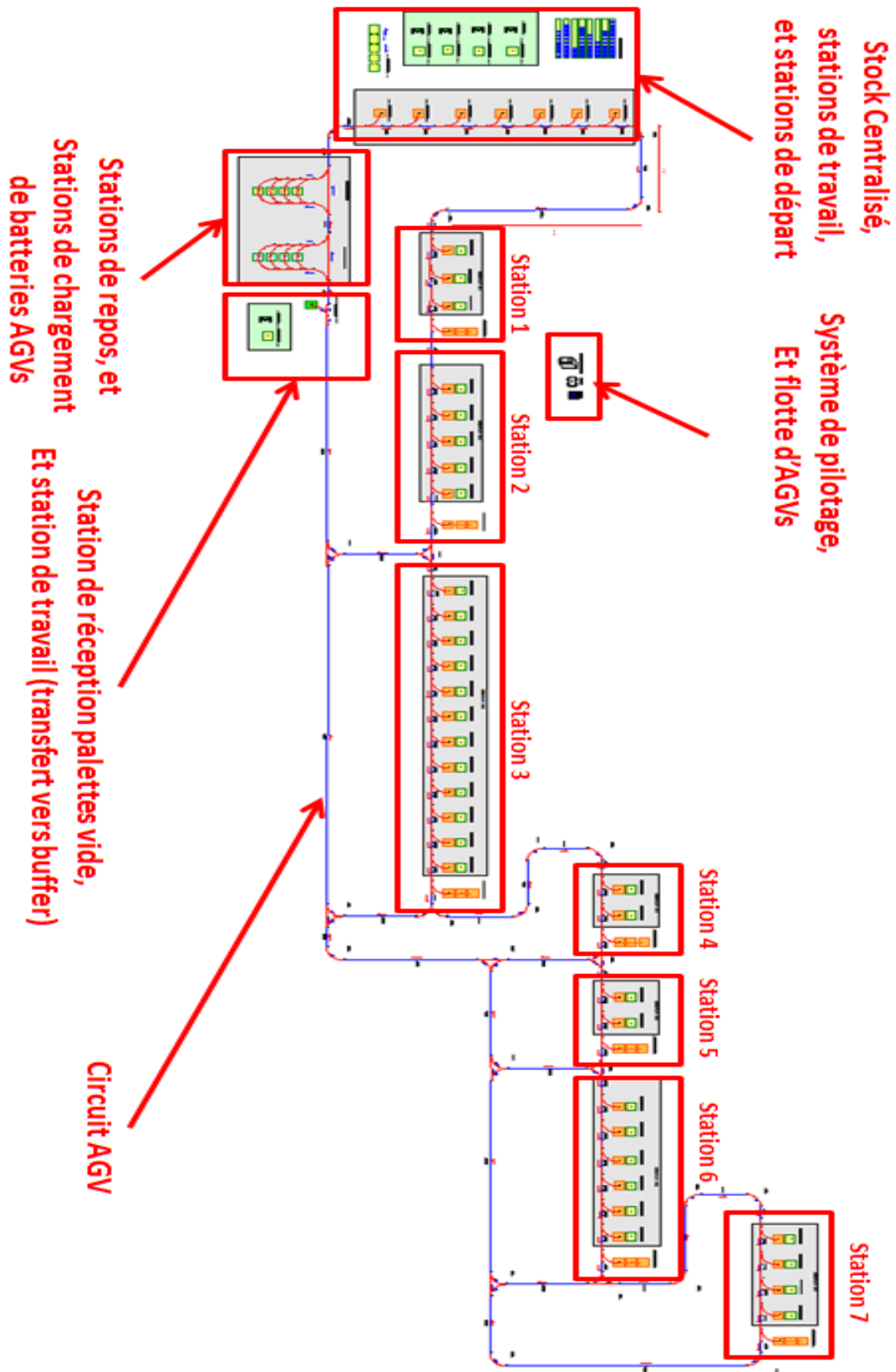


Figure 84 : la maquette numérique de Faurecia, usine Nogent-Sur-Vernisson.

4.1.2. Maquette numérique Daher

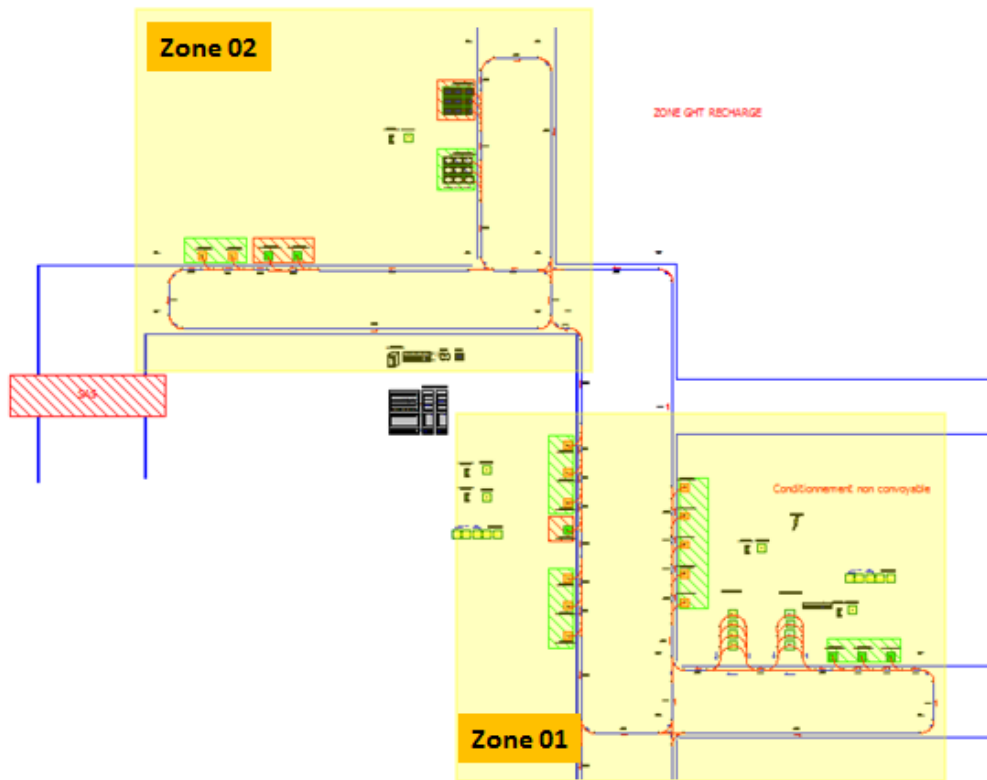


Figure 85 : la maquette numérique de Daher, plate-forme logistique Marignane.

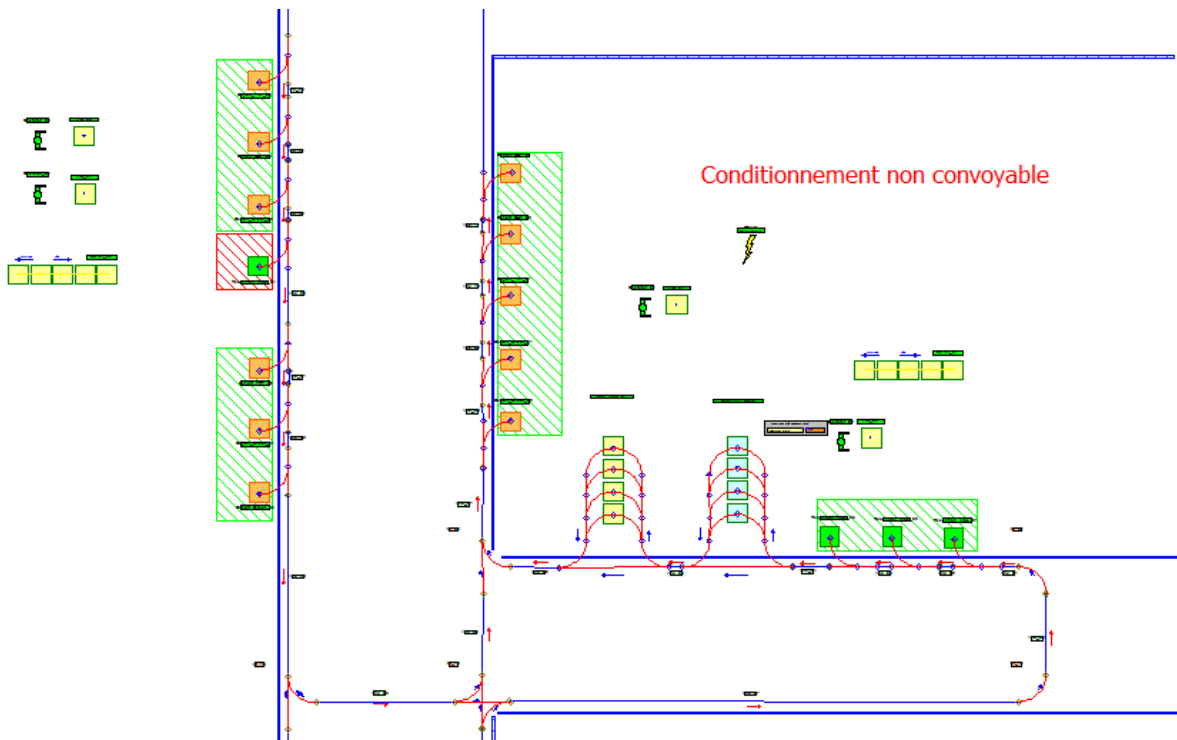


Figure 86 : la maquette numérique de Daher, suite (zoom sur la zone 1, Figure 85).

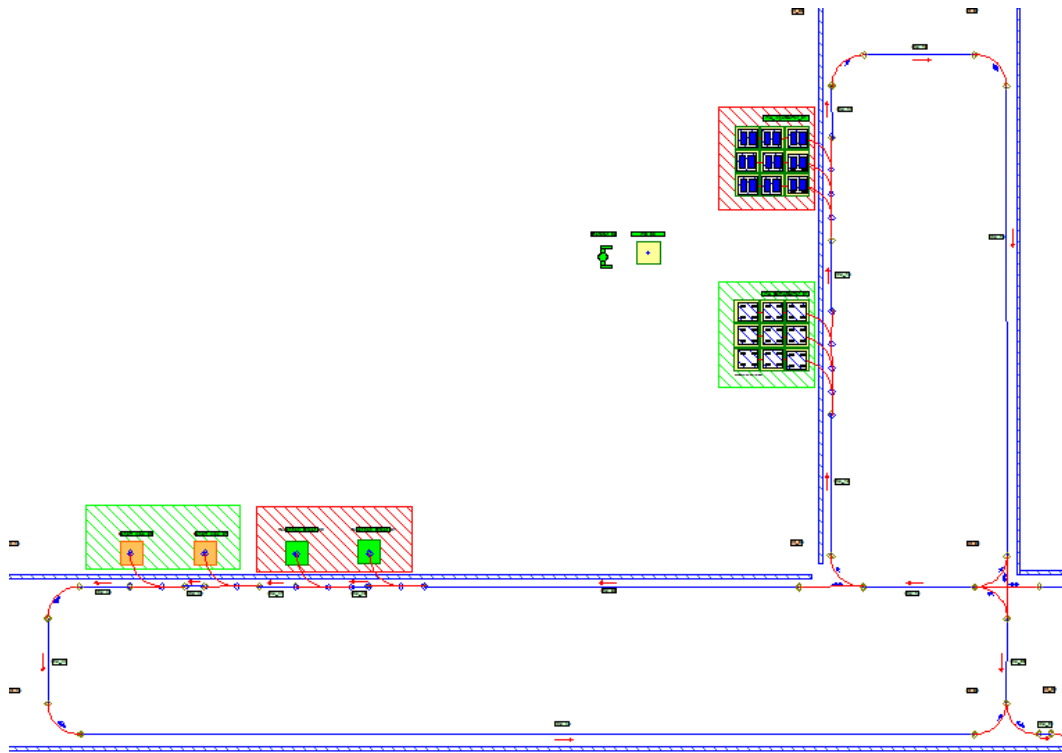


Figure 87 : la maquette numérique de Daher, suite (zoom sur la zone 2, Figure 85).

5. Les résultats de la simulation

On présente ici quelques résultats de la simulation du cas d'étude de Faurecia. Ces résultats concernent la configuration suivante : 3 AGVs avec une autonomie batterie infinie.

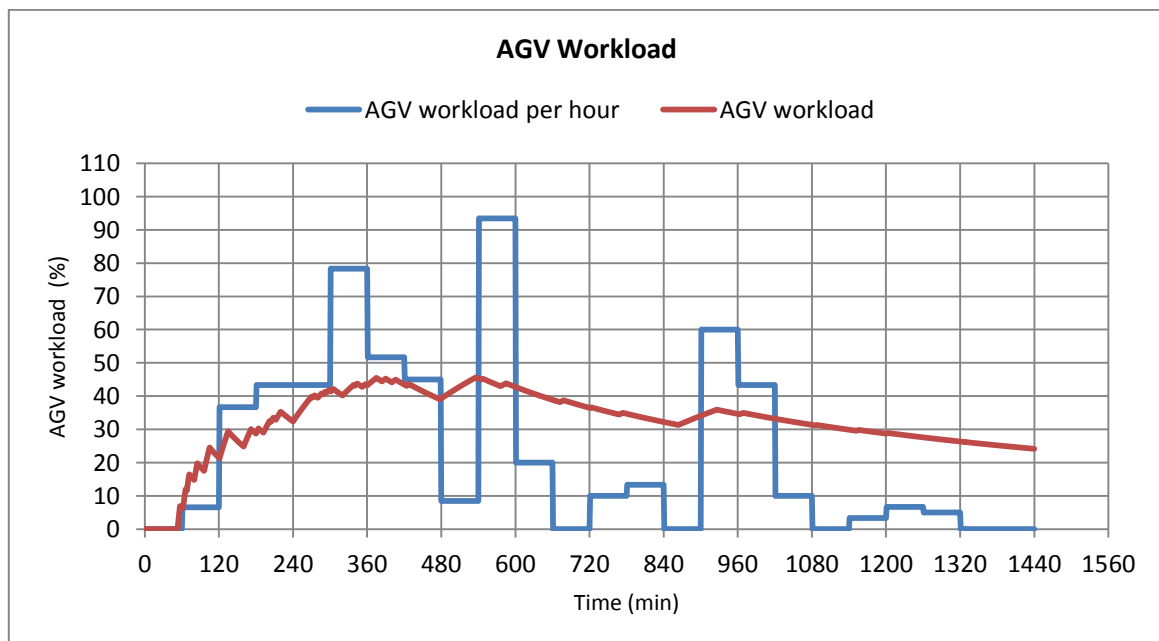


Figure 88 : la charge de travail sur l'AGV N° 01.

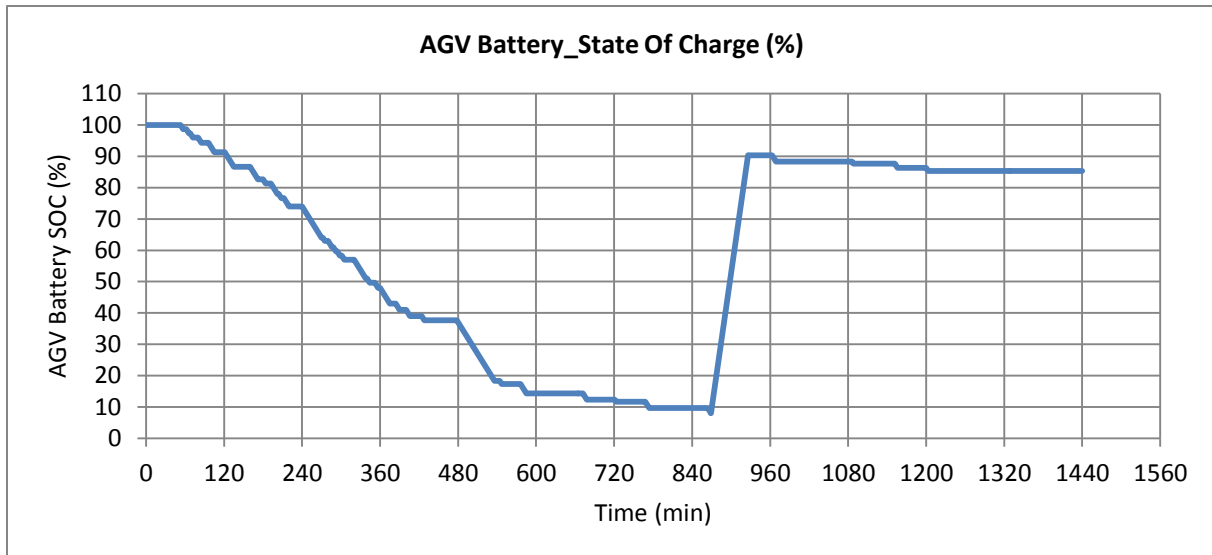


Figure 89 : l'état de charge de la batterie de l'AGV N° 01.

6. Conclusion

Dans ce chapitre, on a décrit les deux cas d'étude sur lesquels la méthodologie, présentée dans le chapitre 4, a été appliquée. Un prototype du système a été déployé, mais malheureusement, suite à des problèmes techniques, il n'a pas pu être validé. Aussi, dans ce chapitre, on a présenté l'outil d'aide à la décision qui a été développé. Cet outil est une bibliothèque virtuelle qui regroupe tous les modules standards du système logistique. Des maquettes numériques des deux cas d'étude ont été construites facilement. Donc l'outil peut être utilisé pendant la phase opérationnelle du système reconfigurable pour aider à l'évaluation et au choix des configurations futures du système logistique.

Conclusion générale et travaux futurs

Les RMS sont les systèmes de production les plus intéressants pour les entreprises manufacturières pour rester compétitive et pour survivre dans le marché actuel qu'est caractérisé par un très haut niveau de compétitivité. En raison de leur structure changeante (physique et logique) à différents niveaux de composition, les RMSs sont des systèmes complexes et leur conception n'est pas une tâche facile. Par conséquent, la plupart des méthodes et techniques utilisées, pour concevoir les systèmes complexes, pourraient être appliquées aux RMSs avec certaines modifications, adaptations et améliorations. Ainsi, sur la base de cette idée, une méthodologie globale basée sur la norme ISO/IEC/IEEE-15288 pour aider à concevoir les RMSs a été proposée dans cette thèse. Cette méthodologie aide les développeurs dans leurs travaux de la définition des besoins des PPs jusqu'au déploiement et la validation du système qui répond à ces besoins. Elle met en avant les évolutions nécessaires de certains concepts de l'ingénierie des systèmes pour prendre en compte la spécificité des systèmes reconfigurables.

La méthodologie proposée implémente plusieurs processus techniques de la norme qui sont adaptés au domaine des systèmes reconfigurables. Elle adopte une version modifiée du cycle en « V ». À chaque fois que la reconfiguration du système est nécessaire, une demande de reconfiguration est générée. Cette demande déclenchera la reprise de la méthodologie pour concevoir une nouvelle configuration du système qui répond aux exigences du nouveau contexte. La méthodologie nécessite l'intégration d'autres activités et tâches spécifiques au domaine des RMSs. En outre, il manque des techniques, des méthodes et des outils pour supporter les différentes activités déjà implémentées. L'objectif ultime est de développer une méthodologie basée sur principe de l'ingénierie des systèmes où les différentes tâches et activités sont réalisées en utilisant principalement des modèles.

La modularité est l'une des techniques utilisées pour gérer la complexité d'un système le long de son cycle de vie (phase de conception, phase de production, phase d'utilisation et même phase de démantèlement et de retrait de service). On a présenté une méthode basée sur l'outil DSM (Design Structured Matrix) qui est un outil déjà utilisé pour concevoir des produits. Nous avons, montré que cette méthode pourrait être utilisée pour concevoir un système de production modulaire ; elle pourrait être appliquée sur l'ensemble du système ainsi que sur ses sous-systèmes.

Dans cette thèse, on a aussi présenté le rôle de la simulation dans la conception des systèmes manufacturiers reconfigurables. On a montré l'importance de la considérer comme une étape incontournable pendant la définition, la spécification et la conception du système. Nous avons montré également comment elle peut aider pour choisir et valider une configuration quand la reconfiguration du système est nécessaire. Cette tâche sera aussi facilitée par la mise en place d'une bibliothèque de modules virtuels facile à utiliser pour la construction du modèle globale du système et ses différentes

variantes (les configurations du système). Cela est très utile une fois le système est opérationnel parce que le passage d'une configuration à une autre nécessite forcément la validation de la configuration future du système avant de la déployer réellement sur le terrain. Donc, à l'aide de la simulation et la bibliothèque mise en place, l'étape de choisir et de valider la configuration future du système sera réduite en termes de temps et d'effort. Les modèles des différentes configurations possibles seront facilement construits, testés, évalués, comparés et la configuration répondant le mieux aux besoins sera sélectionnée pour être déployée.

On prévoit d'aller plus dans l'utilisation de la simulation pour la conception et la gestion des RMSs. L'objectif est de pouvoir générer automatiquement les modèles de simulation à partir des modèles graphiques qui décrivent le système, cela facilitera la construction et la comparaison des différentes configurations possibles du système, et la personne chargée de tester et valider ces configurations n'aura même pas besoin d'avoir des connaissances approfondies dans la simulation de flux pour construire ces modèles. Parmi les objectifs finaux est la mise en place d'un système de pilotage modulaire reprenant les objets de la bibliothèque de simulation et de le coupler avec le modèle de simulation pour plus d'assistance et d'aide à la décision lors des phases de production (voir figure ci-dessous).

Annexe 01 : Implémentation de la méthodologie sur Polarion

1. Introduction

Comme toujours, un outil est indispensable pour mettre en pratique les idées. La gestion d'exigences n'y échappe pas. Le besoin de gérer des exigences avec l'aide d'un outil se fait sentir très rapidement, souvent dès que les premiers changements arrivent. On se penche alors naturellement vers des solutions économiques à portée de main, par exemple MS-Word ou MS-Excel. Ces solutions semblent a priori donner satisfaction si l'on se contente de peu, car elles ne nécessitent pas de formation, sont largement diffusées et ne sont pas très coûteuses. Dès qu'il s'agit de gérer dans le temps un grand nombre d'exigences, les problèmes liés à la technologie de ces outils deviennent prohibitifs. Entre autres : format des exigences non standardisé, non normalisé, accès à l'information difficile, car répartie sur plusieurs systèmes, pas de traçabilité, pas de suivi de l'évolution, données non sécurisées, pas de travail collaboratif en temps réel, etc.

En ingénierie des systèmes, la gestion des exigences et de leurs évolutions sur tout le cycle de vie d'un projet permet de livrer des produits et des solutions conformes aux attentes des utilisateurs, dans les délais impartis. Les outils classiques de gestion des exigences ne permettent pas de supporter le développement des systèmes complexes tels que les systèmes reconfigurables qui présentent un haut niveau de complexité. Donc, l'utilisation d'un outil de gestion des exigences permet de faciliter le développement des systèmes reconfigurables. Après une étude comparative de quelques outils de gestion des exigences disponible sur le marché, on a choisi le logiciel Polarion-ALM. Les outils sont évalués et comparés par rapport aux fonctionnalités requises et par rapport aux coûts. Quelques critères de sélection d'un outil de gestion des exigences peuvent être trouvés dans les travaux suivants : (voir (Bokhari & Siddiqui 2010), (Alghazzawi et al. 2014), (Abma 2009), (Siddiqui & Bokhari 2013)).

2. Polarion

Le logiciel Polarion a été notre choix pour le projet STAR, cela est justifié par : le logiciel respecte tous les critères choisis, le budget prévu ne sera pas dépassé, le logiciel est fiable, il a été testé et approuvé par plusieurs grands industriels dans plusieurs domaines (General Electric, US Air Force, SIEMENS, BOMBARDIER, FESTO, etc.), le logiciel Polarion choisi est un ALM (Application Life cycle Management), donc il ne fait pas que la gestion des exigences, mais il fournit toutes les fonctionnalités ALM avec leurs bénéfices sur une plate-forme collaborative unifiée, le logiciel existe sur le marché depuis 2004, donc une expérience dans le domaine de la gestion des exigences est acquise pendant la période (2004-2015).

3. Implémentation de la méthodologie sur Polarion

3.1. Page d'accueil de la méthodologie

La figure suivante montre l'implémentation, page d'accueil, de la méthodologie sur l'outil Polarion.

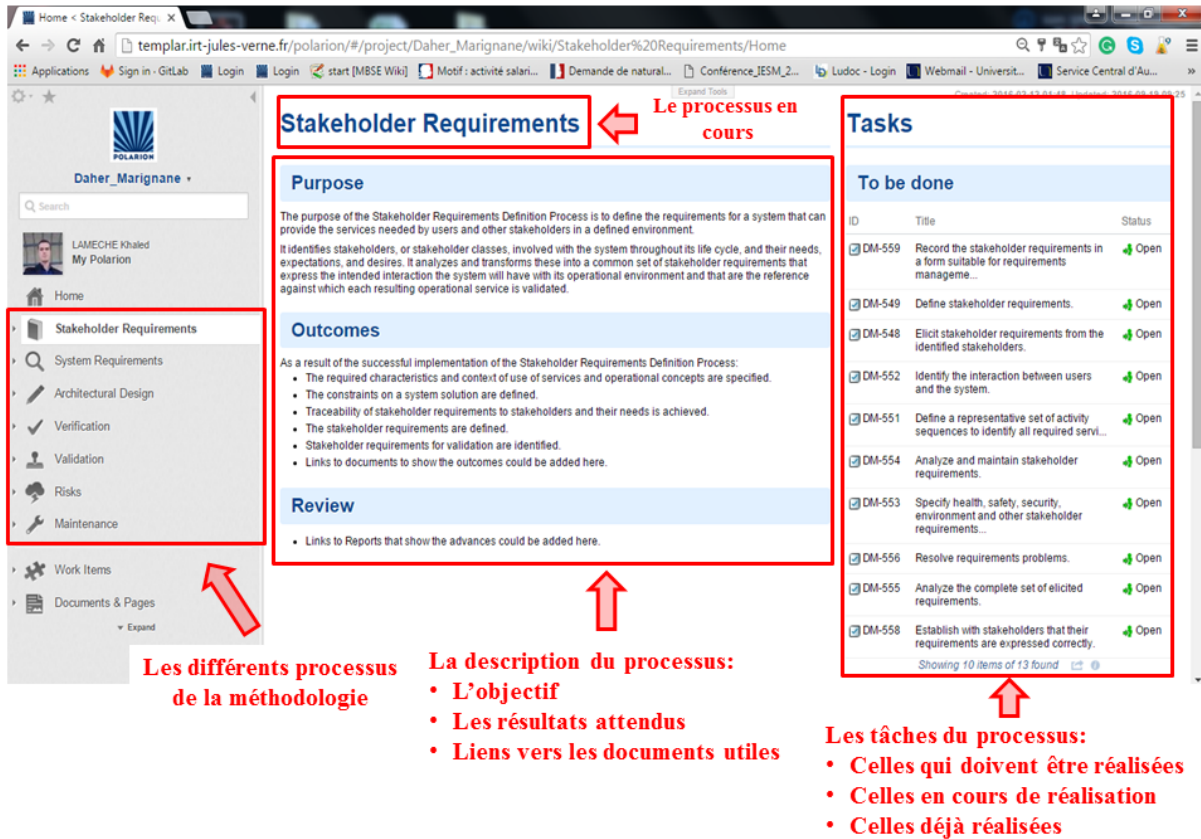


Figure 90 : Implémentation de la méthodologie sur l'outil Polarion, Page d'accueil.

3.2. Processus définition des besoins des parties prenantes

La figure suivante montre l'implémentation du processus 'définition des besoins des parties prenantes' (voir chapitre 4, section 2).

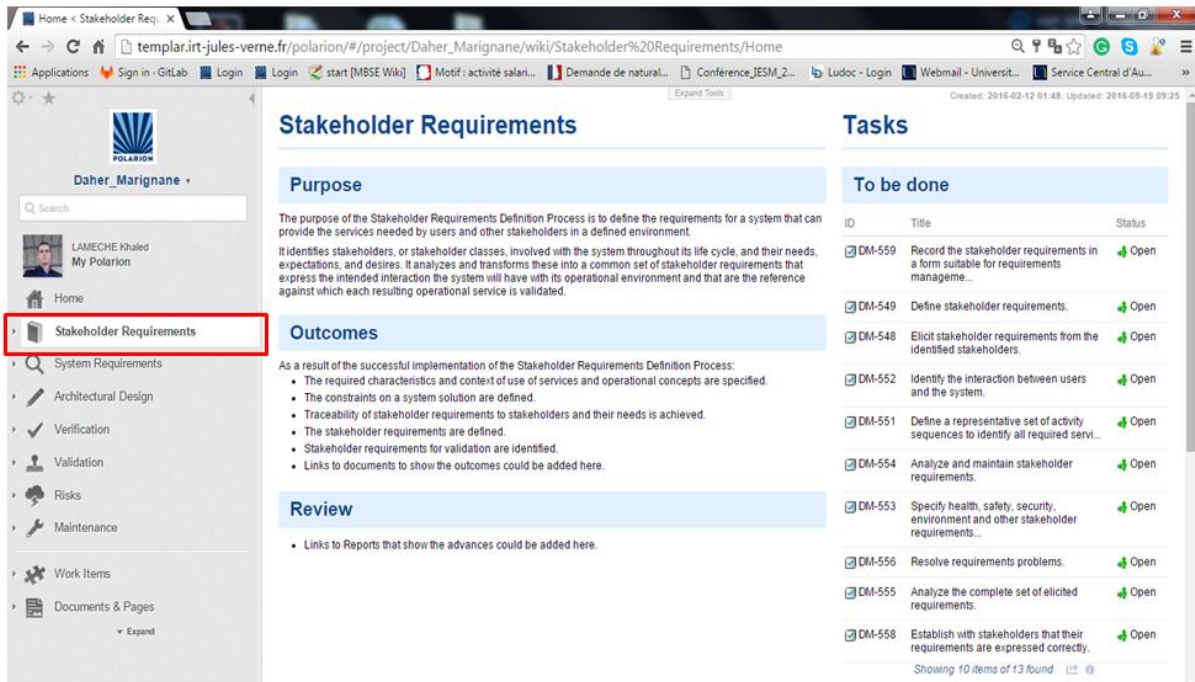


Figure 91 : Implémentation de la méthodologie sur l’outil Polarion, Processus ‘définition des besoins des PP’s’.

3.3. Processus spécification du système

La figure suivante montre l’implémentation du processus ‘spécification du système’ (voir chapitre 4, section 2).

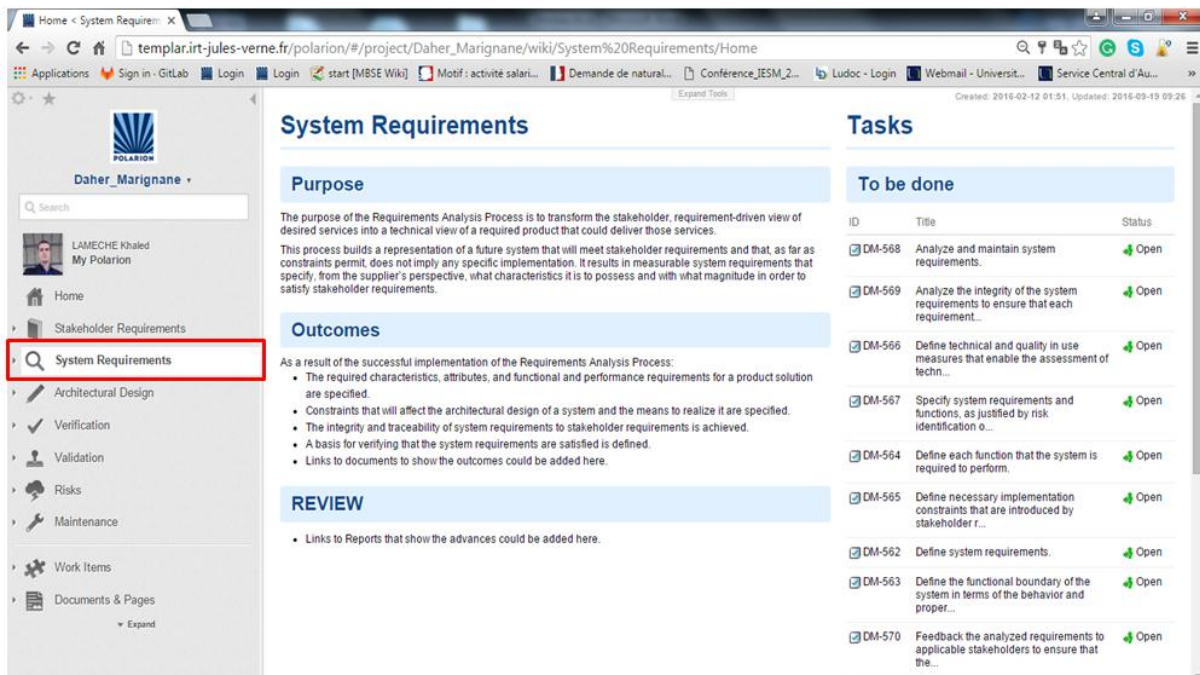


Figure 92 : Implémentation de la méthodologie sur l’outil Polarion, Processus ‘Spécification du système’.

3.4. Processus de conception de l'architecture du système

La figure suivante montre l'implémentation du processus 'conception de l'architecture du système' (voir chapitre 4, section 2).

The screenshot shows the Polarion web interface for the 'Architectural Design' process. The page is titled 'Architectural Design' and includes sections for Purpose, Outcomes, and REVIEW. A 'Tasks' table is also visible on the right side of the page.

Purpose

The purpose of the Architectural Design Process is to synthesize a solution that satisfies system requirements. This process encapsulates and defines areas of solution expressed as a set of separate problems of manageable, conceptual and, ultimately, realizable proportions. It identifies and explores one or more implementation strategies at a level of detail consistent with the system's technical and commercial requirements and risks. From this, an architectural design solution is defined in terms of the requirements for the set of system elements from which the system is configured. The specified design requirements resulting from this process are the basis for verifying the realized system and for devising an assembly and verification strategy.

Outcomes

As a result of the successful implementation of the Architectural Design Process:

- An architectural design baseline is established.
- The implementable set of system element descriptions that satisfy the requirements for the system are specified.
- The interface requirements are incorporated into the architectural design solution.
- The traceability of architectural design to system requirements is established.
- A basis for verifying the system elements is defined.
- A basis for the integration of system elements is established.
- Links to documents to show the outcomes could be added here.

REVIEW

- Links to Reports that show the advances could be added here.

Tasks

ID	Title	Status
DM-578	Analyze and evaluate the architecture.	Open
DM-585	Record the architectural design information.	Open
DM-574	Define the architecture.	Open
DM-575	Define appropriate logical architectural designs.	Open
DM-583	Document and maintain the architecture	Open
DM-577	Define and document the interfaces between system elements and at the system bou...	Open
DM-586	Maintain mutual traceability between specified design and system requirements.	Open
DM-579	Analyze the resulting architectural design to establish design criteria for each...	Open
DM-584	Specify the selected physical design solution as an architectural design baselin...	Open
DM-576	Partition the system functions identified in requirements analysis and allocate...	Open

Figure 93 : Implémentation de la méthodologie sur l'outil Polarion, Processus 'Conception de l'architecture du système'.

3.5. Processus vérification du système

La figure suivante montre l'implémentation du processus 'variation du système' (voir chapitre 4, section 2).

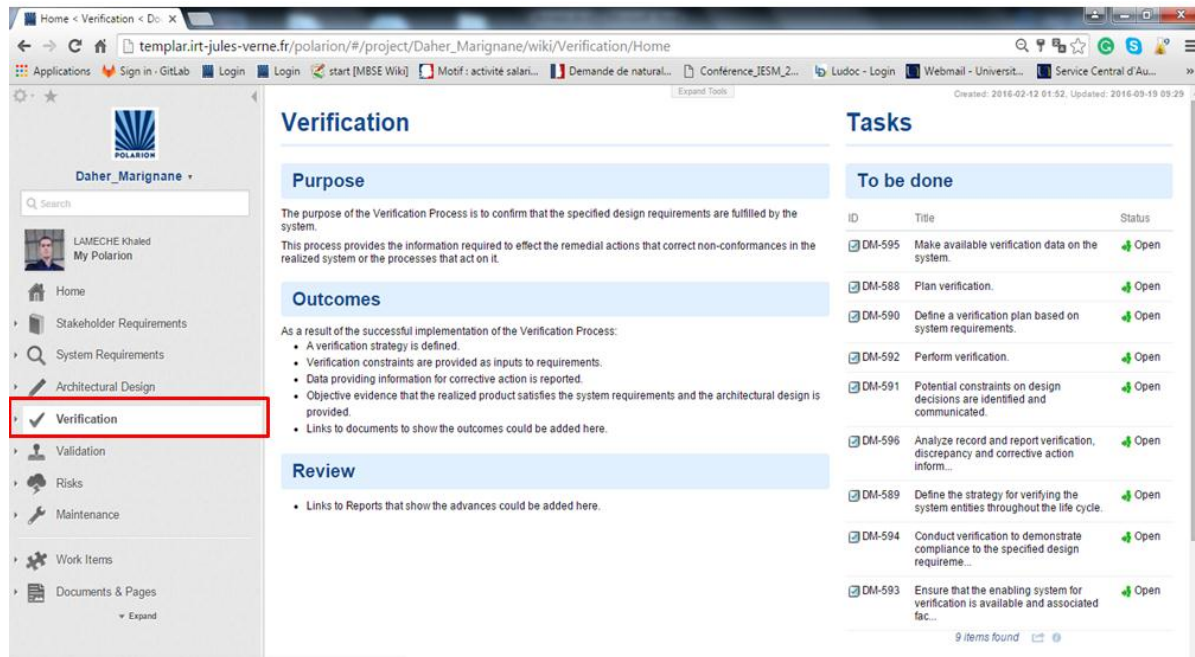


Figure 94 : Implémentation de la méthodologie sur l'outil Polarion, Processus 'Vérification du système'.

3.6. Processus validation du système

La figure suivante montre l'implémentation du processus 'validation du système' (voir chapitre 4, section 2).

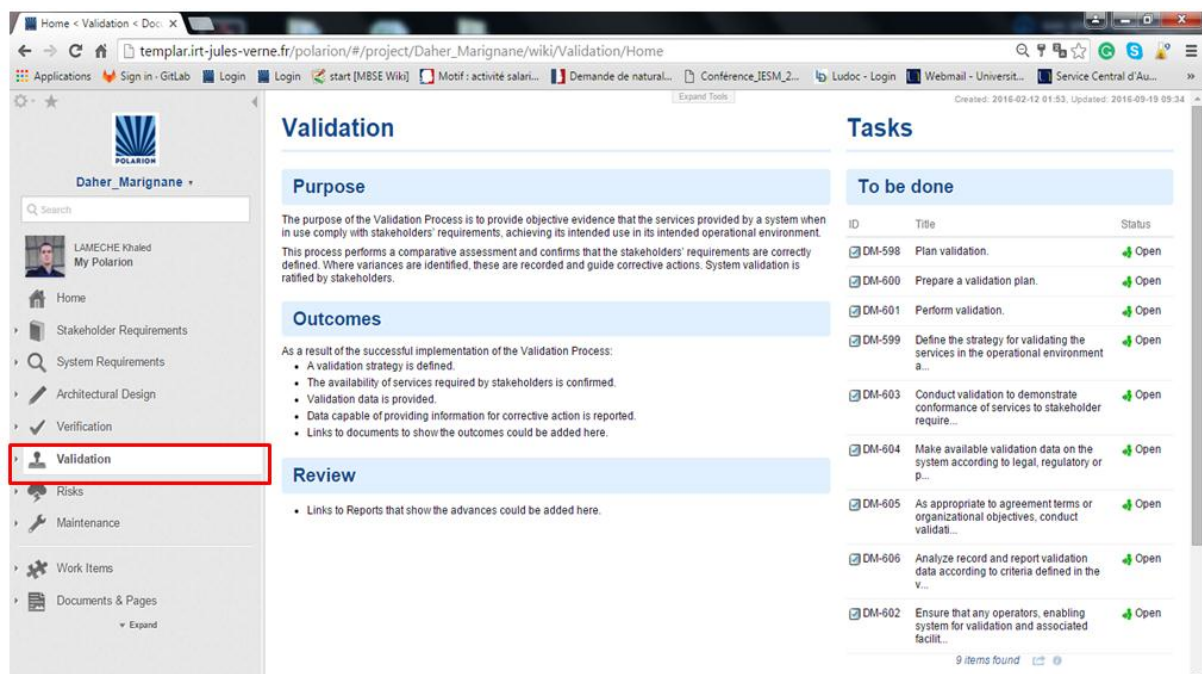


Figure 95 : Implémentation de la méthodologie sur l'outil Polarion, Processus 'Validation du système'.

3.7. Les éléments du travail implémentés

3.7.1. Les éléments du travail généraux

Les figures suivantes montrent les éléments de travail (exigence, test de vérification, une partie prenante, etc.) implémentés sur l’outil Polarion.

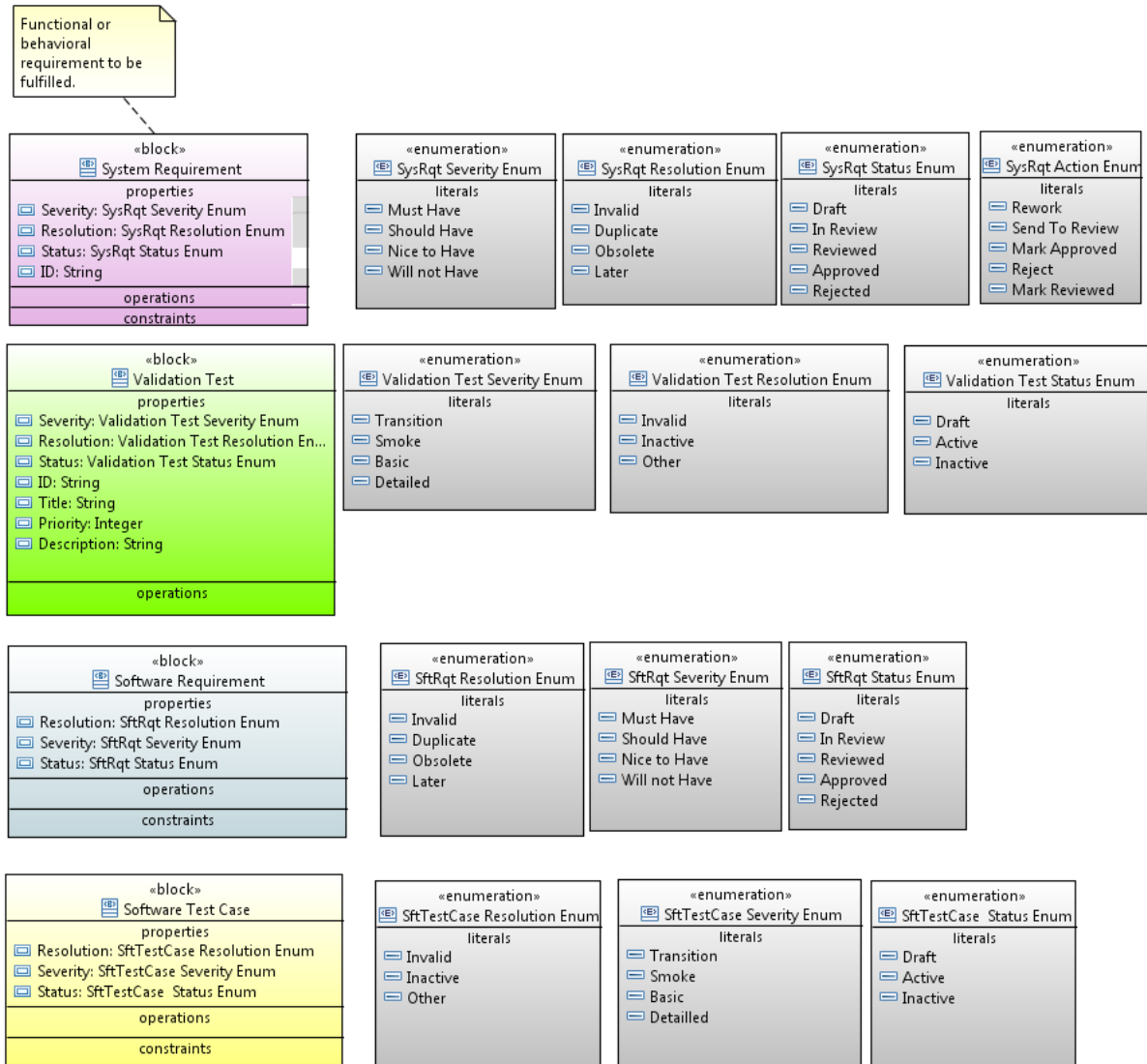


Figure 96 : les éléments du travail, partie 1.

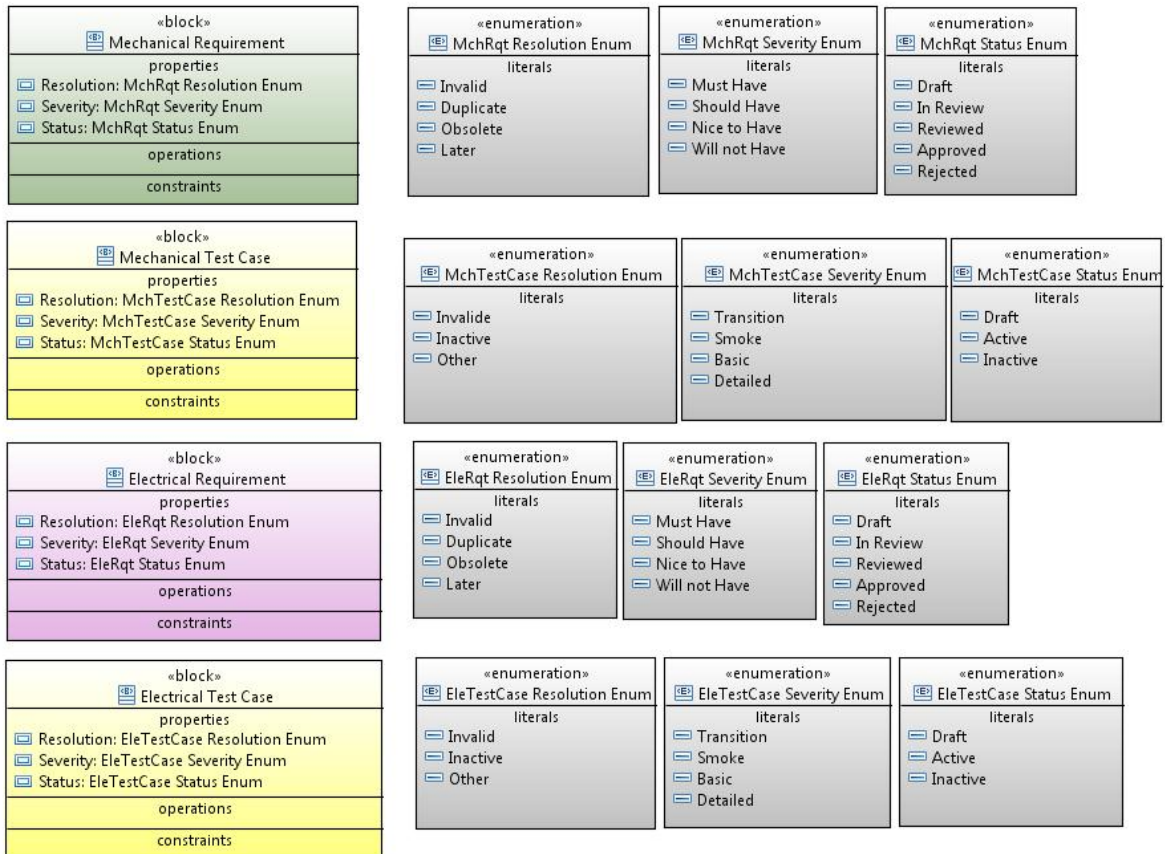


Figure 97 : les éléments du travail, partie 2.

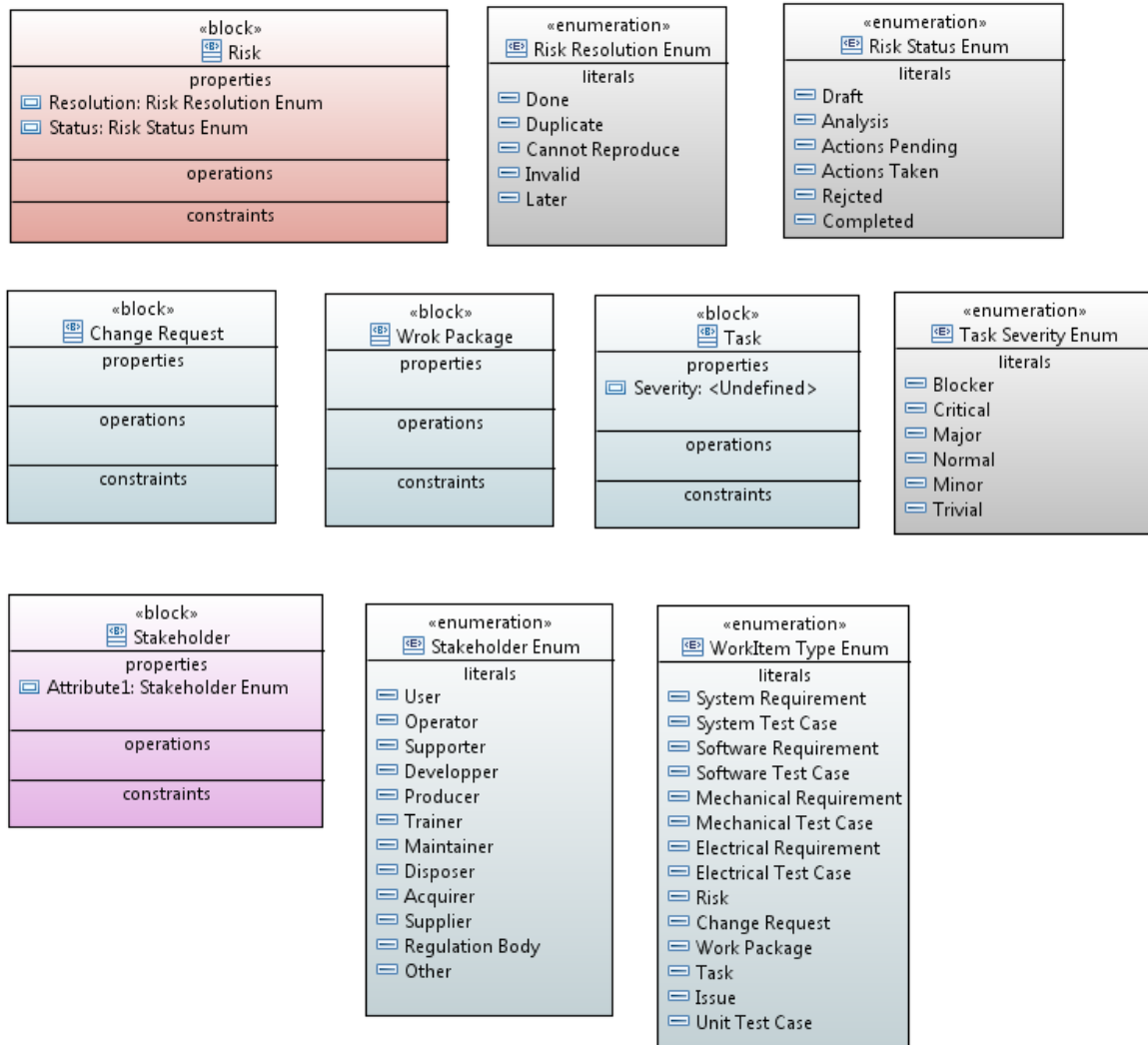


Figure 98 : les éléments du travail, partie 3.

3.7.2. Les éléments de travail des systèmes reconfigurables

Les éléments du travail propres aux systèmes reconfigurables et qui sont implémentés sur l'outil Polarion sont résumés dans la figure suivante.

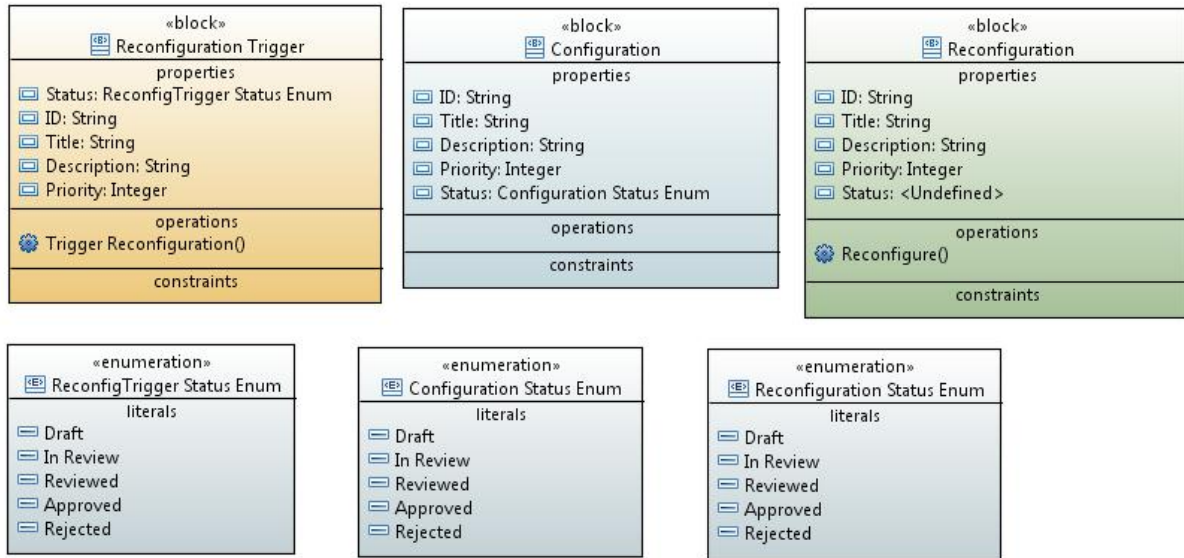


Figure 99 : les éléments du travail des systèmes reconfigurables.

3.8. Le cycle de vie d'une exigence

Le cycle de vie d'une exigence est détaillé dans la figure suivante.

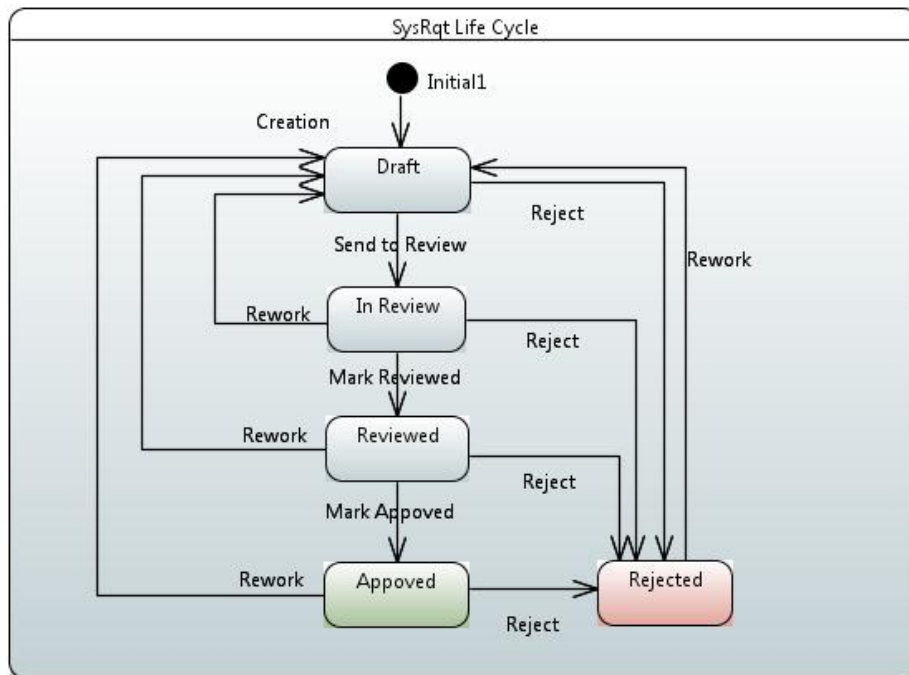


Figure 100 : Cycle de vie d'une exigence.

3.9. Le cycle de vie d'un test de vérification/validation

Le cycle de vie d'un test de vérification/validation est détaillé dans la figure suivante.

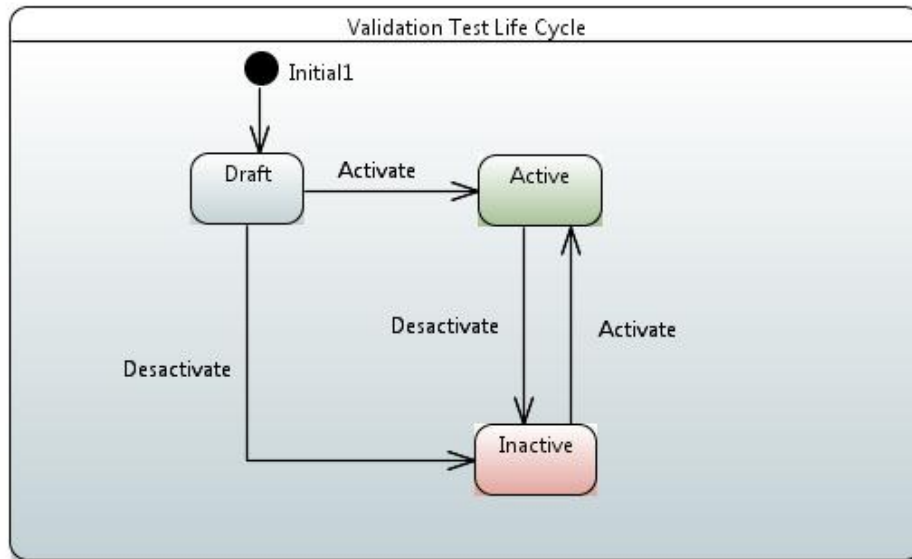


Figure 101 : Cycle de vie d'un test de vérification/validation.

3.10. Le cycle de vie d'un risque

Le cycle de vie d'un risque est détaillé dans la figure suivante.

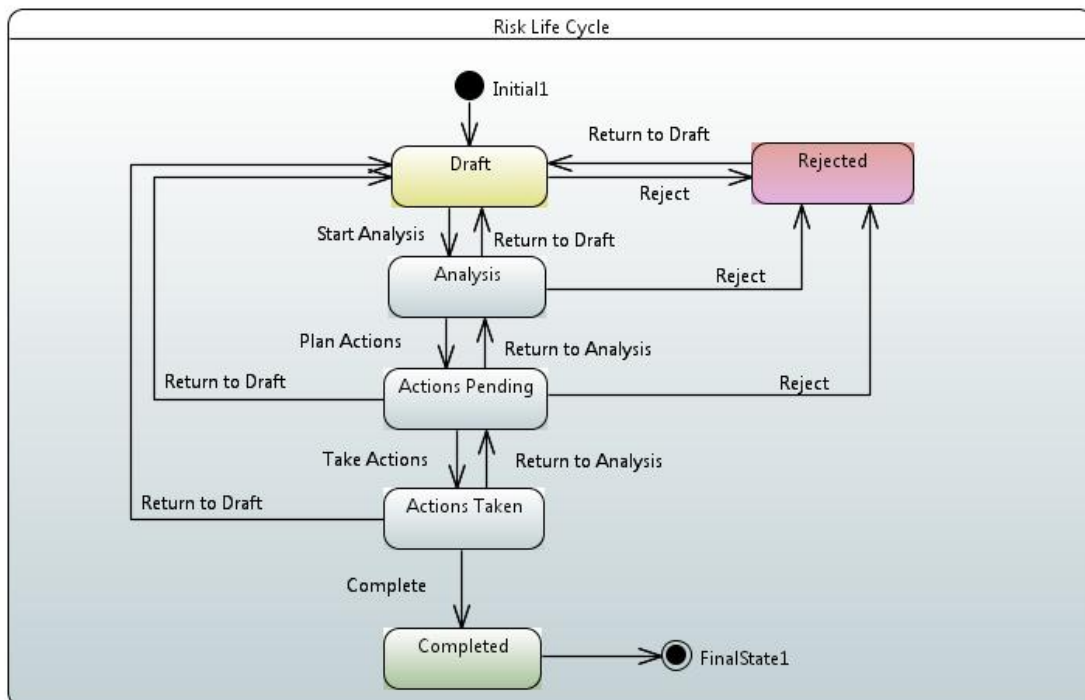


Figure 102 : Cycle de vie d'un risque.

Références

- (Roger) Jiao, Jianxin, Timothy W. Simpson, and Zahed Siddique. 2007. "Product Family Design and Platform-Based Product Development: A State-of-the-Art Review." *Journal of Intelligent Manufacturing* 18 (1): 5–29. doi:10.1007/s10845-007-0003-2.
- Abdi, M. R., and A. W. Labib. 2004a. "Grouping and Selecting Products: The Design Key of Reconfigurable Manufacturing Systems (RMSs)." *International Journal of Production Research* 42 (3): 521–46. doi:10.1080/00207540310001613665.
- Abdi, M. R., and A. W. Labib. 2004b. "Feasibility Study of the Tactical Design Justification for Reconfigurable Manufacturing Systems Using the Fuzzy Analytical Hierarchical Process." *International Journal of Production Research* 42 (15): 3055–76. doi:10.1080/00207540410001696041.
- Abdi, Mohammad Reza, and Ashraf W Labib. 2003. "A Design Strategy for Reconfigurable Manufacturing Systems (RMSs) Using Analytical Hierarchical Process (AHP): A Case Study." *International Journal of Production Research* 41 (10): 2273–99. doi:10.1080/0020754031000077266.
- Abdullah, A.B., M.S. Yusoff, and Z.M. Ripin. 2006. "Integration of Design for Modularity and Design for Assembly to Enhance Product Maintainability." In *Proceedings of the 1st International Conference & 7th AUN/SEED-Net Fieldwise Seminar on Manufacturing and Material Processing*, 263–67.
- Abma, B. J. M. 2009. "Evaluation of Requirements Management Tools with Support for Traceability-Based Change Impact Analysis." Software Engineering Faculty of Electrical Engineering, Mathematics and Computer Science University of Twente. http://www.utwente.nl/ewi/trese/graduation_projects/2009/Abma.pdf.
- Adam, Mickaël. 2013. "Génération Automatique D'un Observateur Réalisé Par Simulation D'un Système de Production." Ecole Centrale de Nantes.
- AFIS. 2009. *Découvrir et Comprendre l'Ingénierie Système*. Edited by AFIS. AFIS - GT Ingénierie Système. 3rded. ORSAY - France: AFIS. www.afis.fr.
- Aguilar, Alejandro, Armando Roman-Flores, and Joel C Huegel. 2013. "Design, Refinement, Implementation and Prototype Testing of a Reconfigurable Lathe-Mill." *Journal of Manufacturing Systems* 32 (2). The Society of Manufacturing Engineers: 364–71. doi:10.1016/j.jmsy.2013.01.003.
- AlGeddawy, T. N., and Hoda A. ElMaraghy. 2009. "Changeability Effect on Manufacturing Systems Design." In *Changeable and Reconfigurable Manufacturing Systems*, 267–83. London: Springer London. doi:10.1007/978-1-84882-067-8_15.
- AlGeddawy, Tarek, and Hoda ElMaraghy. 2011. "A Model for Co-Evolution in Manufacturing Based on Biological Analogy." *International Journal of Production Research* 49 (15): 4415–35. doi:10.1080/00207543.2010.497780.
- Alghazzawi, Daniyal M, Shams Tabrez Siddiqui, Mohammad Ubaidullah Bokhari, and Hatem S Abu Hamatta. 2014. "Selecting Appropriate Requirements Management Tool for Developing Secure Enterprises Software." *International Journal of Information Technology and Computer Science* 6 (4): 49–55. doi:10.5815/ijitcs.2014.04.06.
- Andersen, Ann-Louise, Thomas D. Brunoe, and Kjeld Nielsen. 2015. "Reconfigurable Manufacturing on Multiple Levels: Literature Review and Research Directions." In *IFIP Advances in Information and Communication Technology*, 459:266–73. doi:10.1007/978-3-319-22756-6_33.

- Andersen, Ann-Louise, Thomas Ditlev Brunoe, Kjeld Nielsen, and Carin Rösiö. 2017. "Towards a Generic Design Method for Reconfigurable Manufacturing Systems." *Journal of Manufacturing Systems* 42 (January). The Society of Manufacturing Engineers: 179–95. doi:10.1016/j.jmsy.2016.11.006.
- Andersen, Ann-Louise, Kjeld Nielsen, and Thomas Ditlev Brunoe. 2016. "Prerequisites and Barriers for the Development of Reconfigurable Manufacturing Systems for High Speed Ramp-Up." *Procedia CIRP* 51: 7–12. doi:10.1016/j.procir.2016.05.043.
- Azab, A., H. ElMaraghy, P. Nyhuis, J. Pachow-Frauenhofer, and M. Schmidt. 2013. "Mechanics of Change: A Framework to Reconfigure Manufacturing Systems." *CIRP Journal of Manufacturing Science and Technology* 6 (2). CIRP: 110–19. doi:10.1016/j.cirpj.2012.12.002.
- Azab, Ahmed, and Hoda ElMaraghy. 2007a. "Sequential Process Planning: A Hybrid Optimal Macro-Level Approach." *Journal of Manufacturing Systems* 26 (3–4): 147–60. doi:10.1016/j.jmsy.2008.03.003.
- Azab, Ahmed, and Hoda ElMaraghy. 2007b. "Mathematical Modeling for Reconfigurable Process Planning." *CIRP Annals - Manufacturing Technology* 56 (1): 467–72. doi:10.1016/j.cirp.2007.05.112.
- Baldwin, Carliss Y, and Kim B Clark. 2006. "Modularity in the Design of Complex Engineering Systems." In *Complex Engineered Systems*, edited by Dan Braha, Ali A Minai, and Yaneer Bar-Yam, 175–205. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/3-540-32834-3_9.
- Barata, José, Luis Camarinha-Matos, and Gonçalo Cândido. 2008. "A Multiagent-Based Control System Applied to an Educational Shop Floor." *Robotics and Computer-Integrated Manufacturing* 24 (5): 597–605. doi:10.1016/j.rcim.2007.09.008.
- Barhak, Jacob, Dragan Djurdjanovic, Patrick Spicer, and Reuven Katz. 2005. "Integration of Reconfigurable Inspection with Stream of Variations Methodology." *International Journal of Machine Tools and Manufacture* 45 (April): 407–19. doi:10.1016/j.ijmachtools.2004.09.011.
- Benkamoun, Nadège, Khalid Kouiss, and Anne-Lise Huyet. 2015. "AN INTELLIGENT DESIGN ENVIRONMENT FOR CHANGEABILITY MANAGEMENT - APPLICATION TO MANUFACTURING SYSTEMS." In *Proceedings of the 20th International Conference on Engineering Design (ICED15)*, 1–10.
- Benkamoun, Nadège, Waguih ElMaraghy, Anne-Lise Huyet, and Khalid Kouiss. 2014. "Architecture Framework for Manufacturing System Design." *Procedia CIRP* 17. Elsevier B.V.: 88–93. doi:10.1016/j.procir.2014.01.101.
- Bensmaine, Abderrahmane, Mohammed Dahane, and Lyes Benyoucef. 2013. "A Non-Dominated Sorting Genetic Algorithm Based Approach for Optimal Machines Selection in Reconfigurable Manufacturing Environment." *Computers & Industrial Engineering* 66 (November). Elsevier Ltd: 519–24. doi:10.1016/j.cie.2012.09.008.
- Bi, Z. M., Sherman Y T Lang, M. Verner, and P. Orban. 2008. "Development of Reconfigurable Machines." *The International Journal of Advanced Manufacturing Technology* 39 (11–12): 1227–51. doi:10.1007/s00170-007-1288-1.
- Bi, Z.M., S.Y.T Lang, W. Shen, and L. Wang. 2008. "Reconfigurable Manufacturing Systems: The State of the Art." *International Journal of Production Research* 46 (4): 967–92. doi:10.1080/00207540600905646.
- Boehm, Barry W. 1988. "Spiral Model of Software Development and Enhancement." *Computer* 21 (5). IEEE: 61–72. doi:10.1109/2.59.

- Bokhari, Mohammad, and Shams Siddiqui. 2010. "A Comparative Study of Software Requirements Tools for Secure Software Development Mohammad." In *Climate Change 2013 - The Physical Science Basis*, edited by Intergovernmental Panel on Climate Change, 2:1–30. Cambridge: Cambridge University Press. doi:10.1017/CBO9781107415324.004.
- Borisovsky, Pavel A., Xavier Delorme, and Alexandre Dolgui. 2013. "Genetic Algorithm for Balancing Reconfigurable Machining Lines." *Computers and Industrial Engineering* 66 (3). Elsevier Ltd: 541–47. doi:10.1016/j.cie.2012.12.009.
- Bruccoleri, M., G. Lo Nigro, G. Perrone, P. Renna, and S. Noto La Diega. 2005. "Production Planning in Reconfigurable Enterprises and Reconfigurable Production Systems." *CIRP Annals - Manufacturing Technology* 54 (2): 433–36. doi:10.1016/S0007-8506(07)60138-3.
- Bruccoleri, Manfredi. 2007. "Reconfigurable Control of Robotized Manufacturing Cells." *Robotics and Computer-Integrated Manufacturing* 23 (1): 94–106. doi:10.1016/j.rcim.2005.08.005.
- Bruccoleri, Manfredi, Zbigniew J. Pasek, and Yoram Koren. 2006. "Operation Management in Reconfigurable Manufacturing Systems: Reconfiguration for Error Handling." *International Journal of Production Economics* 100 (1): 87–100. doi:10.1016/j.ijpe.2004.10.009.
- Bruch, Jessica. 2012. "Management of Design Information in the Production System Design Process." <http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A488464&dswid=138>.
- Bryan, A., J. Ko, S. J. Hu, and Y. Koren. 2007. "Co-Evolution of Product Families and Assembly Systems." *CIRP Annals - Manufacturing Technology* 56 (1): 41–44. doi:10.1016/j.cirp.2007.05.012.
- Carpanzano, E., A. Cesta, A. Orlandini, R. Rasconi, and A. Valente. 2014. "Intelligent Dynamic Part Routing Policies in Plug&Produce Reconfigurable Transportation Systems." *CIRP Annals - Manufacturing Technology* 63 (1). CIRP: 425–28. doi:10.1016/j.cirp.2014.03.111.
- Chalfoun, Imad, Khalid Kouiss, Anne-Lise Huyet, Nicolas Bouton, and Pascal Ray. 2013. "Proposal for a Generic Model Dedicated to Reconfigurable and Agile Manufacturing Systems (RAMS)." In *Procedia CIRP*, 7:485–90. Elsevier B.V. doi:10.1016/j.procir.2013.06.020.
- Chaube, A., L. Benyoucef, and M. K. Tiwari. 2012. "An Adapted NSGA-2 Algorithm Based Dynamic Process Plan Generation for a Reconfigurable Manufacturing System." *Journal of Intelligent Manufacturing* 23 (4): 1141–55. doi:10.1007/s10845-010-0453-9.
- Chryssolouris, George. 2006. *Manufacturing Systems: Theory and Practice*. Edited by Frederick F.Ling. Control. 2nd ed. New York: Springer-Verlag New York. doi:10.1007/0-387-28431-1.
- Chun-Che Huang, and Andrew Kusiak. 1998. "Modularity in Design of Products and Systems." *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 28 (1): 66–77. doi:10.1109/3468.650323.
- Clark, John O. 2009. "System of Systems Engineering and Family of Systems Engineering from a Standards, V-Model, and Dual-V Model Perspective." In *2009 3rd Annual IEEE Systems Conference*, 381–87. IEEE. doi:10.1109/SYSTEMS.2009.4815831.
- Da Silva, Robson Marinho, Fabrício Junqueira, Diolino J Santos, and Paulo E Miyagi. 2016. "Control Engineering Practice Control Architecture and Design Method of Recon Fi Gurable Manufacturing Systems." *Control Engineering Practice* 49. Elsevier: 87–100. doi:10.1016/j.conengprac.2016.01.009.
- Dai, X., J. Li, and Z. Meng. 2009. "Hierarchical Petri Net Modelling of Reconfigurable Manufacturing Systems with Improved Net Rewriting Systems." *International Journal of Computer Integrated Manufacturing* 22 (2): 158–77. doi:10.1080/09511920802014904.

- Deif, Ahmed M., and Waguih H. ElMaraghy. 2006. "A Systematic Design Approach for Reconfigurable Manufacturing Systems." In *Advances in Design*, 219–28. London: Springer-Verlag. doi:10.1007/1-84628-210-1_18.
- Dhupia, Jaspreet, Bartosz Powalka, Reuven Katz, and A. Galip Ulsoy. 2007. "Dynamics of the Arch-Type Reconfigurable Machine Tool." *International Journal of Machine Tools and Manufacture* 47 (2): 326–34. doi:10.1016/j.ijmachtools.2006.03.017.
- Dolang. 2017. "DLWL-800A Modular Manufacturing System of Modern Logistics." <http://www.dolang.cn/ying/cpcon.php?pid=843>.
- Dori, Dov. 2002. *Object-Process Methodology*. 1sted. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-56209-9.
- Dvorak, D, R Rasmussen, G Reeves, and A Sacks. 2000. "Software Architecture Themes in JPL's Mission Data System." In *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*, 7:259–68. IEEE. doi:10.1109/AERO.2000.879293.
- EIA/ANSI. 1999. *EIA-632: Processes for Engineering a System*. Edited by EIA. EIA. Arlington: ELECTRONIC INDUSTRIES ALLIANCE. doi:10.1016/S0737-0806(99)80290-1.
- Eisner, Howard. 2002. *Essentials of Project Management and Systems Engineering Management*. Edited by Wiley. 2nded.
- El Maraghy, Hoda A. 2006. "Flexible and Reconfigurable Manufacturing Systems Paradigms." *Flexible Services and Manufacturing Journal* 17 (4 SPECIAL ISSUE): 261–76. doi:10.1007/s10696-006-9028-7.
- ElMaraghy, H., G. Schuh, W. Elmaraghy, F. Piller, P. Sch??nsleben, M. Tseng, and A. Bernard. 2013. "Product Variety Management." *CIRP Annals - Manufacturing Technology* 62 (2): 629–52. doi:10.1016/j.cirp.2013.05.007.
- Elmaraghy, Hoda A. 2007. "Reconfigurable Process Plans for Responsive Manufacturing Systems." In *Digital Enterprise Technology*, edited by P.F. Cunha and P.G. Maropoulos, 35–44. Boston, MA: Springer, Boston, MA. doi:10.1007/978-0-387-49864-5_4.
- ElMaraghy, Hoda A. 2007. "Reconfigurable Process Plans For Responsive Manufacturing Systems." In *Digital Enterprise Technology: Perspectives and Future Challenges*, edited by Pedro Filipe Cunha and Paul G Maropoulos, 35–44. Boston, MA: Springer US. doi:10.1007/978-0-387-49864-5_4.
- ElMaraghy, Hoda A., and H.-P. Wiendahl. 2009. "Changeability – An Introduction." In *Changeable and Reconfigurable Manufacturing Systems*, 3–24. London: Springer London. doi:10.1007/978-1-84882-067-8_1.
- Elmaraghy, W. H., O. A. Nada, and H. A. Elmaraghy. 2008. "Quality Prediction for Reconfigurable Manufacturing Systems via Human Error Modelling." *International Journal of Computer Integrated Manufacturing* 21 (5): 584–98. doi:10.1080/09511920701233464.
- Estefan, Jeff A. 2008. "Survey of Model-Based Systems Engineering (MBSE) Methodologies." Pasadena, California, U.S.A. doi:10.1109/35.295942.
- festo-didactic. 2017. "CP Factory: The Cyber-Physical Factory." <http://www.festo-didactic.com/int-en/learning-systems/learning-factories,cim-fms-systems/cp-factory/53768.htm?fbid=aW50LmVuLjU1Ny4xNy4xOC4xMjkzLjUzNzY4>.
- Francalanza, Emmanuel, Jonathan Borg, and Carmen Constantinescu. 2014. "Deriving a Systematic Approach to Changeable Manufacturing System Design." In *Procedia CIRP: Variety*

- Management in Manufacturing. Proceedings of the 47th CIRP Conference on Manufacturing Systems*, 17:166–71. Elsevier B.V. doi:10.1016/j.procir.2014.01.111.
- Galan, R., J. Racero, I. Eguia, and J.M. Garcia. 2007. “A Systematic Approach for Product Families Formation in Reconfigurable Manufacturing Systems.” *Robotics and Computer-Integrated Manufacturing* 23 (5): 489–502. doi:10.1016/j.rcim.2006.06.001.
- Gendre, Lionel, and Jean-Marie Virely. 2013. “Systems Modeling Language SysML.” Cachan, France: ENS Cachan.
- Gershenson, John K., and G. Jagannath Prasad. 1997. “Product Modularity and Its Effect on Service and Maintenance.” In *Proceedings of the 1997 Maintenance and Reliability Conference*.
- Goyal, Kapil Kumar, P. K. Jain, and Madhu Jain. 2013. “A Comprehensive Approach to Operation Sequence Similarity Based Part Family Formation in the Reconfigurable Manufacturing System.” *International Journal of Production Research* 51 (6): 1762–76. doi:10.1080/00207543.2012.701771.
- Goyal, Kapil Kumar, P.K. Jain, and Madhu Jain. 2012. “Optimal Configuration Selection for Reconfigurable Manufacturing System Using NSGA II and TOPSIS.” *International Journal of Production Research* 50 (15): 4175–91. doi:10.1080/00207543.2011.599345.
- Guan, Xianping, Xianzhong Dai, Baijing Qiu, and Jun Li. 2012. “A Revised Electromagnetism-like Mechanism for Layout Design of Reconfigurable Manufacturing System.” *Computers & Industrial Engineering* 63 (1). Elsevier Ltd: 98–108. doi:10.1016/j.cie.2012.01.016.
- Gyulai, D., Z. Vén, A. Pfeiffer, J. Vánca, and L. Monostori. 2012. “Matching Demand and System Structure in Reconfigurable Assembly Systems.” In *Procedia CIRP*, 3:579–84. D. Gyulai. doi:10.1016/j.procir.2012.07.099.
- Gyulai, Dávid, Botond Kádár, and László Monostori. 2014. “Capacity Planning and Resource Allocation in Assembly Systems Consisting of Dedicated and Reconfigurable Lines.” *Procedia CIRP* 25 (C). Elsevier B.V.: 185–91. doi:10.1016/j.procir.2014.10.028.
- Haskins, Cecilia. 2006. *Systems Engineering Handbook : A Guide for System Life Cycle Processes and Activities*. Edited by Cecilia Haskins. 3rded. INCOSE.
- Haskins, Cecilia, Kevin Forsberg, and Michael Krueger. 2007. *Systems Engineering Handbook : A Guide for System Life Cycle Processes and Activities*. Edited by Cecilia Haskins, Kevin Forsberg, Michael Krueger, and CSEP. *Systems Engineering*. INCOSE. <http://dl.acm.org/citation.cfm?id=552393>.
- Heisel, U., and M. Meitzner. 2006. “Progress in Reconfigurable Manufacturing Systems.” In *Reconfigurable Manufacturing Systems and Transformable Factories*, 47–62. doi:978-3-540-29397-2.
- Hoffmann, Hans-Peter. 2011. *System Engineering Best Practices with the Rational Solution for System and Software Engineering*. Edited by IBM Software Group. 3.1.2. doi:10.1109/MS.2009.80.
- Höltkä-Otto, Katja, and Olivier de Weck. 2007. “Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints.” *Concurrent Engineering* 15 (2): 113–26. doi:10.1177/1063293X07078931.
- Höltkä, Katja, Suh Eun Suk, and Olivier de Weck. 2005. “Tradeoff Between Modularity And Performance For Engineered Systems And Products.” In *International Conference on Engineering Design*, 1–14.
- IEEE. 2005. “IEEE 1220 Standard for Application and Management of the Systems Engineering

- Process.” New York: IEEE. doi:10.1109/IEEESTD.2010.5703195.
- INCOSE. 2016. “International Council on Systems Engineering.” *Wikipedia*. <http://www.incose.org/>.
- ISO/IEC/IEEE15288. 2015. “ISO/IEC/IEEE15288: Systems and Software Engineering – System Life Cycle Processes.” Edited by ISO, IEC, and IEEE. Switzerland: ISO/IEC/IEEE.
- Joergensen, Steffen N., Kjeld Nielsen, and Kaj A. Joergensen. 2010. “Reconfigurable Manufacturing Systems as an Application of Mass Customisation.” *International Journal of Industrial Engineering and Management (IJIEM)* 1 (3): 111–19.
- Kahloul, Laid, Karim Djouani, and Allaoua Chaoui. 2013. “Formal Study of Reconfigurable Manufacturing Systems: A High Level Petri Nets Based Approach.” In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8062 LNAI:106–17. doi:10.1007/978-3-642-40090-2_10.
- Kashkoush, Mohamed, and Hoda ElMaraghy. 2014. “Product Family Formation for Reconfigurable Assembly Systems.” In *Proceedings of the 47th CIRP Conference on Manufacturing Systems*, 17:302–7. Elsevier B.V. doi:10.1016/j.procir.2014.01.131.
- Katz, Reuven. 2007. “Design Principles of Reconfigurable Machines.” *The International Journal of Advanced Manufacturing Technology* 34 (5–6): 430–39. doi:10.1007/s00170-006-0615-2.
- Koren, Y., U. Heisel, F. Jovane, T. Moriwaki, G. Pritschow, G. Ulsoy, and H. Van Brussel. 1999. “Reconfigurable Manufacturing Systems.” *CIRP Annals* 48 (2): 527–40. doi:10.1016/S0007-8506(07)63232-6.
- Koren, Y., S. Jack Hu, and Thomas W. Weber. 1998. “Impact of Manufacturing System Configuration on Performance.” *CIRP Annals - Manufacturing Technology* 47 (1): 369–72. doi:10.1016/S0007-8506(07)62853-4.
- Koren, Y., and Galip Ulsoy. 2002. “Vision, Principles and Impact of Reconfigurable Manufacturing Systems.” *Powertrain International* 5 (3): 14–21.
- Koren, Yoram. 2005. “Reconfigurable Manufacturing and Beyond.” In *Summary of a Keynote Speech at CIRP05, 3rd International Conference on Reconfigurable Manufacturing*, 1–7. Ann Arbor, Michigan, USA.
- Koren, Yoram. 2010. *The Global Manufacturing Revolution: Product-Process-Business Integration and Reconfigurable Systems*. Edited by Yoram Koren. *The Global Manufacturing Revolution: Product-Process-Business Integration and Reconfigurable Systems*. 1sted. Hoboken, New Jersey: John Wiley & Sons, Inc. doi:10.1002/9780470618813.
- Koren, Yoram. 2013. “The Rapid Responsiveness of RMS.” *International Journal of Production Research* 51 (23–24): 6817–27. doi:10.1080/00207543.2013.856528.
- Koren, Yoram, Zbigniew J. Pasek, A. Galip Ulsoy, and Uri Benchetrit. 1996. “Real-Time Open Control Architectures and System Performance.” *CIRP Annals - Manufacturing Technology* 45 (1): 377–80. doi:10.1016/S0007-8506(07)63084-4.
- Koren, Yoram, and Moshe Shpitalni. 2010. “Design of Reconfigurable Manufacturing Systems.” *Journal of Manufacturing Systems* 29 (4). Elsevier Ltd: 130–41. doi:10.1016/j.jmsy.2011.01.001.
- Kruger, K., and A. H. Basson. 2013. “Multi-Agent Systems vs IEC 61499 for Holonic Resource Control in Reconfigurable Systems.” In *Procedia CIRP*, 7:503–8. Elsevier B.V. doi:10.1016/j.procir.2013.06.023.
- Lallican, Jean-Louis. 2007. “Proposition D’une Approche Composant Pour La Conception de La

- Commande Des Systèmes Transistiques.” Université Bretagne-Sud.
- Lameche, Khaled, Najib Mohamed Najid, Pierre Castagna, and Khalid Kouiss. 2016. “Utilisation de La Simulation Dans La Conception Des Systèmes Logistiques Reconfigurables.” In *11e Conférence Francophone d’Optimisation et Simulation- MOSIM’16*.
- Lameche, Khaled, Najib Mohamed Najid, Pierre Castagna, and Khalid Kouiss. 2017a. “Modularity in the Design of Reconfigurable Manufacturing Systems.” In *The 20th World Congress of the International Federation of Automatic Control*.
- Lameche, Khaled, Najib Mohamed Najid, Pierre Castagna, and Khalid Kouiss. 2017b. “System Engineering-Based Methodology to Design Reconfigurable Manufacturing Systems.”
- Landers, R G, B M, and Y Koren. 2001. “Reconfigurable Machine Tools.” *CIRP Annals - Manufacturing Technology* 50 (1): 1–6.
- Law, Averill M., and W.David Kelton. 1991. *Simulation Modeling and Analysis*. Edited by McGraw-Hill series in industrial engineering and and Management Science. 2nded. Singapore. doi:10.1145/1667072.1667074.
- Li, Jun, Xianzhong Dai, Zhengda Meng, Jianping Dou, and Xianping Guan. 2009. “Rapid Design and Reconfiguration of Petri Net Models for Reconfigurable Manufacturing Cells with Improved Net Rewriting Systems and Activity Diagrams.” *Computers & Industrial Engineering* 57 (4). Elsevier Ltd: 1431–51. doi:10.1016/j.cie.2009.07.013.
- Lorenzer, Th., S. Weikert, S. Bossoni, and K. Wegener. 2007. “Modeling and Evaluation Tool for Supporting Decisions on the Design of Reconfigurable Machine Tools.” *Journal of Manufacturing Systems* 26 (3–4): 167–77. doi:10.1016/j.jmsy.2008.01.002.
- Maler-Speredelozzi, V., Y. Koren, and S.J. Hu. 2003. “Convertibility Measures for Manufacturing Systems.” *CIRP Annals - Manufacturing Technology* 52 (1): 367–70. doi:10.1016/S0007-8506(07)60603-9.
- Malhotra, V, T Raj, and A Arora. 2009. “Reconfigurable Manufacturing System : An Overview.” *International Journal of Machine Intelligence* 1 (2): 38–46.
- Malhotra, V, T Raj, and A Arora. 2010. “Excellent Techniques of Manufacturing Systems: RMS and FMS.” *International Journal of Engineering Science and Technology* 2 (3): 137–42.
- Malhotra, Vasdev, Tilak Raj, and Ashok Arora. 2012. “Evaluation of Barriers Affecting Reconfigurable Manufacturing Systems with Graph Theory and Matrix Approach.” *Materials and Manufacturing Processes* 27 (1): 88–94. doi:10.1080/10426914.2011.551963.
- Mehrabi, M. G., A. G. Ulsoy, and Y. Koren. 2000a. “Reconfigurable Manufacturing Systems: Key to Future Manufacturing.” *Journal of Intelligent Manufacturing* 11 (4): 403–419. doi:10.1023/A:1008930403506.
- Mehrabi, M. G., A. G. Ulsoy, and Y. Koren. 2000b. “Reconfigurable Manufacturing Systems and Their Enabling Technologies.” *International Journal of Manufacturing Technology and Management* 1 (1): 1–28.
- Mehrabi, M. G., A. G. Ulsoy, Y. Koren, and P. Heytler. 2002. “Trends and Perspectives in Flexible and Reconfigurable Manufacturing Systems.” *Journal of Intelligent Manufacturing* 13 (2): 135–46. doi:10.1023/A:1014536330551.
- Mehrabi, M.G., and E. Kannatey-Asibu. 2001. “Mapping Theory: A New Approach to Design of Multi-Sensor Monitoring of Reconfigurable Machining Systems (RMS).” *Journal of Manufacturing Systems* 20 (5): 297–304. doi:10.1016/S0278-6125(01)80049-2.

- Meng, Xiuli. 2010. "Modeling of Reconfigurable Manufacturing Systems Based on Colored Timed Object-Oriented Petri Nets." *Journal of Manufacturing Systems* 29 (2–3). Elsevier Ltd: 81–90. doi:10.1016/j.jmsy.2010.11.002.
- Michaelis, M.T., and Hans Johannesson. 2012. "From Dedicated to Platform-Based Co-Development of Products and Manufacturing Systems." In *Enabling Manufacturing Competitiveness and Economic Sustainability*, 196–202. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-23860-4_32.
- Mittal, Kamal Kumar, and Pramod Kumar Jain. 2014. "An Overview of Performance Measures in Reconfigurable Manufacturing System." In *Procedia Engineering*, 69:1125–29. Elsevier B.V. doi:10.1016/j.proeng.2014.03.100.
- Morales-Velazquez, Luis, Rene De Jesus Romero-Troncoso, Roque Alfredo Osornio-Rios, Gilberto Herrera-Ruiz, and Eduardo Cabal-Yepez. 2010. "Open-Architecture System Based on a Reconfigurable Hardware-Software Multi-Agent Platform for CNC Machines." *Journal of Systems Architecture* 56 (9). Elsevier B.V.: 407–18. doi:10.1016/j.sysarc.2010.04.009.
- Murray, Julia. 2012. "Model Based Systems Engineering (MBSE) Media Study."
- Musharavati, F., and A. M S Hamouda. 2012. "Simulated Annealing with Auxiliary Knowledge for Process Planning Optimization in Reconfigurable Manufacturing." *Robotics and Computer-Integrated Manufacturing* 28 (2). Elsevier: 113–31. doi:10.1016/j.rcim.2011.07.003.
- Najid, Najib M., and Khalid Kouiss. 2007. "La Simulation Apporte de La Flexibilité À L'outil de Production." *MESURES* 793.
- Odrey, Nicholas G., and Gonzalo Mejía. 2003. "A Re-Configurable Multi-Agent System Architecture for Error Recovery in Production Systems." *Robotics and Computer-Integrated Manufacturing* 19 (1–2): 35–43. doi:10.1016/S0736-5845(02)00060-1.
- Ossama, Mohamed, Ayman M.A. Youssef, and Mohamed A. Shalaby. 2014. "A Multi-Period Cell Formation Model for Reconfigurable Manufacturing Systems." In *Variety Management in Manufacturing. Proceedings of the 47th CIRP Conference on Manufacturing Systems*, 17:130–35. Elsevier B.V. doi:10.1016/j.procir.2014.01.120.
- Padayachee, J, and G Bright. 2012. "Modular Machine Tools: Design and Barriers to Industrial Implementation." *Journal of Manufacturing Systems* 31 (2). The Society of Manufacturing Engineers: 92–102. doi:10.1016/j.jmsy.2011.10.003.
- Pearce, Paul, and Matthew Hause. 2012. "ISO-15288, OOSEM and Model-Based Submarine Design." In *Systems Engineering, Test and Evaluation Conferece SETE & 6th Asia Pacific Conference on Systems Engineering APOCSE*, edited by INCOSE. Brisbane, Australia.
- Pritschow, Günter, Yusuf Altintas, Francesco Jovane, Yoram Koren, Mamoru Mitsuishi, Shozo Takata, Hendrik van Brussel, et al. 2001. "Open Controller Architecture – Past, Present and Future." *CIRP Annals - Manufacturing Technology* 50 (2): 463–70. doi:10.1016/S0007-8506(07)62993-X.
- Puik, Erik, Paul Gielen, Daniel Telgen, Leo van Moergestel, and Darek Ceglarek. 2014. "A Generic Systems Engineering Method for Concurrent Development of Products and Manufacturing Equipment." In *Precision Assembly Technologies and Systems, Ipas 2014*, 435:139–46. doi:10.1007/978-3-662-45586-9_18.
- Puik, Erik, Daniel Telgen, Leo van Moergeste, and Darek Ceglarek. 2013. "Qualitative Product/process Modelling for Reconfigurable Manufacturing Systems." In *2013 IEEE International Symposium on Assembly and Manufacturing (ISAM)*, 214–18. IEEE.

doi:10.1109/ISAM.2013.6643530.

Rockwell Automation. 2012. "Arena Template Developer's Guide."

Rösiö, C, and M Jackson. 2009. "Enable Changeability in Manufacturing Systems by Adopting a Life Cycle Perspective." In *Proceedings of the 3rd International Conference on Changeable, Agile, Reconfigurable and Virtual Production*, 612–21.

Rösiö, Carin. 2012a. "Considering Reconfigurability Characteristics in Production System Design." In *Enabling Manufacturing Competitiveness and Economic Sustainability*, 57–62. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-23860-4_9.

Rösiö, Carin. 2012b. "Supporting the Design of Reconfigurable Production Systems." School of engineering, University of Malardalen.

Rösiö, Carin, and Kristina Säfsten. 2013. "Reconfigurable Production System Design – Theoretical and Practical Challenges." *Journal of Manufacturing Technology Management* 24 (7): 998–1018. doi:10.1108/JMTM-02-2012-0021.

Royce, W W. 1970. "Managing the Development of Large Software Systems." *Electronics* 26 (August): 1–9. doi:10.1016/0378-4754(91)90107-E.

Schuh, G., M. Lenders, C. Nussbaum, and D. Kupke. 2009. "Design for Changeability." In *Changeable and Reconfigurable Manufacturing Systems*, edited by Hoda A. ElMaraghy, 251–66. Springer Series in Advanced Manufacturing. London: Springer London. doi:10.1007/978-1-84882-067-8.

Siddiqui, Shams Tabrez, and Mohammad Ubaidullah Bokhari. 2013. "Needs, Types and Benefits of Requirements Management Tools." *International Journal of Trends in Computer Science* 2 (11): 433–41. http://www.academia.edu/5132154/Needs_Types_and_Benefits_of_Requirements_Management_Tools.

Son, Hungsun, Hae-Jin Choi, and Hyung Wook Park. 2010. "Design and Dynamic Analysis of an Arch-Type Desktop Reconfigurable Machine." *International Journal of Machine Tools and Manufacture* 50 (6). Elsevier: 575–84. doi:10.1016/j.ijmactools.2010.02.006.

Sosa, Manuel E., Steven D. Eppinger, and Craig M. Rowles. 2003. "Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions." *Journal of Mechanical Design* 125 (2): 240. doi:10.1115/1.1564074.

Spicer, P., Y. Koren, M. Shpitalni, and D. Yip-Hoi. 2002. "Design Principles for Machining System Configurations." *CIRP Annals - Manufacturing Technology* 51 (1): 275–80. doi:10.1016/S0007-8506(07)61516-9.

Spicer, P., D. Yip-Hoi, and Y. Koren. 2005. "Scalable Reconfigurable Equipment Design Principles." *International Journal of Production Research* 43 (22): 4839–52. doi:10.1080/00268970500183042.

Suh, Nam P. 1998. "Axiomatic Design Theory for Systems." *Research in Engineering Design* 10 (4): 189–209. doi:10.1007/s001639870001.

Takus, David A., and David M. Profozich. 1997. "ARENA SOFTWARE TUTORIAL." In *Proceedings of the 1997 Winter Simulation Conference Takus and Profozich*, 541–44.

Terkaj, W., T. Tolio, and A. Valente. 2009. "Focused Flexibility in Production Systems." In *Changeable and Reconfigurable Manufacturing Systems*, edited by Hoda A. ElMaraghy, 4:435. Springer Series in Advanced Manufacturing. London: Springer London. doi:10.1007/978-1-

84882-067-8.

- Tracht, K, and S. Hogleve. 2012. "Decision Making During Design and Reconfiguration of Modular Assembly Lines." In *Enabling Manufacturing Competitiveness and Economic Sustainability*, 105–10. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-23860-4_17.
- Valente, A., and E. Carpanzano. 2011. "Development of Multi-Level Adaptive Control and Scheduling Solutions for Shop-Floor Automation in Reconfigurable Manufacturing Systems." *CIRP Annals - Manufacturing Technology* 60 (1): 449–52. doi:10.1016/j.cirp.2011.03.036.
- Valente, Anna, Andrea Cataldo, and Emanuele Carpanzano. 2013. "A Dispatching Algorithm and Software Tool for Managing the Part Flow of Reconfigurable Transportation System." In *Procedia CIRP*, 7:497–502. Elsevier B.V. doi:10.1016/j.procir.2013.06.022.
- Wang, Wencai, and Yoram Koren. 2012. "Scalability Planning for Reconfigurable Manufacturing Systems." *Journal of Manufacturing Systems* 31 (2). The Society of Manufacturing Engineers: 83–91. doi:10.1016/j.jmsy.2011.11.001.
- Weilkiens, Tim. 2007. *Systems Engineering with SysML/UML. Systems Engineering with SysML/UML*. Morgan Kaufmann OMG Press Morgan. doi:10.1016/B978-0-12-374274-2.X0001-6.
- Wiendahl, H.-P., Hoda. ElMaraghy, P. Nyhuis, M.F. Zäh, H.-H. Wiendahl, N. Duffie, and M. Brieke. 2007. "Changeable Manufacturing - Classification, Design and Operation." *CIRP Annals - Manufacturing Technology* 56 (2): 783–809. doi:10.1016/j.cirp.2007.10.003.
- Wiendahl, H. P., H. A. ElMaraghy, P Nyhuis, M. F. Zäh, H. H. Wiendahl, N Duffie, and M Brieke. 2007. "Changeable Manufacturing - Classification, Design and Operation." *CIRP Annals - Manufacturing Technology* 56 (2): 783–809. doi:10.1016/j.cirp.2007.10.003.
- Xiaobo, Zhao, Wang Jiancai, and Luo Zhenbi. 2000. "A Stochastic Model of a Reconfigurable Manufacturing System Part 1: A Framework." *International Journal of Production Research* 38 (10): 2273–85. doi:10.1080/00207540050028098.
- Xiaobo, Zhao, Jiancai Wang, and Zhenbi Luo. 2000. "A Stochastic Model of a Reconfigurable Manufacturing System Part 2: Optimal Configurations." *International Journal of Production Research* 38 (12): 2829–42. doi:10.1080/002075400411501.
- Xing, Bo, Wenjing Gao, S. Tlale Nkgatho, and Glen Bright. 2007. "Design and Application of Reconfigurable Manufacturing Systems in Agile Mass Customization Manufacturing Environment." In *International Conference on Industrial Engineering and Systems Management*.
- Yigit, A S, A G Ulsoy, and A Allahverdi. 2002. "Optimizing Modular Product Design for Reconfigurable Manufacturing." *Journal of Intelligent Manufacturing* 13 (4): 309–16. <http://www.scopus.com/inward/record.url?eid=2-s2.0-0036681485&partnerID=40&md5=8f2ecdfe51e002fbafb9874242a37e8f>.
- Yigit, Ahmet S., and Ali Allahverdi. 2003. "Optimal Selection of Module Instances for Modular Products in Reconfigurable Manufacturing Systems." *International Journal of Production Research* 41 (17): 4063–74. doi:10.1080/0020754031000149220.
- Youssef, Ayman M.a., and Hoda a. ElMaraghy. 2008. "Performance Analysis of Manufacturing Systems Composed of Modular Machines Using the Universal Generating Function." *Journal of Manufacturing Systems* 27 (2). The Society of Manufacturing Engineers: 55–69. doi:10.1016/j.jmsy.2008.07.003.
- Youssef, Ayman M. A., and Hoda A. ElMaraghy. 2006. "Assessment of Manufacturing Systems Reconfiguration Smoothness." *The International Journal of Advanced Manufacturing*

Technology 30 (1–2): 174–93. doi:10.1007/s00170-005-0034-9.

Youssef, Ayman M A, and Hoda A. ElMaraghy. 2007. “Optimal Configuration Selection for Reconfigurable Manufacturing Systems.” *International Journal of Flexible Manufacturing Systems* 19 (2): 67–106. doi:10.1007/s10696-007-9020-x.

Thèse de Doctorat

Khaled LAMECHE

Proposition d'une méthodologie pour la conception des systèmes de production reconfigurables et d'un outil associé d'aide à la décision par simulation de flux

Proposal of a methodology for the design of reconfigurable production systems and an associated flow simulation tool to support decision

Résumé

Actuellement, le marché est caractérisé par un haut niveau de compétitivité et des variations très fréquentes. L'examen critique des systèmes manufacturiers conventionnels révèle que ces systèmes ne sont pas capables de répondre aux exigences imposées au marché actuel; ces exigences sont principalement le coût et la qualité des produits et la réactivité du système. Par conséquent, la mise en place d'un nouveau paradigme de système manufacturier capable de répondre à ces exigences est nécessaire. Le système manufacturier reconfigurable ou le RMS est ce nouveau paradigme; il est censé être assez réactif pour faire face aux changements soudains du marché tout en gardant la qualité des produits à un coût bas. Le principal défi des RMS est leur conception. La plupart des méthodes proposées dans la littérature n'abordent pas le problème de conception d'un RMS dans son ensemble; elles traitent une partie du problème. Dans cette thèse, on propose une méthodologie générique de conception d'un RMS basée sur les principes de l'ingénierie systèmes. Cette méthodologie supporte la conception d'un RMS tout au long du processus de développement. Elle est dérivée principalement de la norme ISO/IEC/IEEE-15288.

Mots clés : Systèmes Manufacturiers Reconfigurables, Ingénierie des Systèmes, ISO/IEC/IEEE-15288, Méthodologie, Conception, Modularité, Simulation.

Abstract

Nowadays, the market is characterized by a high level of competitiveness and very frequent and sudden variations in the production context. The critical review of the existing manufacturing paradigms which are the dedicated manufacturing lines DMLs and the flexible manufacturing systems FMSs reveals that these systems are not capable of fulfilling the requirements imposed by the actual market; these requirements are mainly resumed in cost, quality and reactivity. Therefore, the need for a new manufacturing paradigm that could fulfill these requirements has arisen. Reconfigurable Manufacturing System or RMSs is this new paradigm; it is supposed to be reactive enough to cope with the sudden changes of the market while keeping the products quality high with a low cost. The main challenge in RMS is their design. Most of the proposed methods in the literature do not address the RMS design issue as a whole; they treat just part of the problem. Hence, as a contribution, we propose in this paper, a generic RMS design methodology based on systems engineering SE. This methodology will support the RMS design along the development process. It is based specifically on the standard ISO/IEC/IEEE15288.

Key Words: Reconfigurable Manufacturing Systems RMSs, Systems Engineering SE, ISO/IEC/IEEE-15288, Methodology, Design, Modularity, Simulation.