



HAL
open science

Modélisation et fouille des processus en vue d'assister la prise de décisions dans le contexte de la conception et la supervision des systèmes

Widad Es Soufi

► To cite this version:

Widad Es Soufi. Modélisation et fouille des processus en vue d'assister la prise de décisions dans le contexte de la conception et la supervision des systèmes. Génie des procédés. Ecole nationale supérieure d'arts et métiers - ENSAM, 2018. Français. NNT : 2018ENAM0067 . tel-02180539

HAL Id: tel-02180539

<https://pastel.hal.science/tel-02180539v1>

Submitted on 11 Jul 2019

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.

École doctorale n° 432 : Sciences des Métiers de l'ingénieur

Doctorat ParisTech

THÈSE

pour obtenir le grade de docteur délivré par

l'École Nationale Supérieure d'Arts et Métiers

Spécialité doctorale "Conception"

présentée et soutenue publiquement par

Widad ES-SOUFI

le 21 Décembre 2018

Modélisation et fouille des processus en vue d'assister la prise de décisions dans le contexte de la conception et la supervision des systèmes

Directeur de thèse : **Lionel ROUCOULES**

Co-encadrante de thèse : **Esma YAHIA**

Jury

Mme. Nada MATTA,

M. Laurent GENESTE,

Mme. Elise VAREILLES,

M. Lionel ROUCOULES,

Mme. Esma YAHIA,

M. Christophe BELLAMY,

Professeur, Université de technologie de Troyes

Professeur, ENI Tarbes

Maître de conférences HDR, IMT Mines Albi-Carmaux

Professeur des Universités, LISPEN, Art et Métiers ParisTech

Maître de Conférences, LISPEN, Art et Métiers ParisTech

Chef de service Réseau & Interventions Clients, R-GDS

Rapporteur

Rapporteur

Examinatrice

Examinateur

Examinatrice

Examinateur

”Keep calm and eat cookies (...) No cry because cookie is finished. Smile because cookie happened.”

— Cookie Monster



”If you can’t explain it simply, you don’t understand it well enough.”

— Albert Einstein



À mes parents **Hoummad Es-Soufi** et **Noura Lamjounah** -les parents les plus "cool & lovely" au monde- pour leur amour inconditionnel et leur support infini, pour tous les fous rires et tous les moments de pur bonheur qu'ils m'offrent; et avant tout, pour m'avoir transformée d'un simple design conceptuel à un produit final implémenté..

À mes grands-pères qui n'ont jamais quitté mes pensées : **El Arbi Lamjounah** et **Issa Es-Soufi**. Ils auraient été très fiers de leur "petite rebelle"..

À **Pluto**, qui a été expulsé du système solaire en 2006. Il restera toujours planète à mes yeux.. À tous les **extraterrestres** que je n'ai pas encore rencontrés..

À toutes les **montagnes** que j'ai escaladées et tous les **arbres** que j'ai grimpés..

À tous mes **professeurs**, de la crèche à l'université.. À **Lionel Roucoules** et **Esmâ Yahia**, les meilleurs!

À mes **meilleurs amis** qui ont la même déficience mentale que moi.. À tous ceux qui sont positifs, souriants, dynamiques, créatifs, cool, différents et **originaux**..

À **CELUI** qui m'a toujours aimée extraordinairement et inépuisablement; **CELUI** qui a pu graver à jamais son **NOM** dans ma petite mémoire.

Remerciements

J'avoue que la rédaction de cette partie de remerciements est la plus délicate ! Pourquoi ? Bein, vu que je ne veux oublier personne, je finirai par remercier les madeleines que j'ai mangées tous les jours à 17h. Non, je rigole !

J'adresse mes premiers remerciements aux deux rapporteurs de ce travail de thèse. Nada Matta, Professeur des Universités à l'Université de Technologie de Troyes, qui m'a également fait l'honneur de présider le jury de thèse, et Laurent Geneste, Professeur des Universités à l'École Nationale d'Ingénieurs de Tarbes. Nada et Laurent, merci d'avoir pris le temps de lire mon manuscrit malgré vos emplois de temps surchargés. Merci pour vos rapports intéressants, vos questions et vos remarques m'ouvrent de nouvelles perspectives de recherche.

Mes remerciements s'adressent également à Elise Vareilles, Maître de conférences HDR à l'IMT Mines Albi-Carmaux et Christophe Bellamy, Chef de Service Réseau & Interventions Clients à R-GDS, pour l'honneur qu'ils m'ont fait en acceptant de faire partie de mon jury de thèse. Merci Elise et Christophe pour vos questions et remarques, elles ont apporté un point de vue extérieur sur mes travaux.

Nada, Laurent, Elise et Christophe, j'espère avoir l'opportunité de collaborer avec vous à l'avenir.

Ce travail de thèse s'inscrit dans le cadre du projet national Gontrand, visant une supervision intelligente des réseaux de gaz favorisant l'injection du gaz vert. Je souhaite remercier les sociétés REGAZ, GDS et GRDF pour leur collaboration.

Mes remerciements les plus sincères, les plus chaleureux et les plus tendres, s'adressent à mon directeur et encadrante de thèse : Lionel Roucoules, Professeur des Universités aux Arts et Métiers ParisTech d'Aix-en-Provence et directeur du Laboratoire d'Ingénierie des Systèmes Physiques et Numériques, et Esmâ Yahia, Maître de Conférences aux Arts et Métiers ParisTech d'Aix-en-Provence.

Lionel, merci pour ta gentillesse, pour tes multiples conseils et pour toutes les heures que tu as consacrées à diriger cette thèse. En plus de ta personnalité très appréciée par toute l'équipe, j'ai beaucoup admiré ta grande disponibilité, ton dynamisme, le champ de vision très vaste et impressionnant que tu as et ta maturité scientifique que j'espère avoir un jour. Lionel, je te remercie profondément pour des millions de raisons, tu as encadré et évalué mon travail d'une manière très approfondie et constructive, tes remarques pragmatiques m'ont vraiment permis de conserver le sens du concret. Merci pour ton indéfectible soutien et surtout pour ta grande confiance. Je remercie également la magnifique Ethele Roucoules pour ses encouragements et j'espère que les petits Roucoules marcheront sur les "traces" de leur papa.

Esmâ, merci de m'avoir chaleureusement accueillie quand j'ai atterri à Aix-en-Provence et de m'avoir aidée à bien m'installer. Merci d'avoir été toujours très disponible malgré ton emploi de temps très chargé entre les préparations des cours et les enfants. Merci d'avoir raté le train du sommeil plusieurs fois pour relire les articles et les chapitres de la thèse. Esmâ, ta lecture est merveilleusement minutieuse, tu finis toujours par me trouver des erreurs dans les documents que je t'envoie (même quand je les relis plusieurs fois avant de les envoyer). Merci pour ton suivi permanent et tes critiques fondées qui m'ont permis d'avancer dans mon projet. Merci parce que ton sourire et tes mots m'ont toujours motivée, merci de m'avoir soutenue avant et après mon opération. Je tiens également à remercier Hatem pour sa gentillesse et ses encouragements et je ne veux surtout pas oublier mes chers Adam et Inès qui sont très adorables et intelligents. Adam et Inès, je vous aimerai pour toujours et j'espère bien vous voir docteurs, tout comme papa et maman.

Lionel et Esma, je ne pourrais jamais vous remercier assez pour m'avoir appris les ABC de la recherche appliquée. Vous avez fait de ma thèse une expérience très enrichissante et agréable, qui restera gravée dans ma mémoire (de poisson rouge). J'espère que nous aurons maintes occasions de travailler ensemble à l'avenir.

Je souhaite également adresser mes très chaleureux et tendres remerciements à Grazyna Cauquil, ma deuxième maman. Grazyna, heureusement qu'il existe encore des personnes comme toi qui rendent le monde meilleur ! Tu es, contrairement à certaines responsables RH, très appréciée de TOUS les doctorants, très serviable, organisée, positive, souriante et toujours motivée. Grazyna, ta joie de vivre et ta spontanéité m'ont motivée encore plus. Merci pour ta disponibilité, tes conseils précieux et pour ton soutien inconditionnel. Tu as certainement participé à l'aboutissement de ce travail de recherche.

Jean-Philippe Pernot, merci pour les discussions très constructives que nous avons eues. J'apprécie bien ton esprit scientifique très ouvert, cependant, tu dois savoir que même en portant une belle cravate orange qui respecte la charte des arts et métiers, le BOSS est toujours Lionel Roucoules.

Je remercie également Philippe Veron pour ses encouragements et ses conseils. Philippe, mise à part tes qualités en tant que professeur, tu dois sérieusement t'approcher d'une agence de casting pour faire la pub de shampooing, tes cheveux sont merveilleusement beaux !

Je ne peux pas oublier de remercier Julien Gomand. Merci Julien pour tes encouragements et pour ta super bonne humeur, Luna a de la chance d'avoir un papa comme toi !

Je remercie également George Moraru (et le petit Gabriel), François Malburet, Lionel et Émilie Martin pour leur sympathie et leurs encouragements.

J'exprime tous mes remerciements à l'ensemble des membres du laboratoire LISPEN et de l'école des Arts et Métiers ParisTech d'Aix-en-Provence, notamment le corps professoral et administratif, l'équipe des ingénieurs et des techniciens (Quentin, Olivier, Alex, Thibaud, Hervé, Pascal, Fabienne, Vanessa, Monique et Richard), l'équipe de la bibliothèque (Cécile et Emilie) et les agents de propreté et d'hygiène. Au sein de cette grande équipe soudée et dans laquelle il fait bon vivre, j'ai eu l'occasion d'apprendre beaucoup de choses tant au plan professionnel qu'au plan personnel, j'espère simplement que mon sérieux et mon humour auront participé à créer cette bonne ambiance et surtout j'espère qu'ils sauront la faire perdurer pour les nouveaux arrivants. Ce fut un réel plaisir de vous rencontrer.

C'est vrai que nous avons tous besoin d'un exemple, d'une personne à qui l'on aimerait ressembler. Pour certaines personnes, ce sont des acteurs, des écrivains ou toute autre personnalité connue. Je me suis alors demandée un jour qui était mon exemple, la réponse été simple et évidente : c'est mon père Hoummad Es-Soufi. Papa, tu es parfait à tous points de vue et tu fais toute ma fierté. Tu as toujours aimé les choses que j'aime, merci d'avoir joué à la boxe et au handball avec moi. Merci parce que tu m'as transmis ta joie de vivre, ta créativité et ton amour pour la photographie et pour l'art en général. Merci parce que tu as toujours été mon meilleur ami, tu m'as toujours fait rire, félicité et récompensé pour mes victoires. Merci parce que tu m'encourage toujours à aller de l'avant et à développer mes talents. Tu m'as toujours appris qu'il faut cultiver son propre jardin et qu'il faut être heureux, merci papa de m'avoir aidé à mûrir et à m'épanouir dans mon travail de recherche, merci surtout de m'avoir appris à persévérer.

Je sais aussi qu'il y a des femmes qui ne changent pas avec le temps et qui rendent le monde meilleur ; maman, Noura Lamjounah, est l'une de ces femmes exceptionnelles, qui ont le talent de faire aimer la vie après l'avoir donné. Elle est pourvue d'une grande tendresse, bonté et sagesse, elle m'a appris que l'on doit toujours continuer d'avancer même après avoir pensé qu'il nous serait impossible de le faire. Merci maman de m'avoir enseigné mes premiers ABC, merci de m'avoir bien expliqué les Maths et de m'avoir aidé à apprendre comment vider mon assiette à table, laver les vaisselles et préparer mon premier cake à l'âge de presque cinq ans. Si maintenant je fais de la bonne cuisine, sache que c'est entièrement grâce à toi.

Papa, maman, merci de m'avoir supporté et appuyé durant toutes ces années. De m'avoir inculqué de VRAIES valeurs et de m'avoir permis de devenir qui je suis aujourd'hui, merci d'avoir été là et de me

donner jour après jour autant d'amour et de soutien. Merci tout simplement d'être mes parents.

Je remercie mes adorables sœurs et frères : Maroua, Safae, Doha, Mohamed Amine et Abdelkader Es-Soufi. Vous avez réussi à me supporter tout au long de mon parcours universitaire. Merci pour tout ce que vous m'avez apporté au quotidien d'attention, de soutien et d'amour. Merci pour tous les moments de folie et de fous rires. Maintenant, je passe le relais à ceux qui n'ont pas encore terminé leurs études. Go Go Go !

Je remercie mon très cher oncle Nacer Kouider M'Sirdi, sa merveilleuse femme Assia M'Sirdi et mes cousines Ghislaine, Sabria et Soukaina M'Sirdi pour des millions de raisons. Vous m'avez toujours encouragée et supportée, votre appui moral et votre bonne humeur ont été des éléments de motivation indispensables. Merci également à toute ma famille pour l'amour et le support dont ils ont fait preuve. Je tiens également à remercier très vivement Catherine et Brad Dixon, pour leur amour et leur soutien inconditionnel. Par la même volonté et la même chaleur, je tiens à remercier Dr. Cecil Joe d'avoir enrichi mes connaissances et de m'avoir guidée, merci encore plus pour sa sympathie et pour les échanges amicaux que nous avons eus. J'ai une pensée très particulière à mes très chers grands-pères El Arbi Lamjounah et Issa Es-Soufi qui n'auront pas pu assister à la fin de mes études, mais à qui je veux bien exprimer mon amour et toute ma gratitude.

Je ne peux évidemment que remercier l'équipe des doctorants avec qui j'ai partagé ces années de thèse. Tout d'abord, Anthony Geromin pour tous les moments de fous rires et pour toutes les discussions existentielles, bon parfois intéressantes quand même, que nous avons eues. Anthony, fiston, merci pour ton amour, pour tes encouragements et pour ton soutien indéfectible. Merci parce que tu as fait de mes pauses des moments très plaisants et adorables. Anthony, sache que tu es différent, spécial et original. Je sais que je te l'avais déjà dit, mais je le redis ici : Tu fais partie des personnes qui seront gravées dans ma mémoire.

Mes profonds et chaleureux remerciements s'adressent à Ahmed Youssef (Migallo). Ahmed, merci pour tes encouragements, pour ta précieuse amitié, pour ta bonne humeur et pour tous les moments de rigolade. Ces années de thèse auront certainement été moins agréables sans ta présence. Merci pour le maamoul et les repas aux saveurs du Yémen (surtout le riz) que tu nous as préparés, tu feras un très bon chef cuisinier. Je remercie également ta merveilleuse femme Arzaq, et ton petit prince Youssef.

Je ne saurais oublier dans ces remerciements : Mohamed Kabibou (pour son sens de l'humour et son goût musical drossien), Meryem Hayyani (pour les tajines qu'elle m'a préparés), David Deloffre (pour le peuplement de la France), Jonathan Rodriguez (pour tous les moments de rigolade), Maxime Gélinau (pour nos merveilleux échanges), Romain Pinquié (ou Cocoa Diko, pour son humour et son caractère très agréable), Yosbel Galavis (pour sa très bonne humeur), Hao Hu, Ayman et Nouha Sahnoun, Théophile Durdan, Ghazanfar ali Shah, Benjamin Vaissier, Laurent Vallet, François Marchand, Nathanael Dougier, Kenza Amzil et Lise Kim. Merci pour votre énorme gentillesse et pour vos encouragements. Je sens que vous avez tous un très gros potentiel ; vous ferez, sans doute, une super carrière.

Je veux manifester mes très sincères reconnaissance et remerciements à mes meilleurs amis : Safae Dahmani, Dounia Mabrouk, Ouafae Amrane, Sarah El Gamoussi et Mouhcine Kahziz. Merci pour votre soutien, votre attention et votre sincère amitié. Vous êtes tous très serviables, différents, originaux, dynamiques, positifs et très généreux. C'est incroyable comment nous avons une excellente chimie entre nous. Merci d'avoir rendu les moments ordinaires extraordinaires. Une simple journée peut devenir spéciale grâce à vous, un court arrêt dans un café fera toujours ressortir les meilleures histoires. Je suis tout simplement heureuse d'avoir des Best Friends Forever, avec lesquels il n'y a jamais un moment d'ennui.

J'adresse aussi mes vifs remerciements à kaoutar chemlal, Amal Sadqi, Aziz et Anass (mes amis d'enfance avec lesquels je n'ai que de bons souvenirs), Nadia El Azzaoui et Zineb Boutaarourt (pour leur amitié et pour tous les moments inoubliables que nous avons vécus ensemble).

Je remercie toutes les personnes qui n'ont pas été citées mais qui auraient mérité de l'être. Je souhaite également remercier mes cellules nerveuses qui, malgré la forte entropie qu'elles ont connue, m'ont accompagnée jusqu'au bout. Et enfin, j'adresse mes profonds remerciements à toutes les madeleines que j'ai mangées tous les jours à 17h.



Widad Es-Soufi

Table des matières

Table des matières	xii
Table des figures	xiii
Liste des tableaux	xxi
1 Introduction générale	1
1.1 Motivations : pourquoi a-t-on besoin d'être assisté lors de la conception ou la supervision d'un système ?	2
1.2 Objectifs	2
1.3 Hypothèse et approche	2
1.4 Contributions	3
1.5 Structure du manuscrit	4
I PROBLÉMATIQUE ET ÉTAT DE L'ART GÉNÉRAL	7
2 Contexte, Objectifs et Approche	9
2.1 Contexte industriel	10
2.1.1 Le besoin d'être compétitif sur le marché	10
2.1.2 Comment être compétitif sur le marché	10
2.1.3 Discussion	14
2.2 Processus de développement produit (PDP)	15
2.2.1 Étapes du PDP	15
2.2.2 Maîtriser le PDP	16
2.2.3 Discussion : contexte de recherche	18
2.3 Problématique, question de recherche, hypothèse et caractéristiques du système recherché	20
2.3.1 Problématique : les besoins industriels	20
2.3.2 Objectifs scientifiques et question de recherche	23
2.3.3 Hypothèse de travail	24
2.3.4 Caractéristiques générales du système recherché	25
3 État de l'art général : Travaux académiques et outils permettant l'aide à la décision	27
3.1 Introduction : la décision sous le microscope	29
3.2 État de l'art académique : travaux de recherche concernant la prise de décision	30
3.2.1 Problème de décision	31
3.2.2 Théorie de décision	32
3.2.3 Quelques approches de la prise de décision	34
3.2.4 Quelques méthodes et techniques de la prise de décision lors de la conception et supervision des systèmes	37
3.2.4.1 Méthodes dédiées à la conception des systèmes	37
3.2.4.2 Méthodes dédiées à la supervision des systèmes	40
3.2.5 Processus de prise de décision	42

3.2.6	Quelques systèmes intelligents supportant la prise de décision	44
3.2.6.1	Les systèmes experts	44
3.2.6.2	Les systèmes de recommandation	47
3.2.6.3	Les systèmes de support de prise de décision (DSS)	50
3.2.7	Synthèse et discussion de l'état de l'art académique	53
3.3	État de l'art commercial : outils supportant la prise de décision	55
3.3.1	Blueworks Live d'IBM	55
3.3.2	SAS Event Stream Processing	57
3.3.3	Autres outils supportant la prise de décision	58
3.3.4	Synthèse et discussion de l'état de l'art commercial	59
3.4	Synthèse et conclusion de l'état de l'art académique et commercial	60
4	État de l'art général : processus de prise de décision dans les milieux industriels	61
4.1	Dans les industries aéronautiques et spatiales	63
4.1.1	Prise de décision lors de la gestion de risque à la NASA	63
4.1.2	Prise de décision lors de la conception des systèmes à la NASA	66
4.1.3	Prise de décision pendant le vol (lors de la supervision)	68
4.2	Dans les industries nucléaires et de l'énergie	70
4.2.1	Prise de décision à l'U.S.NRC	70
4.2.2	Prise de décision à Schneider Electric	73
4.3	Selon quelques organisations et associations contribuant à la prise de décision	75
4.3.1	L'INCOSE	75
4.3.2	Le BKCASE	76
4.3.3	L'OMG	78
4.3.4	L'IIBA	81
4.3.5	L'ISO	82
4.4	Synthèse et discussion de l'état de l'art industriel	83
4.5	Prononciation du verdict : adopter une solution existante ou concevoir une autre ?	83
II PROPOSITION : SYSTÈME D'AIDE À LA DÉCISION INTELLIGENT ET DYNAMIQUE (IDSS)		85
5	Architecture modulaire du système intelligent d'aide à la décision	87
5.1	Introduction	88
5.1.1	Qu'est ce que l'architecture des systèmes ? Et comment nous la définissons dans notre contexte ?	88
5.1.2	État de l'art sur les méthodes d'architecture des systèmes	88
5.1.3	Synthèse et discussion	92
5.2	Architecture du système intelligent d'aide à la décision	92
5.2.1	Point de vue opérationnel (boîte noire)	93
5.2.2	Point de vue logique (boîte blanche)	93
5.2.3	Point de vue physique	95
5.2.4	Points de vue processus et scénarios	96
5.3	Synthèse et conclusion	98
6	Modules de modélisation des processus et de traçabilité des décisions	101
6.1	Module de modélisation des processus et des données produit	103
6.1.1	Introduction : pourquoi avons-nous besoin de modéliser ?	103
6.1.2	État de l'art sur les techniques de modélisation des processus	105
6.1.2.1	Comment est définie la modélisation des processus dans notre contexte de recherche ?	105

6.1.2.2	Travaux de recherche ayant contribué à la méta-modélisation des processus	107
6.1.3	État de l'art sur les techniques de modélisation des données produit	108
6.1.3.1	Comment est définie la modélisation des données produit dans notre contexte de recherche ?	108
6.1.3.2	Quels sont les travaux de recherche qui ont contribué à la méta-modélisation des produits ?	108
6.1.4	Synthèse et discussion de l'état de l'art sur la modélisation des processus et des données produit	110
6.1.4.1	Concernant le choix du langage de modélisation	110
6.1.4.2	Concernant le choix du méta-modèle de processus	110
6.1.4.3	Concernant le choix du méta-modèle des données produit	112
6.1.5	Proposition : méta-modèle PP de processus-produit	114
6.1.5.1	Identification des différents cas d'utilisation du méta-modèle PP	115
6.1.5.2	Sémantique du méta-modèle PP	117
6.2	Module de traçabilité des décisions	120
6.2.1	Introduction	120
6.2.2	État de l'art sur les techniques de traçabilité	121
6.2.2.1	Comment est définie la traçabilité des décisions dans notre contexte de recherche ?	121
6.2.2.2	Travaux de recherche ayant contribué à la méta-modélisation des traces	123
6.2.3	Synthèse et discussion de l'état de l'art sur les techniques de traçabilité	125
6.3	Conclusion	126
7	Module d'aide à la décision	127
7.1	Introduction : vers des systèmes intelligents d'aide à la décision	129
7.2	Comment est définie l'aide à la décision dans notre contexte de recherche ?	129
7.3	État de l'art sur quelques techniques de l'IA et de la BI supportant la prise de décision	132
7.3.1	Quelle est la différence entre l'intelligence artificielle, la fouille de données, l'apprentissage automatique, la fouille de processus et la business intelligence ?	132
7.3.2	Techniques de fouille de données (Data mining)	134
7.3.2.1	Fouille des règles d'association (Association rule mining)	134
7.3.2.2	Fouille des patterns séquentiels (Sequential pattern mining)	135
7.3.3	Techniques de l'apprentissage automatique (Machine learning)	137
7.3.3.1	Apprentissage supervisé (Supervised learning)	138
7.3.3.2	Apprentissage non-supervisé (Unsupervised learning)	138
7.3.3.3	Apprentissage semi-supervisé (semi-supervised learning)	140
7.3.3.4	Apprentissage par renforcement (Reinforcement learning)	141
7.3.4	Techniques de fouille de processus (Process mining)	143
7.3.4.1	Découverte des processus (Process discovery)	143
7.3.4.2	Contrôle de conformité (Conformance checking)	144
7.3.4.3	Amélioration des processus (Enhancement)	146
7.3.5	Synthèse et discussion de l'état de l'art sur les techniques de l'intelligence artificielle et de la business intelligence supportant la prise de décision	147
7.4	Module d'aide à la décision : Aide à la décision globale et locale	150
7.4.1	Processus de l'aide à la décision et exemple d'illustration	150
7.4.2	Préparation du journal des événements	153
7.4.3	Phase d'exploration des différents patterns existants dans un journal des événements	158
7.4.4	Phase de la découverte des règles de décision	161
7.4.4.1	Sous-phase 1 : découverte du modèle réseau de Pétri à partir du journal des événements	161
7.4.4.2	Sous-phase 2 : fouille des règles de décision (Decision Mining)	165
7.4.5	Prédiction des paramètres des activités les plus adaptés au contexte de la décision	171

7.4.6	Synthèse et conclusion du module d'aide à la décision	174
7.5	Module de gestion des changements contextuels : vers un IDSS dynamique	174
7.5.1	Qu'est ce que la gestion des changements ?	174
7.5.2	État de l'art sur les techniques de gestion des changements contextuels	175
7.5.3	Discussion des techniques de gestion des changements contextuels	175
7.5.4	Proposition : application permettant la gestion dynamique des changements contextuels	176
7.6	Synthèse et conclusion	177
III VÉRIFICATION & VALIDATION, CONCLUSION & PERSPECTIVES		179
8	Vérification & Validation de l'IDSS proposé	181
8.1	Introduction	183
8.1.1	Besoin de vérifier et valider son système	183
8.1.2	Qu'est ce que la V&V et comment nous la définissons dans notre contexte ?	183
8.2	État de l'art sur les techniques de vérification et de validation des systèmes	184
8.3	Discussion de l'état de l'art sur les techniques de V&V : vers un cadre de V&V pour les IDSSs	189
8.4	Vérification et validation de l'IDSS proposé	190
8.4.1	Validité des données d'étude	190
8.4.2	Validation conceptuelle et validation structurelle théorique	191
8.4.3	Vérification et validité interne	191
8.4.4	Validité de l'IHM	194
8.4.5	Le projet industriel ayant créé le besoin de concevoir un système IDSS	194
8.4.6	Évaluation des utilisateurs potentiels	196
8.4.7	Validation de performance empirique	198
8.4.8	Validation de performance théorique	198
8.4.9	Capacité de répondre aux caractéristiques souhaitées	198
8.5	Synthèse et conclusion	199
9	Conclusion générale et perspectives	201
9.1	Synthèse générale	202
9.1.1	Attendez ! Mais pourquoi devons nous proposer un système IDSS ?	202
9.1.2	D'accord, et comment nous avons procédé à la proposition du système IDSS ?	202
9.1.3	Parfait ! Mais a-t-il été vérifié cet IDSS ? Quelles sont ses limites ?	203
9.2	Vers de nouvelles perspectives	204
9.2.1	Perspectives à court terme	204
9.2.2	Perspectives à moyen terme	205
9.2.3	Perspectives à long terme	205
9.3	Conclusion de la conclusion	207
IV BIBLIOGRAPHIE, ANNEXE, LISTE DES PUBLICATIONS ET DES NOTES		209
Liste des publications		211
Bibliographie		213
Annexe : Méta-modèles étudiés de processus, de produit et de traçabilité		243
Liste des notes		261

Table des figures

1.1	Représentation IDEF0 de l'approche globale du présent travail de recherche (les deux flèches en pointillé ne font pas partie du formalisme IDEF0, elles ont été introduites pour montrer la l'interaction indirecte entre les activités)	3
1.2	Représentation BPMN de la structure du présent manuscrit de thèse (décrite en utilisant le formalisme BPMN [OMG, 2013]).	4
1.3	Les deux niveaux de l'état de l'art que nous considérons dans le présent travail de recherche..	5
1.3.a	Etat de l'art général : le système est vu comme une boîte noire exécutant des fonctions.	5
1.3.b	Etat de l'art par chapitre : le système est vu comme une boîte grise dont chacun des composants réaalise une fonction.	5
1.4	Structure de la première partie du manuscrit (décrite en utilisant le formalisme BPMN [OMG, 2013]).	8
2.1	Les phases de l'innovation produit selon Roozenburg and Eekels [1995].	14
2.2	Le triangle {Compétitivité, Innovation, Développement produit}.	15
2.3	Définition du processus de développement produit dans le cadre de notre contexte.	16
2.4	Carte conceptuelle de la supervision et la maintenance des produits matériels, des services et des systèmes.	16
2.5	Évolution des objectifs industriels.	19
2.6	Influence des décisions prises sur les coûts [Gautier and Glard, 2000] (adapté de [Berliner and Brimson, 1988])	20
2.7	Coûts des corrections en fonction du temps [INCOSE, 2015]	21
2.8	Pourcentage du temps pris par les différentes activités du PDP selon [Oehmen and Rebentisch, 2010]	22
2.9	Et si on maîtrisait notre chemin (des décisions prises lors de la conception ou supervision des systèmes) comme a fait le Petit Poucet.	24
2.9.a	Maîtriser le chemin pour savoir revenir.	24
2.9.b	Exemple des chemins décisionnels de la conception d'une voiture.	24
2.10	Architecture modulaire du système recherché	25
3.1	Représentation IDEF0 des étapes de résolution d'un problème de décision donné.	31
3.2	Les caractéristiques des décisions simples et complexes [Clark et al., 2009]	33
3.2.a	La problèmes de décision simples et complexes	33
3.2.b	Les causes de complexité de la décision	33
3.3	Étapes de l'approche hybride pour la prise de décisions [Montagna, 2011]	40
3.4	Quelques méthodes de supervision basées sur l'historique [Venkatasubramanian et al., 2003]	42
3.5	Processus de prise de décision selon Mintzberg et al. [1976]	43
3.6	Processus de prise de décision générique pouvant être exécuté à n'importe quel phase du cycle de vie du système [Parnell et al., 2011]	45
3.6.a	Processus de prise de décision des systèmes (SDP)	45
3.6.b	Intervention du processus durant le cycle de vie du système	45
3.7	Composants du système expert selon Hopgood [2001]	46
3.8	Quelques fenêtres du système expert CFMDAS [Mostafa et al., 2012]	46

3.8.a	Boîte de dialogue de l'état de démarrage	46
3.8.b	Fenêtre de localisation	46
3.9	Les différentes techniques de recommandation selon Bobadilla et al. [2013]	49
3.10	Architecture générale d'un DSS [Phillips-Wren et al., 2009]	52
3.11	Amélioration des IDSS avec les techniques de fouilles de données [Liu et al., 2010b]	53
3.12	Système DSS intégrant un DSS (en bas) et un PDG (en haut) [Seletos and Salmon, 2017]	53
3.13	Interface du système Pilote [Guo et al., 2015]	54
3.14	Processus de découverte de décision [Thorpe et al., 2014]	56
3.15	Exemple d'une tâche décisionnelle au sein d'un processus métier dans "Blueworks Live" (adapté de [Thorpe et al., 2014])	57
3.16	Documentation des caractéristiques clés de la décision et sa logique dans "Blueworks Live" (adapté de [Thorpe et al., 2014])	57
3.17	Architecture de <i>SAS Event Stream Processing</i> (adapté de [SAS, 2015])	58
4.1	Évolution du processus de gestion de risque à la NASA	64
4.1.a	Ancien RM : CRM [NASA, 2010b]	64
4.1.b	Nouveau RM : CRM + RIDM [Dezfuli et al., 2010]	64
4.2	Flux d'information pendant l'exécution du processus RIDM [Dezfuli et al., 2010]	64
4.3	Processus de conception des systèmes à la NASA [NASA, 2010b]	66
4.4	Processus d'analyse de décisions à la NASA [NASA, 2010b]	67
4.5	Processus de raffinements successifs [NASA, 2010b]	67
4.6	Effets d'une mauvaise supervision sur la qualité des décisions [Harris and Li, 2017]	69
4.7	Framework réglementaire de la gestion du risque à l'U.S.NRC [U.S.NRC, 2012]	72
4.8	Processus de délibérations (aussi appelé processus décisionnel intégré) [U.S.NRC, 2012]	73
4.9	Processus de prise de décision à Schneider Electric [SE, 2017]	74
4.10	Modèle IPO du processus de gestion de décision selon l'INCOSE [2015], où IPO (Input-Process-Output) est un formalisme permettant de modéliser les processus : https://www.sixsigmadaily.com/input-output-model/	75
4.11	Processus de gestion de décision selon le BKCASE [2017]	77
4.12	La relation entre les trois perspectives du processus de prise de décision selon l'OMG [2016]	79
4.13	Diagramme des exigences de décision [IIBA, 2015; OMG, 2016]	80
4.14	Processus cyclique de la modélisation de décision et le contexte de cette dernière dans la norme DMN	80
4.14.a	Processus de modélisation de décision [DMS, 2016b]	80
4.14.b	Contexte de la décision dans la norme DMN [DMS, 2016a]	80
4.15	Décisions simples : table et arbre de décision montrant les règles d'éligibilité à un prêt [IIBA, 2015]	81
4.15.a	Exemple d'une table de décision	81
4.15.b	Exemple d'un arbre de décision	81
4.16	Structure de la deuxième partie du manuscrit (décrite en utilisant le formalisme BPMN [OMG, 2013]).	86
5.1	Positionnement des activités de l'architecture des systèmes au sein du processus d'ingénierie des systèmes [Boeing, 2005]. Ces activités sont en rouge et sont encadrées par un rectangle avec un contour épais.	89
5.2	Processus d'architecture des systèmes selon l'INCOSE [2015]	89
5.3	Points de vue du frameworks DoDAF [DoD, 2015]	91
5.4	Points de vue du modèle "4+1" de Kruchten [1995]	92
5.5	Abstraction du processus suivi lors de la définition de l'architecture du système intelligent d'aide à la décision	93
5.6	Vue holistique (externe) du système d'aide à la décision à concevoir	94
5.6.a	Représentation de notre système selon le point de vue boîte noire	94

5.6.b Point de vue boîte noire selon Atkins Ltd, Entreprise multinationale spécialisée dans la conception architecturale	94
5.7 Architecture logique de notre système intelligent d'aide à la décision, présentée sous la forme d'un diagramme de classe UML [OMG, 2011].	94
5.8 Architecture physique et logicielle de notre système intelligent d'aide à la décision	96
5.9 Diagramme de cas d'utilisation UML décrivant le scénario typique du système à concevoir	98
5.10 Diagramme de séquence décrivant, dans l'ordre réel d'exécution, les activités du système d'aide à la décision à concevoir	99
5.11 Méta-modèles des diagrammes de cas d'utilisation et de séquence UML montrant leurs concepts et les relations entre eux	100
5.11.aPartie du méta-modèle du diagramme de cas d'utilisation UML [OMG, 2011]	100
5.11.bPartie du méta-modèle du diagramme de séquence UML [OMG, 2011]	100
6.1 Structure du Chapitre 6	102
6.2 Utilisation des modèles de produit pour supporter les processus opérationnels durant le cycle de vie produit [Scholz-Reiter and Krause, 2001]	104
6.3 Les six concepts (désormais notés 6Ws) de Zachman [1987], décrits par Scott Ambler, coach et consultant sénior : http://enterpriseunifiedprocess.com/	105
6.4 Représentation IDEF0 de l'activité de modélisation des processus	106
6.5 Représentation des quatre couches du standard MOF	106
6.6 Évolution des données produit/système par rapport aux activités du processus de conception/supervision	109
6.7 Représentation IDEF0 de la modélisation des données produit dans notre contexte	109
6.8 Passage automatique d'un langage de modélisation à un autre via des transformations ATL.	110
6.9 Comparaison des différents méta-modèles étudiés par rapport à nos critères de recherche. Le méta-modèle que nous cherchons doit occuper toute la surface du radar.	113
6.9.a [Nowak et al., 2004]	113
6.9.b [Ouertani, 2007]	113
6.9.c [Couturier et al., 2014]	113
6.9.d [Moones et al., 2015]	113
6.9.e [Labrousse and Bernard, 2008]	113
6.9.f [Heidari et al., 2011]	113
6.9.g [Thom et al., 2005]	113
6.9.h [Martins and Zacarias, 2015]	113
6.9.i [OMG, 2013]	113
6.9.j [OMG, 2011]	113
6.9.k IDEF0	113
6.10 Résultat de la comparaison des méta-modèles de données produit par rapport à nos critères de recherche	114
6.11 Les deux types de données produits identifiés dans les milieux industriels et scientifiques	115
6.11.aVision boîte blanche	115
6.11.bVision boîte noire	115
6.12 Les différents cas d'usage lors de la modélisation des processus	115
6.13 Représentation IDEF0 de l'activité de modélisation, avec un exemple illustratif	116
6.14 Représentation IDEF0 de l'activité de décision et évolution de l'espace des solutions correspondant	116
6.14.aReprésentation IDEF0 de l'activité de décision, avec un exemple illustratif	116
6.14.bÉvolution de l'espace des solutions lors d'une activité de décision	116
6.15 Proposition : méta-modèle PP de modélisation des processus et des données produit	117
6.16 Exemple de processus contenant les jonctions <i>And</i> , <i>Or</i> et <i>Xor</i>	119
6.17 Processus de traçabilité selon Cleland-Huang et al. [2012]	122
6.17.aPhase de planification de la stratégie de traçabilité	122

6.17.b	Phase de création de la trace	122
6.17.c	Phase de maintenance de la trace	122
6.17.d	Phase de l'utilisation de la trace	122
6.18	Représentation IDEF0 de la traçabilité des décisions dans notre contexte.	123
6.19	Interface Prowhy dédiée à la description du problème (http://www.prowhy.org/prowhy_site/principal/onglet_fonctionalites)	125
6.20	Spécialisation des méta-modèles de trace selon le domaine d'étude.	126
7.1	Structure du Chapitre 7.	128
7.2	Représentation IDEF0 des deux types de la prise de décision. Nous n'avons présenté que la ressource utilisée par la première activité de chaque processus pour ne pas charger la figure.	130
7.3	Les deux niveaux de la prise de décision : local et global. Le premier est exprimé en IDEF0 pour souligner les paramètres de l'activité et le deuxième est exprimé en BPMN pour souligner la présence des jonctions.	131
7.4	Représentation IDEF0 de l'activité de prise de décision	131
7.5	Combinaison des forces de l'apprentissage automatique et de la fouille de données pour construire des systèmes artificiellement intelligents.	133
7.5.a	L'utilisation des techniques de l'apprentissage automatique pour développer les capacités d'un système artificiellement intelligent [Van Duin and Bakhshi, 2017]	133
7.5.b	L'utilisation des techniques de l'apprentissage automatique pour explorer les données [Mayo, 2017]	133
7.5.c	Différence entre les domaines de fouille de données et de l'apprentissage automatique selon https://iotechnologies.com	133
7.6	Différence entre la fouille de processus et la Business Intelligence.	134
7.6.a	Processus de la Business Intelligence selon takefiveconsulting.com	134
7.6.b	Combinaison des techniques de la fouille de données et de la modélisation des processus pour des fins de fouille de processus (fluxicon.com)	134
7.7	Exemple d'une séquence d'événements. Noter que $\langle (a) (b) (d) (e\ g) \rangle$ est une sous-séquence de $\langle (a) (b\ c) (d) (e\ f\ g) \rangle$.	136
7.8	Processus et phases de l'apprentissage supervisé.	139
7.8.a	Processus général de l'apprentissage supervisé selon Kotsiantis [2007]	139
7.8.b	Représentation IDEF0 des deux phases de l'apprentissage supervisé	139
7.9	Processus et phases de l'apprentissage supervisé.	139
7.9.a	Exemple de la classification des e-mails selon certaines caractéristiques x et y . La classe en vert regroupe les e-mails non-spam et la classe en rouge regroupe les e-mails spam	139
7.9.b	Exemple de l'estimation du prix d'une voiture selon une caractéristique x	139
7.10	Processus et résultat de l'apprentissage non-supervisé.	140
7.10.a	Objectifs du partitionnement de données (clustering)	140
7.10.b	Processus de l'apprentissage non-supervisé selon Halkidi et al. [2001]	140
7.11	Objectif et exemple de processus de la classification semi-supervisée.	141
7.11.a	Objectif de la classification semi-supervisée avec deux classes connues à priori (orange et bleue)	141
7.11.b	Processus de la pseudo-labellisation, une méthode de la classification semi-supervisée [SHUBHAM, 2017]	141
7.12	Le partitionnement semi-supervisé avec les contraintes <i>must-link</i> (en vert continu) et <i>cannot-link</i> (en rouge pointillé).	141
7.13	Processus et exemple de l'apprentissage par renforcement.	142
7.13.a	Processus de l'apprentissage par renforcement selon Sutton et al. [1998]	142
7.13.b	Exemple de l'interaction d'un agent avec son environnement [Kaelbling et al., 1996]	142
7.14	Les trois tâches de la fouille de processus.	143
7.14.a	Exemple d'un journal des événements (Event log) [Van der Aalst, 2005]	143

7.14.b	Les trois tâches de la fouille de processus selon Van Der Aalst [2012]	143
7.15	Résultats de la découverte des processus : (a) est le processus (exprimé sous la forme d'un réseau de Pétri) découvert à partir du journal des événements de la Figure 7.14.a, (b) et (c) sont respectivement la structure organisationnelle et le transfert d'activités relatifs au même journal des événements.	144
7.16	Correspondance entre les relations d'ordre, les concepts des réseaux de Pétri et les logiques AND et XOR (adapté de [Van der Aalst and de Medeiros, 2005]).	145
7.17	Contrôle de conformité entre un processus prédéfini (a) et quelques exécutions réelles capturées dans les journaux des événements (b,c,d et e) [Van der Aalst, 2005]	146
7.18	La troisième tâche de la fouille de processus : amélioration d'un modèle de processus existant (adapté de http://fluxicon.com/).	147
7.19	Exemple d'un processus à cinq patterns différents (exprimé en BPMN [OMG, 2013]).	148
7.20	Résultat de l'étude de l'application des techniques étudiées dans notre contexte (La note 0 veut dire que la technique n'est pas applicable, 5 pour dire qu'elle est difficilement applicable sans efforts d'adaptation et 10 pour dire qu'elle est tout à fait applicable).	148
7.21	Processus général du module de l'aide à la décision ($A \xrightarrow{\langle\langle\text{utilise}\rangle\rangle} x$ se lit "A utilise x").	150
7.22	Processus illustratif où les trois jonctions (And, Or et Xor) sont présentes (exprimé en BPMN [OMG, 2013]).	151
7.23	Transformation de la jonction "Or" en "And" si toutes ses conditions sont vérifiées (Extrait du processus de la Figure 7.22).	152
7.24	Méta-modèles des deux langages compatibles avec les outils de fouille de processus.	155
7.24.a	Méta-modèle du langage MXML [van Dongen and Van der Aalst, 2005]	155
7.24.b	Méta-modèle du langage XES [Gunther and Verbeek, 2014]	155
7.25	Représentation, d'une activité, adoptée par le standard XES et celle adoptée par notre travail de recherche qui ne prend en compte que les activités dont l'exécution est terminée.	156
7.25.a	A_1 vue par le standard XES	156
7.25.b	Représentation de A_1 dans notre travail de recherche	156
7.26	Conversion XMI _s 2XES : Passage d'un ensemble de traces, de format XMI, à un seul journal des événements de format XES.	157
7.27	Projections entre les deux espaces techniques MDA et XML (adapté de [Kurtev et al., 2002]).	157
7.28	Processus de l'aide à la décision globale. Les activités en jaune sont introduites pour lever les verrous identifiés.	159
7.29	Partie des patterns identifiés par la perspective ProM "Explore Event Log (Trace Variants/Searchable/Sortable)".	160
7.30	Processus de l'aide à la décision globale amélioré. Les activités en jaune sont introduites pour lever les verrous identifiés.	161
7.31	Génération du modèle réseau de Petri par les algorithmes Alpha (Alpha++), IVM et IM	163
7.32	Traduction des modèles générés, par les algorithmes Alpha (Alpha++), IVM et IM, en modèles BPMN.	164
7.33	Génération des règles de décision en utilisant les deux algorithmes DFD et MPE.	167
7.33.a	Modèle réseau de Petri enrichi avec les règles de décision, généré en utilisant l'algorithme "Data Flow Discovery".	167
7.33.b	Modèle réseau de Petri enrichi avec les règles de décision, généré en utilisant le Multi-Perspective Explorer.	167
7.34	Génération des règles de décision relatives aux variables V_1 et <i>Saison</i> en utilisant le DFD.	168
7.35	Génération des règles de décision relatives aux variables <i>RH</i> et <i>Saison</i> en utilisant le MPE.	169
7.36	Les règles de décision du modèle réseau de Petri résultant après l'agrégation des règles de décision identifiées séparément par le MPE.	169
7.34.a	Règle de décision générée pour la variable de contexte V_1 , en utilisant le DFD	169
7.34.b	Règle de décision générée pour la variable de contexte <i>Saison</i> , en utilisant le DFD	169
7.35.a	Règle de décision générée pour la variable de contexte <i>Saison</i> , en utilisant le MPE	169

7.35.b	Règle de décision générée pour la variable de contexte <i>RH</i> , en utilisant le MPE	169
7.37	Nécessité de propager les règles de décision situées au niveau des activités invisibles aux activités postérieures voisines.	170
7.38	Les règles de décision du modèle réseau de Petri résultant après la propagation des règles de décisions à travers les jonctions.	171
7.39	Processus de l'aide à la décision globale amélioré pour permettre la génération des règles de décision précises [Es-Soufi et al., 2017b]. Les activités en jaune sont introduites pour lever les verrous identifiés.	171
7.40	Transformation d'un journal de patterns (issu de l'exécution des étapes 6 et 7 de la Figure 7.39) contenant toutes les activités des patterns identifiés d'un journal des événements, à un journal par activité.	172
7.41	Représentation BPMN du processus de la classification supervisée.	173
7.42	Résultat de la prédiction de la ressource logicielle lors de l'exécution de l'activité "External functional analysis", par l'algorithme J48 avec les paramètres par défaut.	173
7.43	Nécessité d'adapter la proposition des patterns au contexte de la décision.	174
7.44	Processus suivi par le module de gestion des changements.	176
7.45	Passage d'un pattern à un autre selon la règle et le contexte de la décision.	177
7.46	Structure de la troisième partie du manuscrit (décrite en utilisant le formalisme BPMN [OMG, 2013]).	180
8.1	Activités de vérification et validation dans notre contexte. Se référer à la Figure 1.1 pour plus d'informations sur le formalisme utilisé.	184
8.2	Techniques de vérification et de validation des modèles de simulation selon Balci [1997]. Les techniques surlignées en bleu (et encadrées en noir) sont celles que nous avons utilisées pour vérifier et valider les programmes informatiques des modules de l'IDSS.	185
8.3	La validation des systèmes vue par Landry et al. [1983].	186
8.4	La validation des systèmes vue par Finlay and Wilson [1997].	187
8.5	Processus de validation des systèmes à deux niveaux [Borenstein, 1998].	188
8.6	La validation des systèmes vue par Seepersad et al. [2006].	188
8.7	Cadre de V&V que nous adoptons pour valider le IDSS proposé. Les activités de validation sont en jaune.	190
8.8	Validation conceptuelle de l'IDSS proposé.	192
8.9	La validation automatique des méta-modèles (resp. Modèles) dans l'environnement Eclipse.	192
8.9.a	Possibilité de valider automatiquement les méta-modèles (resp. Modèles) dans l'environnement Eclipse	192
8.9.b	Exemple d'une validation échouée	192
8.9.c	Exemple d'une validation réussie	192
8.10	La trace générée en instanciant le méta-modèle de trace et le fichier XML correspondant.	193
8.10.a	Trace générée en instanciant le méta-modèle de trace. Notons qu'il s'agit du pattern P_6 du Tableau 7.8	193
8.10.b	Fichier XML relatif à la trace de la Figure 8.10.a	193
8.11	Interface de l'IDSS proposé.	195
8.12	Représentation BPMN du processus rassemblant tous les patterns exécutés lors de la gestion de l'anomalie de pression dans le réseau fictif. Notons qu'aucune règle de décision n'est connue a priori.	197
8.13	Partie des règles de décision identifiées, contrôlant l'exécution des patterns de la Figure 8.12.	197
8.14	Le système IDSS dynamique : passage en temps réel d'un pattern à un autre lors de l'aide à la décision.	198
8.14.a	Décisions de t_1 à t_2	198
8.14.b	Décisions à t_3	198
8.15	Architecture modulaire du système IDSS : possibilité de remplacer les modules à condition que le module remplaçant ait les mêmes entrées/sorties que le module remplacé.	200

9.1	Incapacité d'identifier les règles relatives à l'exécution multiple des activités.	206
9.1.a	Processus exemple avec le présence de l'exécution multiple des activités (ici, c'est l'activité A_{13} qui s'exécute plusieurs fois si ($V_1 > 54$)).	206
9.1.b	Règle de décision relative à la variable de contexte V_1 identifiée à partir des patterns du processus de la Figure 9.1.a	206
2	Modèle PPO (Produit-Processus-Organisation) de Nowak et al. [2004]	243
3	Méta-modèle de processus pour la gestion des conflits [Ouertani, 2007]	244
4	Modèle de donnée pour la caractérisation de l'évolution de donnée [Couturier et al., 2014]	244
5	Méta-modèle pour l'identification des changements [Moones et al., 2015]	245
6	Modèle FBS-PPRE de Labrousse and Bernard [2008]	245
7	Méta-modèle de processus orienté qualité [Heidari et al., 2011]	246
8	Méta-modèle transactionnel pour la modélisation des processus basée sur la structure organisationnelle [Thom et al., 2005]	247
9	Méta-modèle BPPA pour modéliser les processus en se basant sur les pratiques quotidiennes [Martins and Zacarias, 2015]	248
9.a	Les trois couches du méta-modèle BPPA [Martins and Zacarias, 2015]	248
9.b	Couche "Action" du méta-modèle BPPA [Martins and Zacarias, 2015]	248
10	Méta-modèle BPMN [OMG, 2013]	249
11	Formalisme de modélisation IDEF0 [Noran, 2007]	250
12	Modèle de données produit basé sur le domaine [Li et al., 2011]	250
13	Framework de modélisation des données produit de Sudarsan et al. [2005]	251
13.a	Premier composant : Modèle produit de base	251
13.b	Deuxième composant : Modèle d'assemblage ouvert	251
13.c	Troisième composant : Modèle d'intégration conception-analyse	251
13.d	Quatrième composant : Modèle d'évolution de la famille du produit	251
14	Modèle de produit pour le système UPLS de Lee and Suh [2009]	252
15	Méta-modèle décrivant les concepts de base pour produire un modèle de produit orienté objet [Wimmer et al., 2003]	253
16	Méta-modèle de trace pour la maintenance [Karray, 2012]	253
17	Méta-modèle de trace orienté caractéristique [Shen et al., 2009]	254
18	Méta-modèle ATF de traçabilité [Anquetil et al., 2010]	254
19	Méta-modèle de traçabilité dédié au développement des entrepôts de données [Maté and Trujillo, 2012]	255
20	Méta-modèle de traçabilité des compositions de modèles [Laghouaouta et al., 2017]	255
21	Méta-modèle de trace MAP [Hug et al., 2012]	255

Liste des tableaux

3.1	Situations de décision durant le cycle de vie d'un produit (adapté de l'INCOSE [2015]; Buede [2011])	29
3.3	Dimensions et caractéristiques d'un problème de décision (adapté de [Clark et al., 2009])	32
3.4	Catégorie d'approches de prise de décision selon Wang and Ruhe [2007]	36
3.5	Les classes des DSSs (adapté de Alter [1980])	51
3.6	Comparaison des systèmes intelligents par rapport aux caractéristiques recherchées. "✓" (resp. "-") veut dire que le système répond parfaitement (resp. ne répond pas) à l'exigence, "∼" veut dire qu'il répond en partie à l'exigence et la case vide veut dire qu'il n'a pas été testé.	55
3.7	Comparaison des outils commercialisés présentés par rapport aux caractéristiques recherchées. "✓" (resp. "-") veut dire que l'outil répond parfaitement (resp. ne répond pas) à l'exigence, "∼" veut dire qu'il répond en partie à l'exigence et la case vide veut dire qu'il n'a pas été testé.	60
4.1	Quelques méthodes et outils mettant en pratique le processus ADM	71
5.1	Les quatre principales méthodologies de l'architecture des systèmes (adapté de Rechtin and Maier [2010])	90
6.1	Types des différentes jonctions successeurs	118
7.1	Les patterns résultants des différentes exécutions possibles du processus de la Figure 7.3	130
7.2	Exemple des achats d'un client.	135
7.3	Indices de support et de confiance pour les différentes règles d'association de la base de données du Tableau 7.2.	136
7.4	Exemple de base de données séquentielles contenant cinq séquences, adapté de [Agrawal and Srikant, 1995].	136
7.5	La forme d'un jeu de données où deux paramètres sont connus : X étant le vecteur de caractéristiques et Y étant la classe à laquelle appartient X	137
7.6	La forme d'un jeu de données où seul X est connu.	137
7.7	La forme d'un jeu de données où X et quelques Y sont connus.	137
7.8	50 patterns, parmi les 94 possibles, de l'exemple d'illustration de la Figure 7.22	153
7.9	Logique de construction des traces du processus exemple de la Figure 7.22	157
7.10	Résultat de comparaison des algorithmes de fouille de pattern étudié. ✓ (resp. -) veut dire "oui" (resp. "non") et ∼ veut dire "pas vraiment".	160
7.11	Résultat de la comparaison des algorithmes permettant la génération automatique du modèle réseau de Petri à partir de notre journal des événements	162
7.12	Résultat de la comparaison des deux algorithmes de fouille de décision que nous avons gardés.	166
8.1	Données contextuelles aux instants t_1 , t_2 et t_3 lors de la supervision du réseau de gaz virtuel.	197
8.2	Étude des caractéristiques du système IDSS proposé.	199
8.3	Récapitulatif du résultat de la V&V du système IDSS proposé.	200

Introduction générale

”Anybody who has been seriously engaged in scientific work of any kind realizes that over the entrance to the gates of the temple of science are written the words : ”Ye must have faith”. It is a quality which the scientist cannot dispense with.”

Max Planck

1.1 Motivations : pourquoi a-t-on besoin d’être assisté lors de la conception ou la supervision d’un système ?	2
1.2 Objectifs	2
1.3 Hypothèse et approche	2
1.4 Contributions	3
1.5 Structure du manuscrit	4

Tous les progrès scientifiques, technologiques et humains, entre autres, sont les résultats de nombreux travaux de recherche ambitieux et rigoureux, qui commencent par définir un problème donné puis l’analyser pour identifier une ou plusieurs solutions possibles. Cette analyse est souvent présentée en détail dans un document, d’une centaine de pages, qui sert de référence ¹ pour les nouvelles recherches visant à raffiner ou benchmarker les solutions proposées. Il est, par conséquent, nécessaire que le document soit clair (en termes de définition du problème, des objectifs, des hypothèses, etc.), bien organisé, agréable et facile à lire ; pour permettre au lecteur de plonger dans un monde apportant un vrai moment de plaisir.

Cette introduction générale représente l’aperçu, le plus concis possible, du présent manuscrit de thèse, dont chaque section est présentée plus en détails dans les chapitres suivants en justifiant le pourquoi de la chose. Nous commençons d’abord par brièvement lister les raisons qui justifient pourquoi le présent sujet de recherche nécessite d’être abordé. Ensuite, nous listons brièvement les objectifs que nous souhaitons atteindre ainsi que l’hypothèse et l’approche adoptées pour les atteindre. Nos contributions sont ensuite listées avant de présenter la structure du présent manuscrit de thèse.

1. Il est vrai que dans le cas présent, les articles scientifiques sont très majoritairement lus et synthétisés pour comparer son travail aux ceux des autres équipes de recherche, mais un manuscrit de thèse reste toujours la source la plus précieuse pour comprendre plus en détail le contexte, les objectifs et les contributions, entre autres, des autres travaux.

1.1 Motivations : pourquoi a-t-on besoin d'être assisté lors de la conception ou la supervision d'un système ?

Il est clair que l'industrie actuelle est en développement permanent suite à l'apparition du concept de l'industrie du futur (Industrie 4.0) visant à faire de l'usine une entité de plus en plus intelligente (smart factory) en combinant les processus et pratiques industriels avec les technologies de l'information et de communication récentes, comme les systèmes cyber-physique, l'internet des objets (IoT) et le cloud computing, entre autres. Ce développement industriel, ainsi que le besoin d'innovation pour atteindre la compétitivité favorisent une augmentation phénoménale du volume de données (connue sous le nom de Big Data), rendant ainsi les processus de conception et de supervision² de plus en plus complexes, les données au sein des organisations de plus en plus difficiles à exploiter et les ingénieurs de plus en plus perdus lors de la prise de décision. En effet, plusieurs problèmes sont identifiés dans les industries actuelles (discutés plus en détails dans la Section 2.3.1) :

- Difficultés lors de la recherche et l'échange de l'information, notamment la :
 - Perte de temps suite à la recherche de l'information,
 - Perte de l'information elle même,
 - Perte de temps suite à l'échange des données entre les collaborateurs.
- Difficultés au niveau de la prise de décision et la réutilisation des décisions passées, notamment la :
 - Qualité médiocre des décisions prises,
 - Perte de temps lors de la prise de décision,
 - Perte de temps sur des activités à non valeur ajoutée,
 - Non maîtrise du temps de développement.
- Difficultés et perte de temps lors de l'identification et la propagation des différents changements.

Il est par conséquent avantageux - et nécessaire - d'étudier le problème de prise de la décision et l'exploitation des données dans le contexte technologique et économique actuel. Ceci fait bien l'objet du présent travail de recherche.

1.2 Objectifs

L'objectif global, de ce travail, est de permettre aux entreprises de gagner le temps en assistant leurs ingénieurs dans leur processus de prise de décision. Cet objectif est subdivisé en sous-objectifs :

- Développer une méthode qui permettra de modéliser et visualiser les processus,
- Développer une méthode qui permettra de tracer les informations des processus pour faciliter leur quête,
- Développer une méthode qui permettra d'analyser et d'évaluer les décisions passées pour pouvoir proposer la décision la plus appropriée, au bon moment et à la bonne personne,
- Développer une méthode qui permettra d'identifier les changements de contexte et de les propager au bon moment et au bon endroit pendant la prise de décision,
- implémenter les méthodes développées sous la forme de modules, de façon à avoir un système d'aide à la décision intelligent et modulaire.

1.3 Hypothèse et approche

Comme dans tout travail de recherche scientifique, l'analyse de la problématique doit se faire d'une façon rationnelle et systématique pour expliquer le problème et lui apporter des éléments de réponse. Dans ce

2. Dans ce travail de recherche, nous nous intéressons plus particulièrement aux processus de conception et de supervision des systèmes, étant les plus critiques du processus de développement produit (voir la Section 2.2.3 pour plus de détails).

présent travail de recherche, nous adoptons une démarche déductive, dans laquelle une hypothèse, *”is put to the test by confronting it with observations that either lead to a confirmation or a rejection of the hypothesis”* [Snieder and Larner, 2009]. Nous faisons, donc, l’hypothèse que le couplage des techniques de modélisation des processus, de traçabilité, de fouille de données et de fouille de processus, permet de répondre à nos objectifs. De plus, nous abordons le problème de la prise de décision comme décrit par la Figure 1.1 :

- Les processus, de conception et de supervision des systèmes, sont d’abord tracés (la modélisation est, quant à elle, optionnelle, car elle dépend de la demande de l’utilisateur). Le lien de dépendance entre les activités de traçabilité et de modélisation sera étudié dans le chapitre 5,
- Ces traces servent de base pour le module d’aide à la décision qui les analyse pour proposer la décision la plus appropriée en tenant compte du contexte de décision à l’instant t ,
- Le contexte de la décision est sujet à des changements dynamiques et parfois non planifiés et imprévisibles. Il est donc nécessaire d’introduire un module de gestion des changements contextuels permettant de continuellement mettre à jour la décision proposée par le module d’aide à la décision.

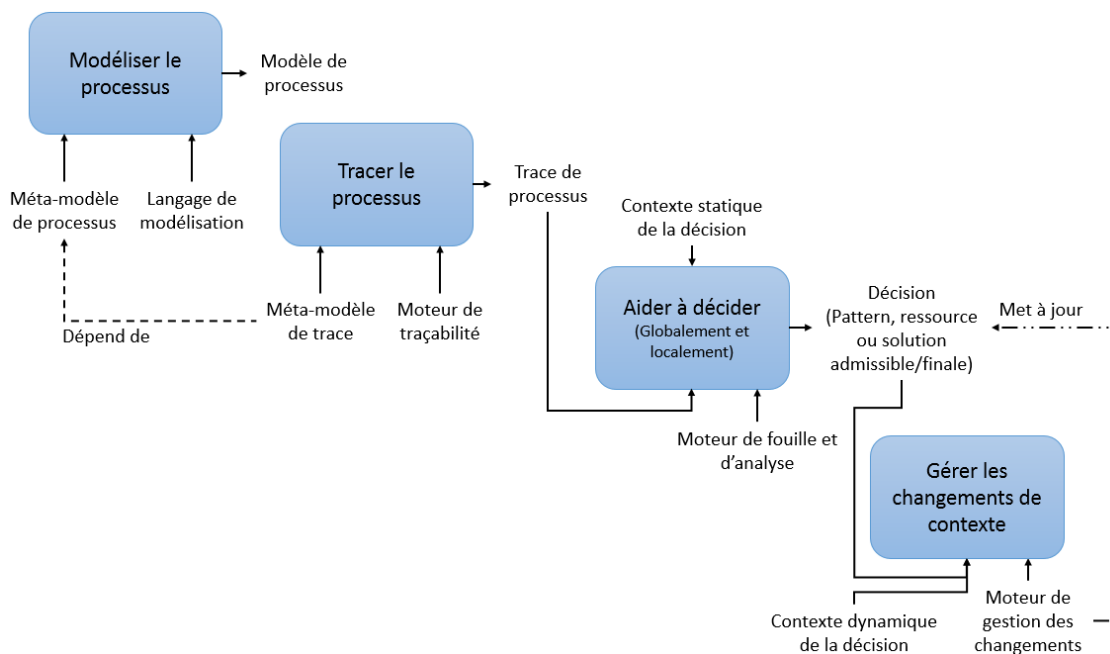


FIGURE 1.1 – Représentation IDEF0 de l’approche globale du présent travail de recherche (les deux flèches en pointillé ne font pas partie du formalisme IDEF0, elles ont été introduites pour montrer la l’interaction indirecte entre les activités)

1.4 Contributions

La contribution principale du présent travail de recherche est la **conception (opérationnelle, logique et physique) et l’implémentation d’un système logiciel, générique, intelligent et modulaire d’aide à la décision** ; permettant de supporter les ingénieurs dans leurs processus décisionnels lors de la conception et/ou la supervision de n’importe quel système. De plus, quelques éléments techniques de notre contribution sont également génériques et peuvent être réutilisés pour implémenter d’autres systèmes ou développer d’autres applications :

- **Module de modélisation et traçabilité des processus** : le méta-modèle de processus proposé, soulignant les éléments à modéliser ainsi que les relations entre eux, peut être réutilisé pour modéliser les processus de conception et de supervision d’un système quelconque. De ce méta-modèle de processus générique peuvent être dérivés plusieurs méta-modèles de traces mentionnant les éléments à tracer lors de l’exécution des processus de conception ou de supervision des systèmes.

- **Module d'aide à la décision :**
 - Une transformation ATL permettant de convertir les traces sous forme de fichiers XMI (XML Metadata Interchange) en un seul fichier exprimé sous le format XES (eXtensible Event Stream), étant le format d'entrée des outils de fouille de processus.
 - Une application JAVA permettant de générer automatiquement le modèle réseau de pétri à partir d'un journal des événements exprimé sous le format XES.
 - Une application JAVA permettant d'agrèger les fichiers PNML (Petri Net Markup Language) dont chacun contient la règle de décision relative à une seule variable de décision, en un fichier PNML final contenant les règles de décisions générales (relatives à toutes les variables de décision).
 - Une application JAVA permettant de propager les règles de décisions à travers les points de décisions.
 - Une application JAVA permettant de fouiller dans des documents CSV contenant les anciennes traces d'un processus, et de les post-traiter pour produire un document qui rassemble tous les patterns existants, non dupliqués, en mentionnant la fréquence d'apparition de chaque pattern.
- **Module de gestion des changements contextuels :** Une application de filtrage statique et dynamique qui, en ayant comme entrées un fichier PNML contenant les règles de décisions et un fichier CSV contenant les patterns ordonnés, permet de continuellement mettre à jour le pattern et les activités à proposer selon le contexte de décision à l'instant t .
- Un **cadre de vérification et de validation** des systèmes d'aide à la décision intelligents (IDSS).

1.5 Structure du manuscrit

La suite du manuscrit se divise en huit chapitres dont la structure est présentée dans la Figure 1.2 et dont le contenu est brièvement synthétisé ci-dessous.

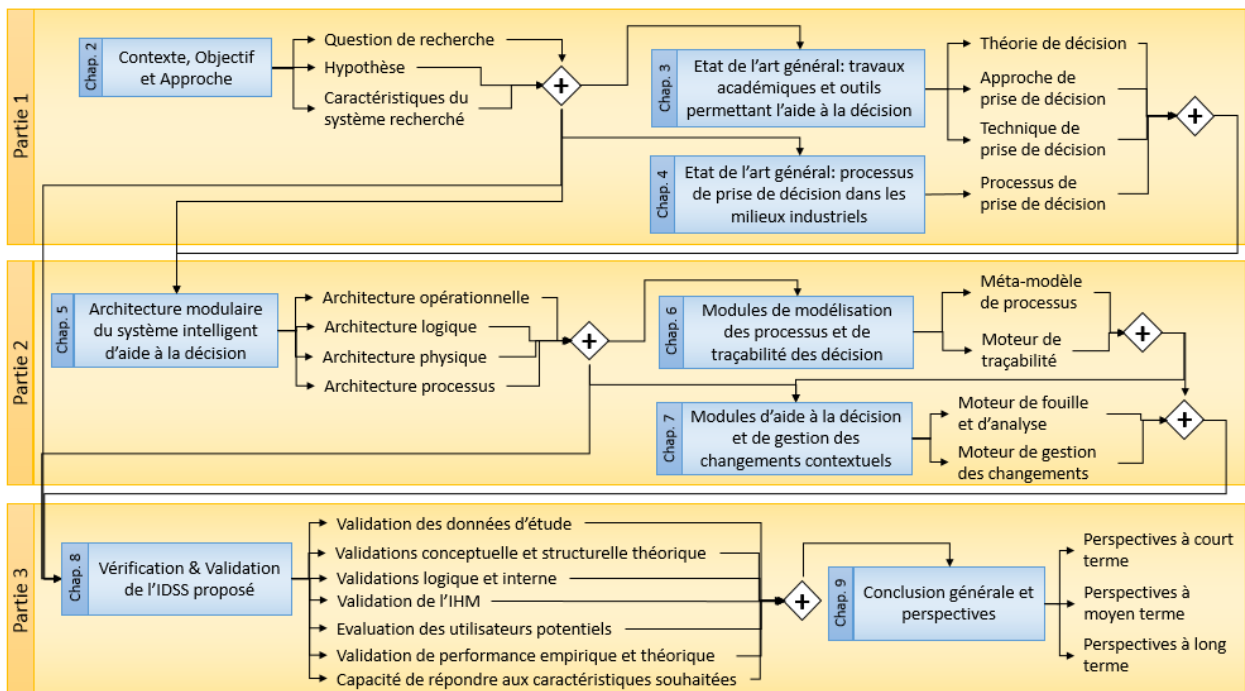


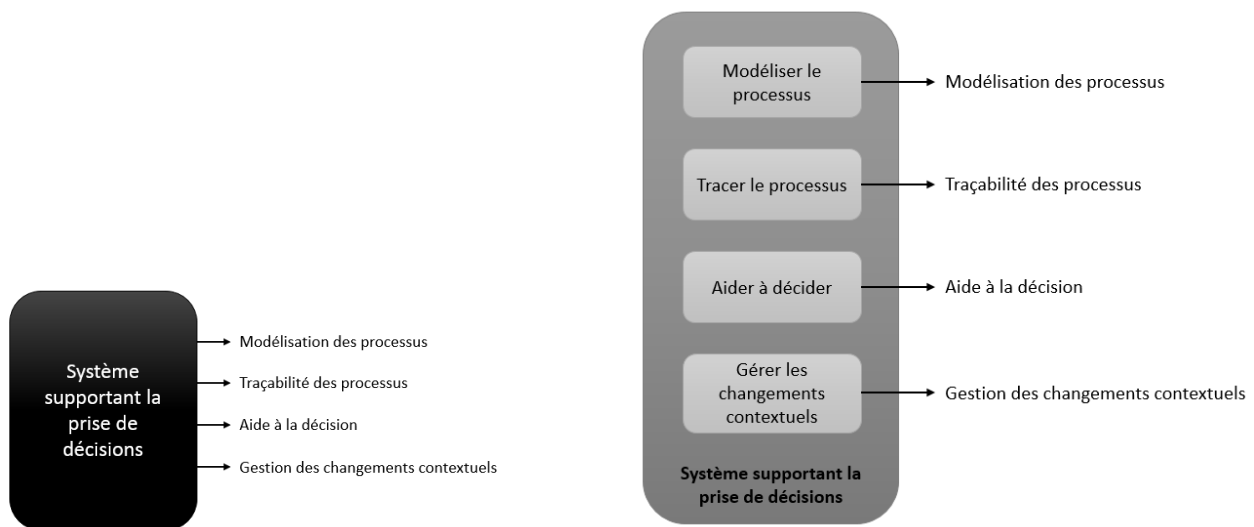
FIGURE 1.2 – Représentation BPMN de la structure du présent manuscrit de thèse (décrite en utilisant le formalisme BPMN [OMG, 2013]).

Le Chapitre 2 a pour objectif de justifier l'intérêt de notre contexte de recherche en adoptant la logique Top-down (an allant du besoin d'être compétitif sur le marché, au besoin de maîtriser les processus de

conception et de supervision des systèmes), d'introduire la problématique industrielle et d'identifier les objectifs scientifiques afin de formuler la question de recherche. Dans le même chapitre, l'hypothèse de travail -étant la première piste vers l'apport des éléments de réponse à la question de recherche- est ensuite présentée avant de détailler les caractéristiques du système d'aide à la décision recherché. Ces caractéristiques nous serviront de critères pour comparer les systèmes que nous étudions dans le chapitre suivant.

Avant de proposer une quelconque solution, il est nécessaire de faire un tour d'horizon sur les travaux et solutions existants dans la littérature. L'objectif de ce travail étant de proposer un système d'aide à la décision permettant de réaliser les sous-objectifs (fonctions) que nous avons identifiés (modélisation des processus, traçabilité, aide à la décision et gestion des changements contextuels), nous souhaitons -à travers les Chapitres 3 et 4- investiguer l'existence d'un tel système en explorant les milieux académiques, commerciaux et industriels.

Il est important de noter que, dans le présent travail de recherche, nous considérons **deux niveaux de l'état de l'art**. Un **état de l'art général** dans lequel le système que nous cherchons est vu comme une **boîte noire** dont les fonctions exécutées sont relatives à nos quatre objectifs, comme le montre la Figure 1.3.a. Si toutefois aucun des systèmes existants ne répond à la totalité de nos objectifs, nous verrons le système recherché comme une boîte grise composée de composants dont chacun réalise un de nos objectifs, comme le montre la Figure 1.3.b; et nous faisons, ainsi, un **état de l'art par fonction** pour découvrir les travaux de recherche relatifs aux quatre composants.



1.3.a. Etat de l'art général : le système est vu comme une boîte noire exécutant des fonctions.

1.3.b. Etat de l'art par chapitre : le système est vu comme une boîte grise dont chacun des composants réalise une fonction.

FIGURE 1.3 – Les deux niveaux de l'état de l'art que nous considérons dans le présent travail de recherche..

Le chapitre 3 commence par introduire ce qu'est la décision, le problème de décision, les approches, les méthodes, les techniques et les processus de décision existants dans la littérature. Ensuite, quelques systèmes d'aide à la décision proposés dans les milieux académiques, et ceux commercialisés, sont présentés, synthétisés et discutés à la fin du chapitre. Le Chapitre 4 est un état de l'art industriel visant à explorer les solutions adoptées par les industries et par quelques organisations et associations. Les deux chapitres sont clôturés par une synthèse et comparaison des solutions proposées par rapport à nos critères de recherche, établis dans le Chapitre 2. Le verdict (ré-utiliser un système existant ou pas?) est ensuite prononcé à la fin du Chapitre 4.

Les chapitres 5, 6 et 7, de la partie proposition, présentent les modules de notre système proposé. Ces chapitres adoptent la même structure : introduction, définition de la fonction selon notre contexte de

recherche, état de l'art, synthèse et proposition.

Le Chapitre 5 a pour but de concevoir l'architecture de notre système intelligent d'aide à la décision. Cette architecture a pour but de définir notre système selon quatre points de vue. Le point de vue opérationnel permet de représenter le système comme une boîte noire ayant des entrées, des sorties et devant satisfaire les contraintes imposées par son environnement. Le point de vue logique représente le système sous la forme d'une boîte blanche, cette représentation permet d'identifier les fonctions exigées par l'utilisateur final. Le point de vue physique décrit les solutions de conception permettant d'implémenter les fonctions déjà identifiées et le point de vue processus a pour objectif de décrire l'enchaînement réel des activités du système à concevoir.

Le Chapitre 6 introduit le module de modélisation des processus et celui de la traçabilité. Concernant le premier module, nous adoptant l'ingénierie dirigée par les modèles comme façon de penser, et nous proposons, ainsi, un méta-modèle de processus permettant d'identifier les concepts à tracer. Nous laissons le choix du langage de modélisation à l'utilisateur final vu que nous avons la possibilité de passer d'un modèle de processus établi en utilisant un langage L_1 à un autre établi en utilisant un langage L_2 via des transformations de modèles. Concernant le module de traçabilité étudié dans le même chapitre, et toujours en adoptant l'ingénierie dirigée par les modèles, nous proposons une traçabilité intégrale en gardant le méta-modèle de processus comme méta-modèle de trace.

Le Chapitre 7 introduit le module d'aide à la décision permettant d'exécuter deux sous-fonctions. La première est l'aide à la décision globale, il s'agit ici de proposer le pattern de processus le plus adéquat pour un contexte donné en explorant les anciennes traces d'exécution, notamment l'identification des points et règles de décision, et en les filtrant par rapport au contexte de la décision. La deuxième fonction est l'aide à la décision locale, il s'agit ici de proposer les paramètres (ressources et/ou décisions) les plus significatifs pour chaque activité du pattern choisi en tenant compte du contexte de la décision. Le même chapitre introduit le module de gestion des changements contextuels qui, via la technique de filtrage dynamique proposée, permet de continuellement mettre à jour le module d'aide à la décision en tenant compte de l'aspect dynamique du contexte de la décision.

Le Chapitre 8, de la troisième partie, est une vérification et validation de la proposition. Il s'agit ici de valider le système selon plusieurs points de vue et d'illustrer son fonctionnement en l'appliquant sur un cas d'étude réel de supervision, en temps réel, des réseaux de gaz favorisant l'injection du gaz vert. Le système a pour but, ici, de proposer les actions correctives à mener pour résoudre un problème quelconque lors de la supervision. Dans le même chapitre, nous validons le système intelligent d'aide à la décision proposé en étudiant s'il répond bien aux critères de recherche établis dans le Chapitre 2.

Le Chapitre 9 est une conclusion générale résumant et synthétisant ce qui a été fait dans le présent travail de recherche. À la fin du chapitre, les perspectives de recherche à court, moyen et long termes - visant à améliorer les performances du système proposé - sont présentées.

Première partie

**PROBLÉMATIQUE ET ÉTAT DE L'ART
GÉNÉRAL**

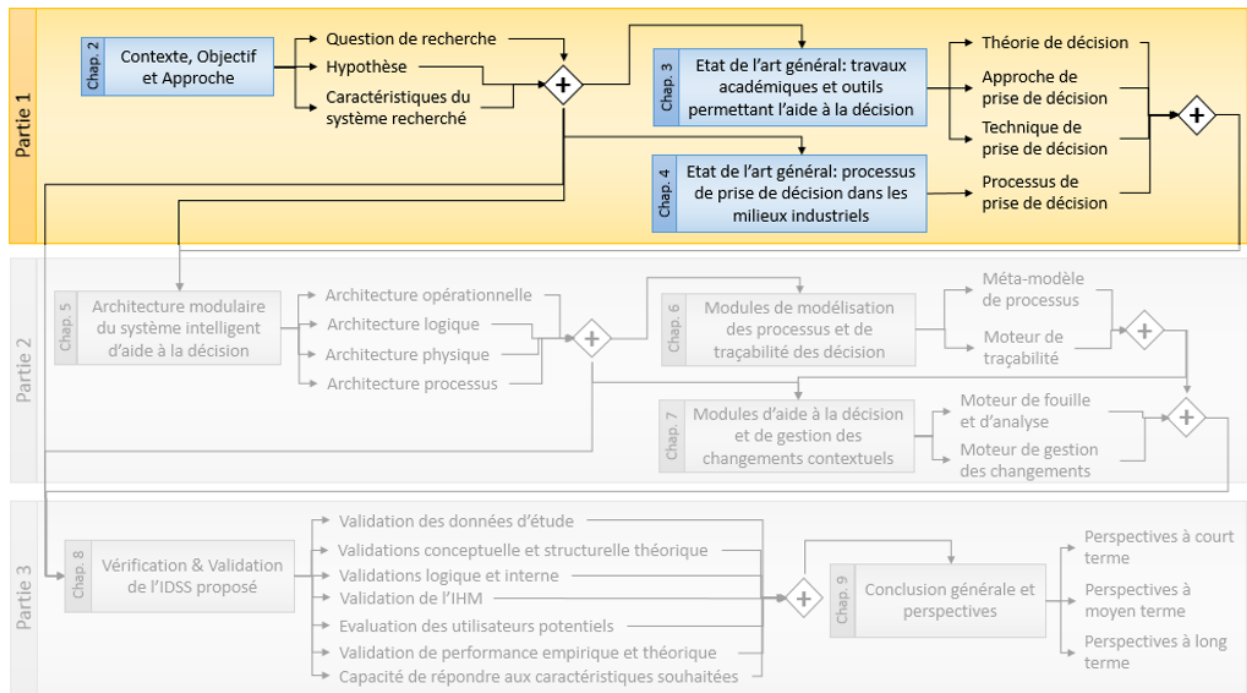


FIGURE 1.4 – Structure de la première partie du manuscrit (décrite en utilisant le formalisme BPMN [OMG, 2013]).

Cette première partie se compose de trois chapitres (2, 3 et 4). Dans le chapitre 2, nous introduisons le contexte industriel qui nous amène à poser la question de recherche, étant le verrou scientifique que nous cherchons à lever en partant d'une hypothèse, que nous discutons dans le même chapitre avant de lister les caractéristiques desquelles le système que nous cherchons doit se doter. Les deux chapitres suivants (3 et 4) concernent un état de l'art général portant sur la prise de décision. Dans le chapitre 3, nous présentons les travaux de recherche académiques et les outils commercialisés, qui permettent de supporter la prise de décision. Ensuite, nous présentons, dans le chapitre 4, les différents processus industriels de prise de décision. Ces chapitres sont conclus par une synthèse où les travaux étudiés sont discutés et comparés par rapport à nos critères de recherche.

Contexte, Objectifs et Approche

"Science is fun. Science is curiosity. We all have natural curiosity. Science is a process of investigating. It's posing questions and coming up with a method. It's delving in."

Sally Ride

2.1 Contexte industriel	10
2.1.1 Le besoin d'être compétitif sur le marché	10
2.1.2 Comment être compétitif sur le marché	10
2.1.3 Discussion	14
2.2 Processus de développement produit (PDP)	15
2.2.1 Étapes du PDP	15
2.2.2 Maîtriser le PDP	16
2.2.3 Discussion : contexte de recherche	18
2.3 Problématique, question de recherche, hypothèse et caractéristiques du système recherché	20
2.3.1 Problématique : les besoins industriels	20
2.3.2 Objectifs scientifiques et question de recherche	23
2.3.3 Hypothèse de travail	24
2.3.4 Caractéristiques générales du système recherché	25

Combien de fois vous avez acheté un produit et que, quelques mois plus tard, vous vous rendez compte qu'une nouvelle version de ce même produit a déjà été introduite sur le marché? Votre réponse serait potentiellement "plusieurs fois" avec un soupir déçu. Il est en effet difficile de suivre l'évolution continue des produits, dès que vous commencez à croire que vous possédez la dernière version d'un téléphone portable par exemple, une nouvelle version améliorée et plus performante voit le jour. Cette introduction constante de nouveaux produits sur le marché, connue sous le nom de "Développement Produit" (DP), fait partie des stratégies gagnantes que les entreprises adoptent pour survivre dans un marché de plus en plus concurrentiel. L'objectif ultime de ce travail de recherche est de proposer une démarche qui permet aux entreprises de gagner le temps, qu'elles perdent inutilement, pour l'exploiter à des fins d'innovation afin de monter en compétitivité. Dans ce chapitre, nous essayons de répondre aux questions suivantes : Qu'est ce que la compétitivité? Comment est ce que les entreprises deviennent compétitives? Pourquoi est ce que le processus de DP est-t'il important vis-à-vis de la compétitivité? Pourquoi est ce que les processus de conception et supervision sont les plus critiques parmi tous les autres processus du DP? Quels sont les objectifs et question de recherche? Quelle est l'hypothèse de travail?

2.1 Contexte industriel

2.1.1 Le besoin d'être compétitif sur le marché

Il n'existe pas de définition unique de la compétitivité, mais la plus simple serait la capacité d'une entreprise à affronter la concurrence des autres entreprises sans subir de désavantages économiques significatifs [Ketels, 2016]. La compétitivité est au cœur de la réussite ou l'échec d'une entreprise, elle est par conséquent considérée comme un domaine d'étude qui préoccupe fortement tout type d'industrie et plusieurs milieux universitaires. En effet, de nombreuses recherches ont mis en valeur l'équilibre concurrentiel entre les entreprises implantées dans différentes zones géographiques au niveau national ou international. Un exemple classique est la recherche menée à l'Institut de technologie du Massachusetts (MIT)³ par Dertouzos et al. [1989] qui se sont intéressés à la comparaison de la compétitivité des industries japonaises et européennes avec l'industrie américaine qui a connu des problèmes majeurs suite à la deuxième guerre mondiale. Un autre exemple est la recherche menée par Porter [1990] dans laquelle la notion de la compétitivité au niveau national et international a été abordée.

L'entreprise actuelle, indépendamment de sa taille, a besoin d'être compétitive pour :

- Augmenter sa base de clients. En effet, à travers ses stratégies promotionnelles, une entreprise cherche à attirer de nouveaux clients en proposant des produits/services ayant des prix séduisants et qui se dotent d'une qualité irréprochable.
- Augmenter le nombre de ses ventes. En plus du point précédent, une entreprise cherche à encourager les clients existants pour acheter plus, en réduisant les prix de quelques produits/services par exemple.
- Maintenir ou accroître ses parts de marché. En effet, une entreprise cherche non seulement à dominer et contrôler sa part de marché mais aussi à conquérir de nouvelles parts pour faire connaître ses produits/services à grande échelle.
- Atteindre la supériorité. Une entreprise cherche à proposer un produit/service qui domine le marché en étant clairement meilleur que ceux proposés par ses concurrents.
- Améliorer son image.
- Maximiser le profit, etc.

2.1.2 Comment être compétitif sur le marché

La compétitivité est un jeu dont les règles ne sont pas faciles à dégager car elles sont définies par l'environnement compétitif qui ne cesse de changer ainsi que le contexte de chaque entreprise qui peut dépendre de son activité, sa taille, sa politique industrielle, sa structure organisationnelle, ses capacités d'investissement, etc. Plusieurs travaux de recherche se sont intéressés à l'identification de ces règles, connues sous le nom de *facteurs de compétitivité*. Lau [2002] identifie et classe ces facteurs et montre que ce classement est critique pour l'établissement de la stratégie de production en priorisant les critères à partir desquels la performance de production serait évaluée. Li [2000] a étudié les sources de la performance des fabricants chinois et a établi un classement des facteurs de compétitivité par rapport à quatre domaines fonctionnels : conception et développement de produit, marketing, production et développement des ressources humaines. Nous introduisons, ci-dessous, les facteurs que nous supposons être les plus intéressants.

Connaître son concurrent : Pour dépasser son concurrent il faut commencer par le connaître et identifier ce qu'il propose, ceci a pour but d'identifier les secteurs où l'entreprise peut rivaliser. Porter [2008], professeur à l'école de commerce de l'université Harvard, a développé un modèle de rivalité (modèle des cinq forces) qui permet de déterminer l'existence, ou pas, d'une concurrence entre les entreprises du marché ainsi que l'entreprise dominante. González-Moreno and Sáez-Martínez [2008] ont montré que le concept des groupes stratégiques, dont chaque groupe est composé des entreprises adoptant une stratégie identique à de nombreux points de vue, aide les gestionnaires à identifier leurs principaux concurrents.

3. <http://web.mit.edu/>

Définir ses objectifs et sa culture : Une entreprise peut être compétitive en développant sa propre raison d'être, ce concept a été défini par [Montgomery \[2012\]](#) comme étant tout ce qui définit une entreprise d'un point de vue fondamental. Notamment, quelle est la raison de son existence? Qu'est ce qui la distingue des autres entreprises concurrentes? Qu'est ce qu'elle apporte de plus à ses clients? Etc. [Collins \[2011\]](#) montre que les entreprises qui ont une certaine culture, et donc ayant des valeurs et objectifs bien définis, sont capables de surpasser leurs concurrents. Chris Cancialosi⁴, partenaire fondateur de la société de conseil en gestion "gothamCulture" a montré, à travers sa contribution "How Exceptional Companies Create High Performance Cultures"⁵ publiée par [Forbes](#)⁶, que l'atteinte d'une performance durable dépend de plusieurs variables dont la plus commune à toutes les entreprises (et la plus intéressante) est l'établissement d'une culture. Cette dernière, représente la façon comment les choses sont faites, est développée par l'entreprise à travers ses essais et erreurs avec le temps et c'est elle qui détermine son comportement et sa performance, toujours selon Chris Cancialosi⁴.

Gérer ses ressources et processus : Plusieurs gestionnaires d'entreprises croient que la gestion des processus, et notamment le processus de production, présente le plus fort potentiel pour améliorer la compétitivité d'une entreprise [[Rolstadås, 1995](#)]. [Chase et al. \[2004\]](#) affirment que la **conception** et le **contrôle** des systèmes organisationnels, responsables de l'utilisation productive des ressources matérielles, humaines ou toutes autres ressources productives, dessinent une orientation qui gagne et maintient l'avantage compétitif (ensemble des conditions qui permettent à une entreprise de surpasser ses concurrents). Plusieurs autres travaux montrent que la gestion des ressources humaines est étroitement associée à la performance organisationnelle d'une entreprise ([[Khatri, 2000](#)], [[Appelbaum, 2000](#)], [[Vui-Yee, 2015](#)], [[Takeuchi et al., 2003](#)]).

Être flexible, efficace et efficiente : Héraclite d'Éphèse⁷, le philosophe grec, l'a bien annoncé : "On ne se baigne jamais deux fois dans le même fleuve...". Tout est en perpétuel changement, même notre vie intérieure. Une entreprise est, elle aussi, sujette à des changements constants qui peuvent être de nature technologique (apparition de nouvelles technologies), organisationnelle (transformation des structures et des compétences), comportementale (changement des préférences ou des besoins des clients), etc. L'entreprise, qui vise d'être compétitive, doit être flexible. [Daoudi and Nurcan \[2007\]](#) définissent la flexibilité comme étant la capacité de réagir rapidement face aux changements internes et externes de l'entreprise, c'est aussi la capacité de modifier les modèles du processus métier après chaque changement de stratégie ou de fonctionnement de l'entreprise. Être flexible c'est aussi être réactif, pro-actif et prédictif. En effet, une entreprise est dite pro-active si elle anticipe les changements ou problèmes avant leur occurrence en mettant en place des stratégies de détection et propagation des changements ou de résolution de problèmes. Vu que ce n'est pas possible d'anticiper tous les changements/problèmes possibles, une entreprise ne peut pas être pro-active dans toutes les situations, elle doit donc être réactive en essayant de résoudre le problème ou de propager les changements dès qu'ils surviennent. En plus de la pro-activité et la réactivité, une entreprise doit adopter une stratégie prédictive. Les notions de prédiction et pro-activité sont considérées interchangeable par plusieurs entreprises, ce n'est pas inexact mais pas non plus exact. Le dictionnaire Oxford⁸ définit la pro-activité comme suit : "the action of creating or controlling a situation rather than just responding to it after it has happened.", le même dictionnaire définit le verbe prédire comme suit : "to say or estimate that (a specified thing) will happen in the future or will be a consequence of something.". La pro-activité est, par conséquent, une notion qualitative (qualifier si une situation a tendance d'être problématique et agir avant qu'elle le soit), tandis que la prédiction est une notion quantitative plutôt associée à la performance (selon les expériences ou l'historique, choisir le processus le plus performant dans un contexte donné). En plus de la flexibilité, une entreprise doit être efficace et efficiente. Bien que ces deux notions (en

4. <https://www.linkedin.com/in/chrisancialosi/>

5. <https://www.forbes.com/sites/chrisancialosi/2015/06/15/how-exceptional-companies-create-high-performance-cultures/#5acad2445fd1>

6. Magazine économique américain qui s'intéresse au business, investissement, technologie, entrepreneuriat, leadership et au style de vie.

7. <https://fr.wikipedia.org/wiki/H%C3%A9raclite>

8. <https://en.oxforddictionaries.com/>

Anglais effectiveness et efficiency respectivement) sont aussi interchangeables par plusieurs entreprises, la différence entre eux dépasse le niveau sémantique. En effet, une entreprise est dite efficace si elle [Mouzas, 2006] :

- Possède des objectifs bien définis et cohérents avec sa finalité pour être crédible auprès des clients. Un client n’osera pas acheter une voiture CHANEL, par exemple, car il sait que cette marque de luxe est plutôt dédiée à la mode, joaillerie, parfum et cosmétiques.
- Analyse les consommateurs actifs ou potentiels et reste informée de leurs attentes pour proposer les produits qui arrivent à les satisfaire,
- Est flexible,
- Utilise les dernières technologies pour optimiser ses processus.

En revanche, une entreprise est dite efficiente si [Mouzas, 2006] :

- Les moyens utilisés pour produire un produit/service ont été exploités avec un but d’économie. En d’autres mots, si le produit/service sont au moindre coût,
- Elle planifie et priorise les activités qui créent de la valeur ajoutée et qui ont une influence directe sur la productivité,
- Les activités qui n’ont pas de valeur ajoutée sont sous-traitées pour exploiter ce temps à d’autres fins,
- Les processus sont structurés et redéfinis en fonction des changements.

Nous remarquons que l’efficacité est directement associée au client tandis que l’efficience concerne l’entreprise parce qu’elle impacte les coûts de production. Une entreprise peut être efficace sans être efficiente, et vice versa, comme elle peut à la fois être efficace et efficiente pendant une durée limitée ou pendant toute sa durée de vie. Pour être compétitive, la meilleure approche consiste à trouver le bon dosage entre efficacité/efficience et instaurer ces deux notions comme une culture d’entreprise.

Développer, dans les meilleurs délais, des produits/services de meilleure qualité avec un prix compétitif : Dans les années 40, les entreprises exploitaient les mêmes produits pendant des dizaines d’années sans se soucier de leur qualité et leur performance. Au début des années 70, les entreprises japonaises (notamment Toyota dans l’industrie automobile et Sony dans l’industrie électronique) ont émergé et le marché est devenu de plus en plus compétitif⁹. Pour faire face à la concurrence, les entreprises ont commencé à investiguer les exigences (ou besoins) accrues et changeantes des clients et à mettre en place les processus qui transforment, dans les meilleurs délais, ces exigences en produits. A l’issue de cette mobilisation ont émergé les notions de ”*Processus de Développement Produit (PDP)*” et les outils d’optimisation tels que le ”*Design For X*” (DFX), la ”*Conception et Fabrication Assistées par Ordinateur*” (CFAO), ”*l’analyse fonctionnelle*”, etc⁹.

Plusieurs travaux de recherche ont souligné le rôle du processus de développement produit pour l’amélioration de la compétitivité. Rönnbäck [1992] a étudié le cas de la compagnie minière suédoise LKAB¹⁰ qui allait presque mourir suite à la perte de ses avantages compétitifs, durant la période 1960-1970, car elle a été surpassée par ses compétiteurs qui utilisaient des techniques plus avancées dans le transport de masse. La crise internationale qui a touché l’industrie du minerai de fer durant l’année 1980 a affreusement affecté la compagnie LKAB qui, en plus des pertes déjà enregistrées, a perdu plus de la moitié de son marché. Pour récupérer sa position, la compagnie a établi un programme d’actions qui inclut la réévaluation du PDP et la réduction des coûts de production. Elle s’est, ainsi, transformée d’un producteur de matières premières standardisées vendues avec des marges très faibles dans des marchés très concurrentiels à un fabricant de produits personnalisés conçus pour répondre aux besoins spécifiques de ses clients et vendus à des prix records, générant ainsi un rendement du capital très satisfaisant. L’entreprise a, par conséquent, retrouvé son niveau compétitif en 1983, avant même que la crise qu’a connue l’industrie du minerai soit finie. Clark and Fujimoto [1992] se sont intéressés à l’étude de l’industrie automobile et ont affirmé, après

9. Session de formation sur les meilleures pratiques en développement de produits : <https://www.economie.gouv.qc.ca/bibliotheques/formations/formations-mpa-les-meilleures-pratiques-daffaires/meilleures-pratiques-en-developpement-de-produits/>

10. <https://www.lkab.com/en/>

avoir étudié les données collectées de plus de 20 constructeurs automobiles, que la performance d'un PDP, en termes de renouvellement de produit, variation des gammes et diminution du délais de production, joue un rôle très important dans la détermination de la position compétitive d'une entreprise.

Le triangle classique {coût (de produit et de production), qualité (de produit), délais (de production)} est critique vis-à-vis de l'amélioration de la compétitivité d'une entreprise. Après l'analyse des données récupérées des entreprises manufacturières australiennes, [Prajogo \[2007\]](#) montre que la qualité des produits peut efficacement être exploitée dans la réalisation de la stratégie compétitive de l'entreprise. [Li \[2000\]](#) montre que le contrôle qualité lors de la phase de production et le respect des délais de production contribue dans l'amélioration de la compétitivité d'une entreprise. [Porter \[2011\]](#) a exposé les trois principaux moyens pour réaliser un avantage compétitif durable, il s'agit de la maîtrise des coûts (minimisation des coûts de production), la différenciation (des produits en les rendant uniques ou plus attractifs que ceux des concurrents) et la concentration (sur des marchés de niche). Une meilleure maîtrise des coûts, qui se traduit par la diminution des coûts de produit et de production en étant efficace et efficiente (comme déjà discuté), permet d'accroître la compétitivité. [Buckley et al. \[1988\]](#) différencient entre les notions de coût et de prix. Ils montrent que plus l'entreprise encoure de faibles coûts (de production, de main d'œuvre, de matière première, etc.), plus elle est compétitive. Pareillement, plus les prix des produits proposés sont faibles, plus l'entreprise monte en compétitivité. Ils soulignent, en revanche, que la compétitivité des prix doit être associée à plusieurs autres facteurs, notamment la qualité, pour générer les profits et maintenir la compétitivité.

Innovier pour se différencier des autres entreprises : L'innovation a suscité un énorme intérêt ces dernières années. C'est un concept qui fait partie des trois sous-indices à partir desquels est calculé l'indice de la compétitivité globale du rapport annuel de la compétitivité globale publié par le Forum Economique Mondial (FEM)¹¹. Plusieurs définitions de l'innovation sont présentes dans la littérature. Le dictionnaire Oxford⁸ la définit comme étant "*the introduction of a new method, idea, product, etc.*". L'innovation est définie par l'organisation de coopération et de développement économiques (OCDE) comme étant l'introduction d'un produit, service, méthode ou processus qu'il soit nouveau ou *considérablement amélioré* en termes de spécifications techniques, composants, matières, ou toute autre caractéristique fonctionnelle [[OCDE, 2005](#)]. Ces définitions affirment que tout nouveau produit, ou produit sensiblement amélioré, est innovant. [Sawhney et al. \[2006\]](#) définissent l'innovation comme étant la création d'une nouvelle valeur pour les clients et non pas que la création des nouveaux produits parce que cette dernière n'est, selon [Schumpeter \[1934\]](#), ni nécessaire ni suffisante pour l'innovation.

Il est admis que l'innovation est une nécessité pour s'adapter aux conditions changeantes du marché, c'est le moteur de la compétitivité par excellence [[Lengnick-Hall, 1992](#)]. William Ford Jr., directeur général du constructeur automobile *Ford Motor Company*¹² qui a été classé, par *Forbes*⁶ en fin 2011, parmi les constructeurs dont les voitures sont les plus vendues au monde (à travers son *Ford Focus* et *Ford Fiesta* classées respectivement dans la 4^{ème} et 6^{ème} position¹³), a annoncé que "*... from this point onward, innovation will be the compass by which the company sets its direction.*" et que Ford Motor company "*will adopt innovation as its core business strategy going forward*" [[Ford, 2005](#)]. Dans le monde de la téléphonie mobile, l'entreprise coréenne "*Samsung*" qui était largement dépassée par le géant japonais "*Sony*", qui a révolutionné le marché à travers le produit Sony walkman [[Beverland et al., 2006](#)], a adopté une approche innovatrice et a commencé à développer des smartphones haut de gamme avec des prix élevés; l'entreprise réalise ainsi un profit important. Un excellent exemple de l'innovation est l'entreprise américaine "*Apple*". Elle se place au-dessus du lot parce qu'elle a toujours été innovatrice en proposant des produits très attractifs avec des prix très élevés (parfois exorbitants) mais qui se dotent d'une bonne qualité et utilisent les dernières technologies. La marque à la pomme qui a révolutionné l'industrie de la téléphonie mobile à travers l'iPhone [[Beverland et al., 2006](#)], a su créer le besoin chez les consommateurs en leur proposant, vers la fin 2009, un nouveau produit innovant : l'iPad. Ce dernier, qui peut objectivement être remplacé par un téléphone

11. <https://www.weforum.org/>

12. <http://corporate.ford.com/homepage.html>

13. <https://www.forbes.com/sites/joanmuller/2011/12/25/the-worlds-most-popular-cars-are-changing/#71b5eb14180e>

portable et un ordinateur (que presque tout le monde possède déjà), a connu une forte réussite au niveau international, toutes tranches d'âge confondues.

L'innovation peut se manifester à plusieurs niveaux dans une entreprise [Schumpeter, 1976, 1934] : niveau produit, processus et organisation, entre autres. L'innovation produit fait intervenir des nouvelles technologies, nouvelles connaissances ou des combinaisons des connaissances et/ou technologies déjà existantes. L'innovation de processus est l'adoption des méthodes de production nouvelles ou améliorées. Ces deux types contribuent à la réduction des coûts et temps de production et permettent donc d'accroître le rendement et l'efficacité de la production. L'innovation d'organisation est l'implémentation des nouvelles méthodes organisationnelles dans les pratiques de management de l'entreprise (organisation de travail, coopérations, etc.), elle permet d'améliorer la performance en réduisant, entre autres, les coûts administratifs qui ne contribuent pas directement à la fabrication d'un produit (par exemple, les avantages sociaux pour le personnel).

2.1.3 Discussion

Dans la Section 2.1.2, nous avons listé les facteurs qui seraient, à notre opinion, les plus intéressants pour atteindre la compétitivité. Cette dernière peut être évaluée à travers plusieurs indicateurs [Sirikrai and Tang, 2006], dont les plus connus sont la productivité, la capacité d'innover et les parts de marché (les deux premiers indicateurs impliquent le dernier).

La productivité est directement liée au *développement produit* et à *l'innovation produit*, que nous avons introduits dans la sous-section précédente. Ces deux dernières notions sont sujettes à des interprétations parfois erronées. Madame Lucy MacLennan, Expert technique de l'industrie alimentaire, a clarifié la différence à travers sa discussion ¹⁴ : le développement produit est vu comme étant une innovation produit lorsqu'il s'agit d'un produit nouveau. Roozenburg and Eekels [1995] ont listé les étapes principales qui constituent le processus d'innovation produit (Figure 2.1). Dans la phase de *plannification de produit* est décidé quel produit développer et quand. Dans la phase *développement strict* est généré le plan détaillé du produit et de la production. Dans la phase de *réalisation*, les plans détaillés issus de la phase précédente sont transformés en réalité.

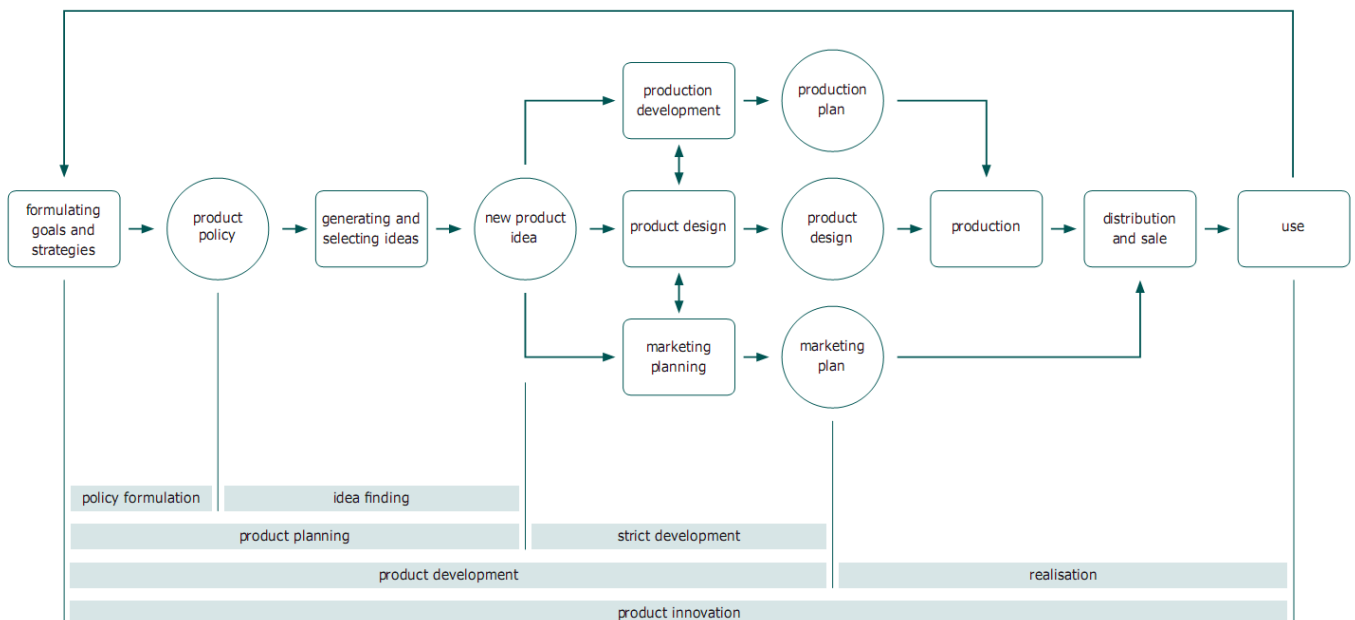


FIGURE 2.1 – Les phases de l'innovation produit selon Roozenburg and Eekels [1995].

14. <https://www.linkedin.com/pulse/what-difference-between-innovation-product-lucy-maclennan/>

Le processus de développement produit est non seulement un facteur de compétitivité, mais il contribue aussi à l'innovation [Brown and Eisenhardt, 1995] qui elle aussi est un facteur déterminant de la compétitivité, comme nous l'avons décrit sur la Figure 2.2. Par conséquent, la question "Comment être compétitif?", que nous nous sommes posée au début, sera réduite - dans ce travail de thèse - à la question "Comment maîtriser le processus de développement produit?".

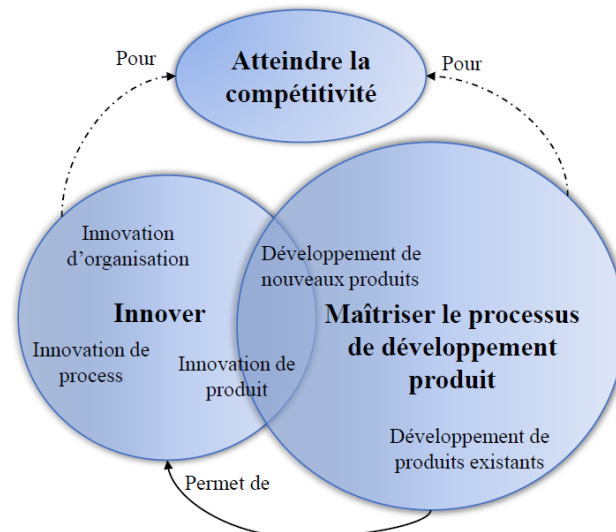


FIGURE 2.2 – Le triangle {Compétitivité, Innovation, Développement produit}.

2.2 Processus de développement produit (PDP)

C'est un processus dans lequel un produit est conçu, mis en œuvre et commercialisé [Ulrich and Eppinger, 2015]. Il prend les besoins clients et les transforme, à l'aide des ressources internes de l'entreprise (connaissances, savoir-faire, technologies, etc.), en solutions techniques sous la forme de produits / services [El Gamoussi, 2016]. C'est un processus via lequel une entreprise exécute une des deux tâches :

- Améliorer ou mettre à jour le produit qui existe déjà en le rendant plus performant (grâce à l'utilisation d'une nouvelle technologie, nouveaux composants, etc.),
- Créer (développer) un nouveau produit qui, n'ayant jamais existé auparavant, s'ajoute à une gamme déjà existante ou qui occupe une nouvelle niche. Cette tâche est connue sous le nom de *Développement de Nouveaux Produits* (NPD), elle est vue comme une partie de l'innovation produit (Figure 2.1 et Figure 2.2).

2.2.1 Étapes du PDP

Plusieurs travaux de recherche se sont intéressés à la définition des étapes du processus de développement produit [Andreasen and Hein, 1987; Quarante, 1994; Cavaillès et al., 1995; Ullman, 2003; Tomiyama et al., 2009; Fanchon, 2010; NASA, 2010b; Parnell et al., 2011; Buede, 2011; Pahl et al., 2013; Ulrich and Eppinger, 2015; INCOSE, 2015; BKCASE, 2017]. De ces approches et modèles, nous reprenons le processus présenté dans la Figure 2.3.

L'entreprise commence d'abord par définir les besoins des clients, ou du marché cible, pour ainsi formuler les spécifications du ou des produits qu'elle veut introduire. Elle mène ensuite une étude de faisabilité pour analyser, entre autres, la rentabilité du produit, les implications économiques de l'entreprise, la possibilité de la réalisation du produit d'un point de vue technique, organisationnel, etc. Dans le cas où l'étude de faisabilité est satisfaisante, une équipe pluridisciplinaire commence par proposer un maximum de solutions (alternatives) qui répondent au "pourquoi?" du produit, puis choisit la solution la plus performante qui

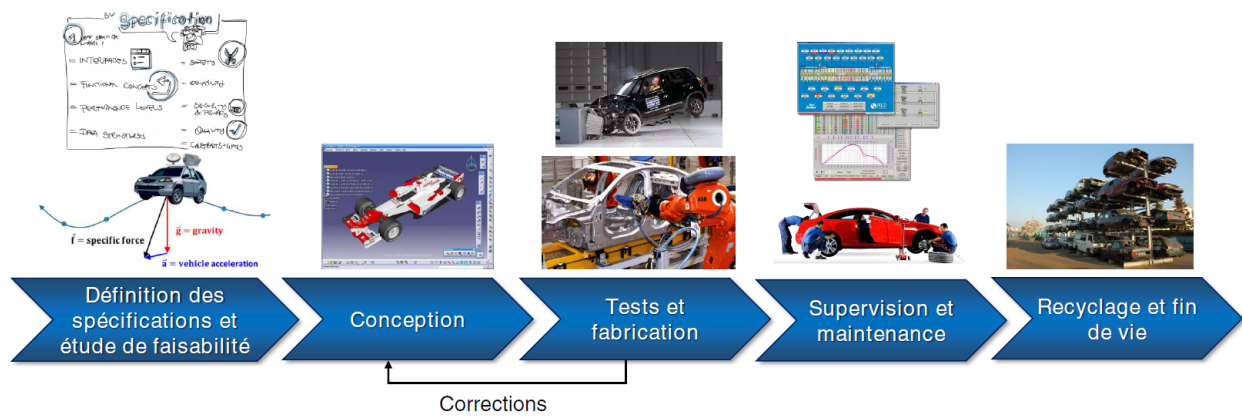


FIGURE 2.3 – Définition du processus de développement produit dans le cadre de notre contexte.

permet, entre autres, de minimiser les coûts et temps de production. Le produit conçu sera ensuite soumis à des tests avant de lancer sa fabrication et commercialisation.

Une fois fabriqué et vendu, un produit - qu'il soit matériel (*produit physique*), immatériel (*service*¹⁵) ou rassemblant les deux (*système*¹⁶) - doit être maintenu par l'entreprise pour maximiser sa fiabilité et rallonger sa durée de vie pour que l'entreprise reste toujours crédible auprès de ses clients en augmentant, en même temps, le retour sur son investissement. En plus d'être maintenus, les *services* ou les *systèmes* doivent être supervisés (surveillés) pour garantir leur bon fonctionnement ou, dans le cas échéant, signaler un mal-fonctionnement pour lancer l'activité de maintenance comme le montre la Figure 2.4. Au moment où le produit arrive à sa fin de vie, il est recyclé et exploité en vue de la production d'un nouveau produit.

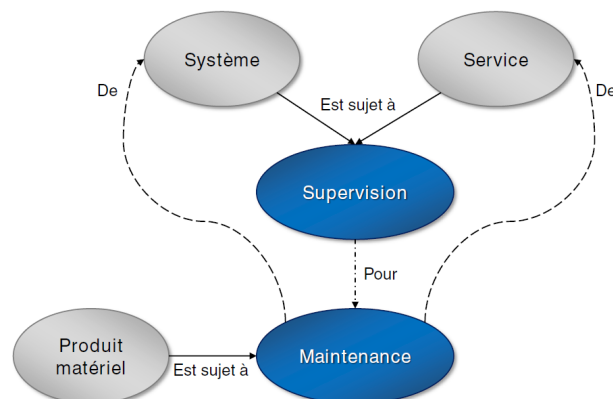


FIGURE 2.4 – Carte conceptuelle de la supervision et la maintenance des produits matériels, des services et des systèmes.

2.2.2 Maîtriser le PDP

Morgan and Liker [2006] considèrent le PDP comme un des processus les plus complexes d'une entreprise. En effet, la plupart des problèmes que rencontre cette dernière se situent dans le PDP [Clark and Fujimoto, 1991]; nous citons, comme exemple de ces problèmes, la mauvaise compréhension des besoins des clients qui peut donner lieu à un produit non souhaité [Clark and Fujimoto, 1991]. Le PDP est défini comme étant un ensemble d'activités interdépendantes qui manipulent des connaissances [Clark et al., 1987; Morgan and Liker, 2006] et qui peuvent être classées en deux catégories : activités de prise de décision et activités

15. Produit immatériel, par exemple, le suivi 24h/24 et mise en sécurité des réseaux d'exploitation de gaz ou encore l'exemple des logiciels SaaS - Software as a Service - installés sur des serveurs distants plutôt que sur la machine de l'utilisateur. Ce dernier s'abonne au service et ne paie pas la licence d'utilisation de ces logiciels.

16. Ensemble d'éléments matériels et immatériels interagissant entre eux pour produire un produit ou un service [Bellagamba, 2012], par exemple, le système de production.

d'exécution [El Gamoussi, 2016]. Dans les activités de prise de décision, les ingénieurs choisissent les paramètres optimaux à prendre en compte dans la suite du processus comme la technologie, apparence, méthodes d'industrialisation, etc. Cette première catégorie est suivie par les activités d'exécution qui ont pour objectif la réalisation physique du produit qui a été décrit, elles sont par conséquent affectées par les décisions prises en amont. En effet, l'enquête de la *Product Development Management Association (PDMA)*¹⁷ menée en 2010 et qui a eu comme objectif l'étude de la performance du processus de développement produit, a montré que :

- 80% des projets ne respectent pas les délais prévus, 50% d'entre eux dépassent plus de 60% des délais,
- 20% à 50% des coûts sont dus aux **retours pour corrections**.

Plusieurs travaux de recherche se sont intéressés à la maîtrise du PDP. Tomiyama et al. [2009] ont classé et évalué les méthodes de "*Design Theory and Methodology (DTM)*", permettant de maîtriser une ou plusieurs étapes du processus de développement produit, selon plusieurs points de vue (abstrait, concret, éducation, pratique, etc.). La méthode de la conception axiomatique, développée par Suh [2001] au MIT³, a pour but de maîtriser la phase de conception. Elle utilise les méthodes mathématiques matricielles pour transformer les besoins clients en besoins fonctionnels et repose sur deux axiomes : maximiser l'indépendance entre les éléments fonctionnels et minimiser la quantité d'information dans la phase de conception. L'ingénierie concurrente¹⁸ (concurrent engineering), appelée aussi ingénierie simultanée, consiste à démarrer toutes les étapes du projet simultanément et réunir tous les acteurs pour la compréhension collective des objectifs du projet [Hartley, 1990]. Elle permet de maîtriser le PDP parce qu'elle permet d'améliorer la qualité du produit, réduire le temps de son développement et diminuer sensiblement les retours pour corrections, en anticipant la détection des problèmes potentiels. L'ingénierie système est une approche qui, en se focalisant sur la définition des besoins clients, l'analyse et documentation des exigences fonctionnelles, permet de maîtriser la conception, le développement et la vérification des systèmes complexes en s'appuyant sur la décomposition architecturale [INCOSE, 2015]. La méthode "*Set-Based Concurrent Engineering (SBCE)*" [Sobek et al., 1999], développée par le constructeur automobile japonais "Toyota Motor Corporation"¹⁹ connu par son système de production, adapte la méthode "*Lean*" visant à la diminution du gaspillage sans toucher à la productivité, à la phase de conception en développant un maximum de solutions possibles (alternatives) puis, en suivant un processus d'entonnoir, restreindre l'espace des solutions et converger vers la solution optimale, elle passe ainsi de la logique "*Design then Test*" à la logique "*Test then Design*" et diminue, par conséquent, les retours pour correction. En se basant sur l'approche Lean, El Gamoussi [2016] a développé une méthodologie permettant de piloter l'amélioration du PDP en identifiant où et quand la maîtrise technique et l'amélioration organisationnelle devront être apportées. La méthode "*Design for X (DFX)*" permet d'améliorer la conception par rapport à une perspective représentée par un "X" [Huang, 1996], plusieurs interprétations de ce "X" existent selon Tichem [1997] (coût, qualité, délais, manufacturability, distribution, etc.). Le Design For Manufacturability (DFM), par exemple, a pour objectif d'optimiser la fabrication des produits en traitant les problèmes potentiels de la fabrication tôt dans le processus de développement produit, c'est à dire pendant la phase de conception. Les "*méthodes de prise de décision relatives à la conception*" ont pour but de maîtriser la conception, elles s'intéressent au choix optimal d'une ou plusieurs alternatives, parmi toutes les alternatives possibles, par rapport à un objectif donné [Tomiyama et al., 2009].

Plusieurs outils numériques supportent la maîtrise du processus de développement produit. L'approche de la gestion du cycle de vie produit (Product Lifecycle Management - PLM) qui englobe tout le processus de développement produit [Garetti et al., 2005] permet de gagner en efficacité, en productivité et en intelligence, entre autres. Le modèle PLM a l'avantage, par rapport aux modèles linéaires (parcourant les activités du PDP d'une façon linéaire une après l'autre), d'être centralisé, rapide, intelligent et

17. <http://www.pdma.org/>

18. Attention à la différence sémantique entre concurrente et concurrente

19. <http://www.toyota-global.com/>

connecté. La fonction de gestion de données produit (Product data management - PDM) [Philpotts, 1996] qu'englobent les outils PLM, permet d'organiser et capitaliser les données produit. Les outils de gestion de connaissances²⁰ [Rao, 2012] permettent aussi de supporter le processus de développement produit en capitalisant et réutilisant les connaissances de l'entreprise.

2.2.3 Discussion : contexte de recherche

La plupart des méthodes de maîtrise du PDP, que nous venons de lister, concernent la conception. En effet, cette dernière est un des processus les plus critiques du PDP parce que :

- C'est une phase d'innovation où les idées et/ou exigences des clients sont transformées (formulées) en donnant lieu à plusieurs choix de décision. Yannou [2008] a souligné que l'entreprise doit être performante en conception pour qu'elle puisse efficacement innover.
- Elle définit l'allure du produit final, au niveau physique, esthétique et fonctionnel. Le produit final dépend donc des choix décisionnels pris en considération (chemin décisionnel suivi).
- C'est une phase qui, dans le cadre de l'ingénierie concurrente, intègre très tôt toutes les contraintes du cycle de vie du produit [Sohlenius, 1992].
- Les erreurs de conception introduites dans cette phase auront un impact majeur sur le produit final (le produit ne répond pas à la fonction à la quelle il devait répondre, ou il a une forme différente de celle prévue, etc.).
- L'incompréhension et la mauvaise formulation des besoins clients au niveau de la phase de conception. En effet, une formulation erronée des exigences donnera lieu à un produit différent de celui demandé par les clients.
- C'est une phase de connaissances [Tomiyama et al., 2009] et de prise de décision [Berliner and Brimson, 1988].

En outre, une autre raison de l'échec dans le processus de développement produit est une supervision non rigoureuse des systèmes de développement produit et des services issus du processus de développement produit. Chase et al. [2004] affirment que la supervision des systèmes responsables de l'utilisation productive des ressources est directement liée au maintien de la compétitivité de l'entreprise. En effet, si le système de production n'est pas bien supervisé, les situations anormales que peut vivre le PDP ne seront pas reconnues et corrigées au bon moment, ce qui diminue la réactivité de l'entreprise. L'entreprise doit également superviser ses services pour rallonger leur durée de vie et entretenir sa crédibilité auprès de ses clients.

Par conséquent, une *bonne maîtrise du processus de développement produit* se traduit par la *bonne maîtrise des processus de conception et de supervision* (présentés dans la Figure 2.3). Ainsi, nous formulons le contexte dans lequel s'inscrit notre travail de recherche :



Contexte de recherche

Nos travaux de recherche s'intéressent aux processus de **conception** et **supervision** des systèmes, étant des **processus métiers, collaboratifs, créatifs** et **dynamiques**.

La Figure 2.5 montre l'évolution des objectifs industriels qui a donné lieu à notre contexte de recherche dont les mots clés (écrits en gras) sont définis comme suit :

- Un **processus** est, selon Theroude [2002], un enchaînement d'**activités** qui, en se basant sur des ressources techniques et humaines, transforment des données d'entrée en données de sortie en vue de réaliser un objectif donné. Un processus **métier** est, selon Davenport [1993], un ensemble d'activités

20. Comme l'application BPR (Business Process Re-Engineering), développée par Ford Motor, ayant pour objectif de supporter le processus manuel existant visant le partage des améliorations de processus entre les usines d'assemblage de véhicules [Rao, 2012].

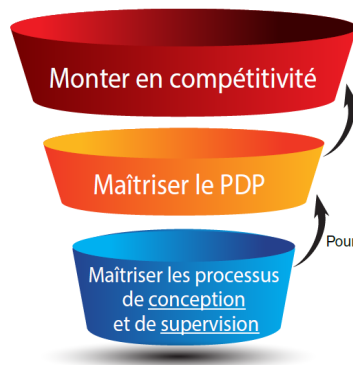


FIGURE 2.5 – Évolution des objectifs industriels.

structurées ayant une sortie spécifique à un métier donné. Il est défini par [Weske \[2012\]](#) comme étant un ensemble d'activités qui sont exécutées en coordination dans un environnement technique et organisationnel. Il est aussi défini par [Davenport and Short \[2003\]](#) comme étant un ensemble de tâches logiques reliées entre elles et qui, en s'exécutant, réalisent un objectif bien défini dans l'entreprise. Dans notre contexte, nous définissons le processus métier comme étant un ensemble d'activités qui, après leur exécution, produisent une valeur ajoutée tangible à partir des spécifications initiales et réalisant ainsi un objectif répondant aux besoins d'un client interne ou externe à l'organisation.

- Une **activité** selon [van der Aalst et al. \[2007\]](#) (appelée aussi tâche, opération, action ou élément de travail) est définie comme étant une opération d'une instance de processus en cours d'exécution. Elle est définie par [Godart and Perrin \[2009\]](#) comme étant une étape logique à l'intérieur d'un processus, exécutée en utilisant des ressources. Dans le cadre de notre contexte, nous définissons une activité comme étant le plus petit élément logique indivisible d'un processus qui, après son exécution, produit une sortie adéquate avec son objectif, en respectant les contraintes imposées et en utilisant les ressources disponibles (humaines, matérielles, logicielles, etc.).
- Un processus est dit **collaboratif** s'il est exécuté par un groupe d'acteurs pluridisciplinaires appartenant ou pas à l'entreprise. La collaboration est définie par [Sandler \[1992\]](#) comme étant un processus où plusieurs personnes travaillent ensemble d'une façon interdépendante pour atteindre un objectif meilleur que celui atteint si chacune de ces personnes travailla toute seule. [Pahng et al. \[1998\]](#) affirment que la collaboration est primordiale pour aborder un problème complexe, parce que ce dernier nécessite des connaissances spécialisées, venant de plusieurs domaines, souvent non maîtrisées par un seul acteur.
- Un processus **créatif** est défini par [Howard et al. \[2007\]](#) comme étant un processus cognitif aboutissant à la génération d'une idée. Il est défini par [Godart and Perrin \[2009\]](#) comme étant un processus de haute valeur ajoutée caractérisé par une grande complexité, une variabilité d'une application à une autre et une forte dynamique (processus adapté au cours de son exécution suite aux changements fréquents). Il est aussi défini par [Perrin \[2010\]](#) comme étant un processus dynamique potentiellement de longue durée et propice aux changements. Les processus de conception et de supervision, délimitant notre contexte de recherche, sont deux exemples de processus de travail intensif aboutissant à la proposition d'un produit, procédé ou service et sont considérés comme processus créatifs puisqu'ils sont d'une grande valeur ajoutée et sont propices aux changements [[Howard et al., 2007](#)].
- Un processus **dynamique**, dans notre contexte, est un processus dont les activités s'organisent d'une façon flexible [[Daoudi and Nurcan, 2007](#)] suite aux changements de contexte.
- La **conception** est un processus **métier, collaboratif** [[Badke-Schaub and Gehrlicher, 2003](#)], complexe, de prise de décisions et manipulant des connaissances partagées et exploitées par plusieurs acteurs [[Olsen et al., 1994](#)]. C'est un processus **dynamique**, parce qu'il est sujet aux changements (technologiques, organisationnels, des besoins du marché, etc.), et **créatif** qui produit un produit ou service de haute valeur ajoutée [[Lawson, 2006](#)] répondant aux besoins de l'utilisateur et des parties

prenantes [Blessing and Chakrabarti, 2009].

- La **supervision** est un processus **métier, collaboratif**, complexe manipulant des connaissances, critique puisque une éventuelle anomalie doit être traitée le plutôt possible pour éviter les dégâts (matériels, humains, etc.). Il a pour objectif le suivi et la garantie du bon fonctionnement d'un service ou d'un système [Leyval et al., 1994; Acosta et al., 2001].

2.3 Problématique, question de recherche, hypothèse et caractéristiques du système recherché

2.3.1 Problématique : les besoins industriels

La compétitivité d'une entreprise repose, entre autres, sur la maîtrise des processus de conception et/ou de supervision des systèmes vu qu'ils sont les processus les plus critiques et complexes du PDP, comme discuté dans la Section 2.2.3. Selon les retours d'expérience dans les industries, plusieurs problèmes peuvent se poser lors de l'exécution de ces processus, nous listons ci dessous les problèmes que nous supposons être les plus importants :

Problème 1 - Qualité des décisions prises : Dans les phases de conception et de supervision, les acteurs échangent les informations produit/service/système ainsi que leur savoir-faire. Ceci donne lieu à plusieurs choix décisionnels dont celui qui est pris en compte donnera lieu au produit/service/système final. Ce sont deux phases itératives puisque les produits conçus qui ne franchissent pas la phase de test devront être retournés pour une éventuelle re-conception (Figure 2.3), et les services/systèmes supervisés qui présentent une anomalie de fonctionnement, sont sous action de résolution de problème jusqu'à ce que leur fonctionnement redevienne normal. Schonberger [1982] souligne que 85% des problèmes rencontrés au niveau du processus de production sont liés aux décisions prises dans la phase de la conception. Berliner and Brimson [1988] montrent que 85% des décisions prises dans cette phase impactent plus que 80% du coût final du produit (Figure 2.6). Anderson [2001] montre que malgré le fait que la phase de conception ne représente que 5% des coûts du projet, elle impacte plus de 60% sur le coût final du produit. Il a aussi montré qu'une bonne maîtrise du processus de conception engendre une baisse significative des coûts totaux du projet (coût de fabrication, matériel, etc.). Les retours en arrière, qui entraînent un coût supplémentaire et un dépassement au niveau du délais de production, sont dus aux mauvaises décisions prises et sont significativement réduits si la phase de la conception est **bien maîtrisée** et si, en plus du point précédent, les considérations opérationnelles sont prises en compte tôt dans le PDP, c'est à dire au niveau du processus de conception. En effet, le coût d'une correction effectuée au niveau de la phase de production est multiplié par 500 à 1000, contre seulement 3 à 6 fois en phase de conception, comme le montre la Figure 2.7.

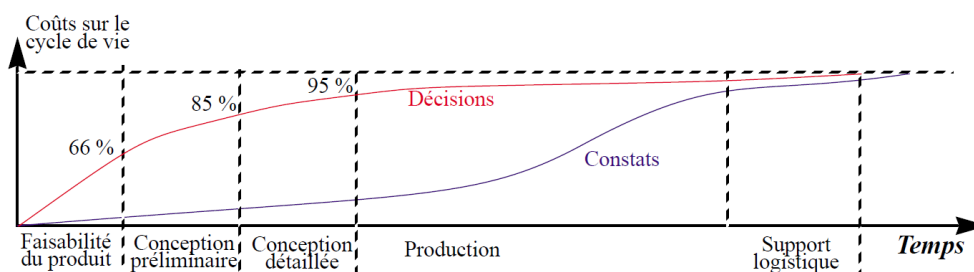


FIGURE 2.6 – Influence des décisions prises sur les coûts [Gautier and Glard, 2000] (adapté de [Berliner and Brimson, 1988])

Problème 2 - Perte de temps suite à la recherche de l'information : Les ingénieurs ont besoin de l'information pour explorer une idée, raffiner une solution, résoudre un problème et prendre des décisions, entre autres [Kwasitsu, 2003]. FREY et al. [2007] et Ullman [2003] ont montré que les ingénieurs passent

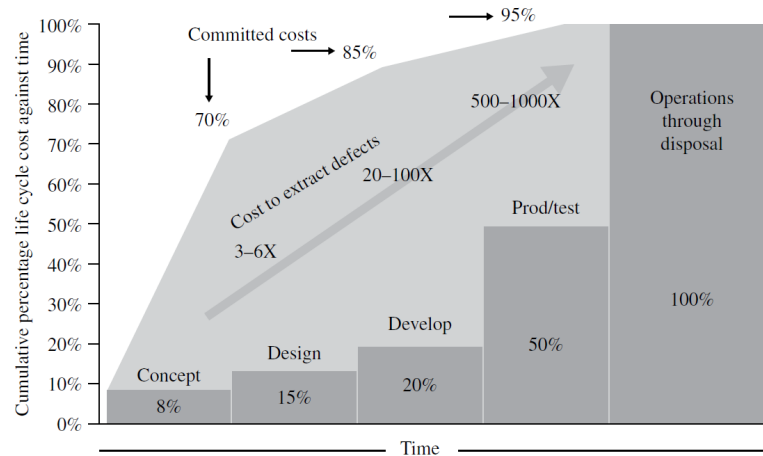


FIGURE 2.7 – Coûts des corrections en fonction du temps [INCOSE, 2015]

un temps conséquent par rapport à la durée de conception, lors de la recherche de l'information nécessaire à la finalisation de leurs tâches. En effet, ce problème est dû, entre autres, à l'existence des différentes sources d'information (moteurs de recherche, bibliothèques, conférences, associations, personnes, mémoire personnelle, fichiers personnels, etc.) Kwasisu [2003]. Les mêmes chercheurs ont montré que 72% des personnes, parmi les 35 interrogés, utilisaient comme source d'information, leurs mémoires, fichiers personnels et les personnes appartenant à leurs business groupes. Ceci entraîne un temps supplémentaire lors de la recherche de l'information (recherche des fichiers, essayer de se rappeler de l'information, attendre la réponse -souvent non immédiate- d'une personne, etc.).

Problème 3 - Perte de l'information (décisions passées) : Après avoir passé des semaines, voire des années, pour concevoir un produit réussi qui a bien répondu aux besoins du marché, les ingénieurs affirment "to not even being able to remember how their group developed the main idea for a design." et que "at times we have tried to remember who had a particular idea, and have usually found we can't" [Lawson, 2006]. Cette perte d'information empêche l'entreprise de réutiliser ses expériences passées pour réussir un nouveau produit.

Problème 4 - Perte de temps suite à l'échange des données entre les collaborateurs : Comme nous l'avons déjà décrit, la conception et la supervision des systèmes sont deux processus collaboratifs qui regroupent plusieurs équipes pluri-disciplinaires, souvent géographiquement dispersées. Ceci introduit un temps supplémentaire nécessaire pour échanger les données, les traduire, les déchiffrer, etc. En effet, l'échange de données est vu comme un processus risqué (perte d'information, incohérence entre les données échangées), qui consomme beaucoup de temps et parfois compliqué du point de vue légal (confidentialité des données) [Thomas, 2009]. De plus, l'échange des données créées et sauvegardées sous des formats différents entraîne des problèmes d'interopérabilité, et crée ainsi le besoin d'une activité supplémentaire pour assurer la continuité numérique.

Problème 5 - Perte de temps lors de la prise de décision : En plus de la qualité des décisions, étant le premier problème industriel discuté, le temps pendant lequel ces dernières sont prises a une influence majeure sur la performance. En effet, dans une organisation ayant une structure hiérarchique, les salariés ne sont pas toujours techniquement capables de prendre les bonnes décisions dans un laps de temps très court sans passer par le processus d'approbation à plusieurs niveaux, qui n'est pas souvent immédiat [Mallak, 1998]. En outre, les investigations empiriques montrent que la plupart des décisions, lors de la conception d'un produit, s'étendent sur une période de temps et ne sont pas instantanément faites [Hansen and Andreasen, 2004]. Les ingénieurs utilisent, en effet, les informations, les considérations et les critères disponibles pour faire des décisions dites "provisaires" ou "tentatives", qui ne sont validées

que lorsque l'étude de leurs conséquences atteint un niveau satisfaisant ou un seuil prédéfini [Hansen and Andreasen, 2004]. Badke-Schaub and Gehrlacher [2003] ont effectué une étude empirique dans une moyenne entreprise qui a pour activité d'équiper l'industrie automobile avec des dispositifs automoteurs, entre autres. Le département conception de l'entreprise compte 11 employés divisés en deux unités, une qui fait la conception et l'autre teste et valide le produit conçu. Au cours de cette étude qui s'est étendue sur deux semaines, 40 décisions ont été identifiées et analysées. Le résultat a montré que :

- Le processus de conception se fait d'une manière collaborative avec les autres départements et unités de l'entreprise. En effet, au cours des 40 décisions analysées, l'unité de conception a participé dans 21 décisions, le département de gestion dans 19, l'unité de test dans 28 et les autres départements ont participé dans 13 décisions. En outre, 11 décisions ont été marquées par une participation des personnes externes à l'entreprise.
- Seulement 45% des décisions ont été planifiées à l'avance et que la majorité des décisions les plus importantes étaient prises d'une façon non planifiée parce qu'elles dépendent d'autres décisions qui devaient être prises avant.

Une autre étude empirique a été effectuée par Mortensen and Tichem [1994] à Bang and Olufsen²¹, société industrielle danoise qui conçoit et fabrique des produits audiovisuels haut de gamme. L'étude avait pour but d'analyser les décisions qui se sont faites au cours de la conception d'un produit donné dont l'objectif était d'avoir une finition argentée. Trois solutions candidates ont été analysées et la solution qui apparaissait la plus prometteuse était "le fraisage aluminium". De ce fait, une décision tentative de production utilisant ce mode d'usinage a été faite en attendant de la vérifier par des expériences. Ceci a permis à l'entreprise de continuer le développement du concept pendant que les expériences étaient entrain d'être menées. Le résultat final des expériences a permis de valider la décision tentative qui est devenue une décision finale, permettant ainsi à l'équipe de commencer l'achat des machines, outils et matériaux nécessaires à la production en parallèle du développement continu du concept.

Des deux études précédentes, nous concluons que la décision n'est pas faite instantanément vu qu'elle fait appel à plusieurs participants, et qu'elle dépend généralement de plusieurs décisions dont chaque retard entraîne un temps supplémentaire.

Problème 6 - Perte de temps sur des activités à non valeur ajoutée : Les activités du PDP peuvent être classées en trois catégories [Oehmen and Rebentisch, 2010] : les *activités à valeur ajoutée* comme l'interaction avec les clients pour comprendre leurs besoins, les *activités nécessaires mais à non valeur ajoutée* comme la manutention de produits et le *gaspillage* comme le retour pour corrections (Figure 2.3). Evers et al. [1998] ont montré que les ingénieurs passent 33% du temps de conception sur des activités à valeur ajoutée, 38% sur des activités nécessaires mais à non-valeur ajoutée, et 29% sur du gaspillage. Oehmen and Rebentisch [2010] ont montré que seulement 12% du temps de conception est consacré aux activités à valeur ajoutée contre 15% consacré pour des activités à non valeur ajoutée (Figure 2.8).

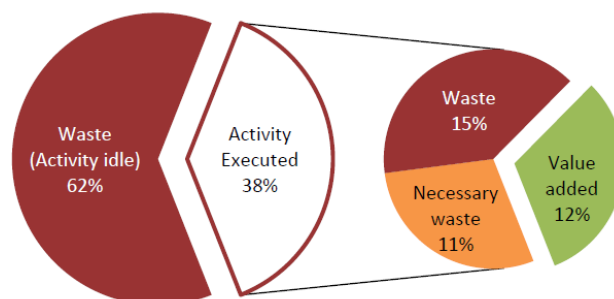


FIGURE 2.8 – Pourcentage du temps pris par les différentes activités du PDP selon [Oehmen and Rebentisch, 2010]

21. <https://www.bang-olufsen.com/fr>

Problème 7 - Perte de temps lors de la gestion des différents changements : L'entreprise vit de plus en plus des changements internes et externes, qui peuvent venir des nouveaux besoins du marché, d'une nouvelle technologie, nouveau responsable, nouvelle structure, nouveau contexte, etc. Pour qu'elle continue d'être performante, l'entreprise doit faire preuve d'une bonne réactivité et fluidité en gérant et propageant ces changements dès la phase de conception pour ne pas subir de grosses pertes de temps et d'argent [Berliner and Brimson, 1988]. Quelques changements peuvent survenir en temps réel lors de la prise de décision et nécessitent d'être propagés instantanément pour permettre au décideur d'en tenir compte et de prendre, ainsi, une décision efficace [Burstein et al., 2011]. Comme exemple de ce type de changements, nous citons le rythme cardiaque qui peut changer considérablement au cours d'une opération chirurgicale critique, et dont la valeur en temps réel (à l'instant t_i) doit être prise en compte par le chirurgien lors de sa décision à l'instant t_i .

Problème 8 - Non maîtrise du temps de développement : Mano et al. [2007] ont montré que le temps entre la décision de développer un produit et son développement effectif est énorme. L'étude réalisée en 2004 par le *Gartner Group*²² sur l'échec des projets informatiques a montré que 66% des projets ont échoué parce qu'ils étaient en retard et/ou parce qu'ils n'ont pas respecté les besoins clients. Le dépassement du temps de développement est principalement dû aux problèmes mentionnés dans les paragraphes précédents ainsi qu'à l'absence d'une **supervision** rigoureuse des processus compris dans le développement produit [Combacau et al., 2000].

2.3.2 Objectifs scientifiques et question de recherche

Nous avons listé, dans la section précédente, les principaux problèmes rencontrés lors de la conception et supervision des systèmes. Ces problèmes peuvent être classés en trois catégories :

1. **Recherche et échange de l'information** : problèmes 2, 3 et 4.
2. **Prise de décision et réutilisation des décisions passées** : problèmes 1, 5, 6 et 8.
3. **Propagation des changements** : problème 7.
 - Pour aborder la première catégorie des besoins industriels, il serait intéressant de développer une méthode qui permettra de capturer l'information et de faciliter sa quête tout en permettant à n'importe qui de visualiser le processus en question et retrouver l'information qu'il cherche à n'importe quel moment. La méthode doit permettre de réutiliser ces informations lors du lancement d'un nouveau produit pour ainsi accélérer les étapes de lancement. En outre, La méthode doit permettre une gestion centralisée de l'information pour aider les entreprises à optimiser la collaboration et les échanges de données entre les différentes équipes concernées.
 - Pour apporter des solutions à la deuxième catégorie de besoins, il serait intéressant de développer une méthode qui permet d'analyser et d'évaluer les décisions passées pour pouvoir proposer la bonne décision au bon moment à la bonne personne.
 - Pour pouvoir propager les changements contextuels, qui surviennent au cours de la prise de décision, il serait intéressant de développer une méthode qui les identifie d'abord et les propage au bon moment et au bon endroit.

Ces objectifs scientifiques nous emmènent à la question de recherche suivante, qui contient des sous-questions élémentaires dont chacune concerne une des catégories de problèmes définies précédemment :

22. <https://www.gartner.com/technology/home.jsp>



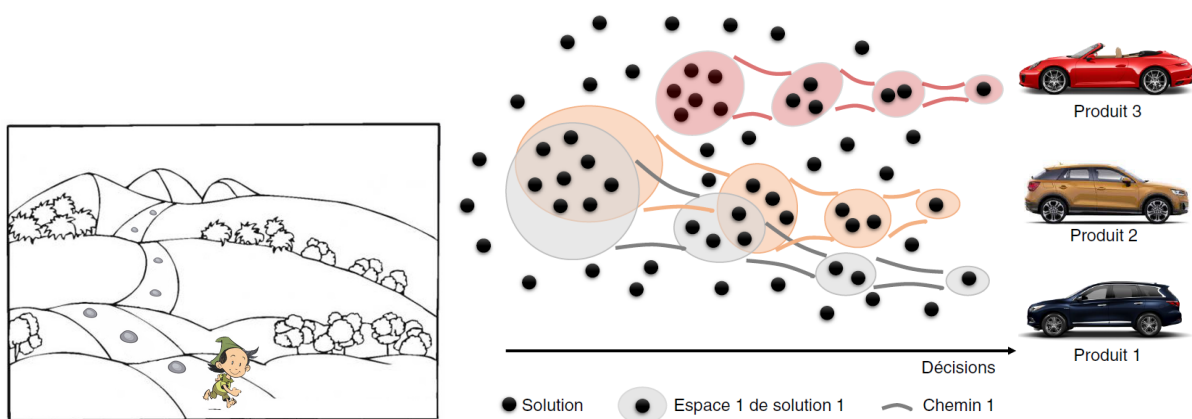
Question de recherche : verrou scientifique 1

Comment **identifier** et **visualiser** les processus de conception et de supervision des systèmes sous la forme d'activités inter-connectées ? Comment **capitaliser** : **Qui** ? a fait **Quoi** ? **Quand** ? **Comment** ? **Où** ? Et **Pourquoi** ? Comment analyser ces informations pour **aider les ingénieurs à décider** des choix de conception ou de supervision ? Et comment **identifier et propager les changements contextuels** pour mettre à jour le processus décisionnel ?

2.3.3 Hypothèse de travail

Notre objectif tracé dans la Figure 2.5, consiste à maîtriser les processus de conception et de supervision des systèmes en répondant aux besoins industriels listés dans la Section 2.3.1 et à la question de recherche qui en résulte. À l'instar du Petit Poucet (Figure 2.9.a) qui "... les laissait crier, sachant bien par où il reviendrait à la maison ; car en marchant il avait laissé tomber le long du chemin les petits cailloux blancs qu'il avait dans ses poches...", nous souhaitons maîtriser nos processus en donnant une description exhaustive du chemin décisionnel suivi et qui mènera au produit/système/service final (Figure 2.9.b), tout en mémorisant tous les choix rejetés et en justifiant la raison du rejet. Ceci permettra à l'entreprise de pouvoir revenir en arrière pour examiner tous les chemins décisionnels afin d' :

- Améliorer un produit existant en modifiant ou adaptant son chemin décisionnel.
- Innover en concevant un nouveau produit par une combinaison des chemins décisionnels d'un ou de plusieurs produits déjà existants ou en explorant les autres chemins possibles.



2.9.a. Maîtriser le chemin pour savoir revenir. 2.9.b. Exemple des chemins décisionnels de la conception d'une voiture.

FIGURE 2.9 – Et si on maîtrisait notre chemin (des décisions prises lors de la conception ou supervision des systèmes) comme a fait le Petit Poucet.

Les entreprises sont composées de plusieurs processus qui peuvent être complexes et corrélés les uns aux autres. Il est quasiment impossible de faire des analyses poussées et d'améliorer ces processus sans les comprendre et les visualiser. L'approche de la modélisation (*Process Modelling* en Anglais) vise à structurer et visualiser les processus afin d'assurer une cohérence globale des besoins métiers. Elle permet aussi, conjointement avec les techniques de traçabilité (*Traceability* en Anglais), de capturer et documenter les processus, notamment les chemins décisionnels pour avoir un suivi cohérent et gérer leurs mises à jour à différents niveaux.

La documentation des processus et traces des processus sont souvent sous la forme de modèles de processus et de données textuelles, respectivement. Il nous semble, donc, que les techniques de fouille de processus (*Process Mining* en Anglais) et de fouille de données (*Data Mining* en Anglais) permettent d'analyser et réutiliser les connaissances et les anciennes décisions tracées pour aider les ingénieurs à décider. Par

conséquent, nous formulons notre hypothèse de travail comme suit :



Hypothèse de recherche

Le couplage des techniques de **modélisation des processus**, de **traçabilité**, de **fouille de données** et de **fouille de processus** permet de répondre à notre question de recherche établie dans la Section 2.3.2.

2.3.4 Caractéristiques générales du système recherché

Pour répondre à nos critères, le système doit **impérativement** avoir la capacité de :

- **modéliser et visualiser** les processus et les activités, comme présenté dans la Section 2.3.2,
- **tracer** les informations relatives au QQQCOP (Qui a fait Quoi, Quand, Comment, Où et Pourquoi), comme présenté dans la Section 2.3.2,
- **aider à décider**, comme présenté dans la Section 2.3.2,
- **identifier et propager les changements contextuels**, comme présenté dans la Section 2.3.2.

En outre, le système doit **préférentiellement** avoir les caractéristiques suivantes (dont le classement n'a aucune importance) :

- **Modulaire d'architecture** : le système doit se composer de différents éléments, qui sont inter-connectés, dont chacun réalise un de nos objectifs (Figure 2.10). L'avantage de cette architecture est double. Premièrement, elle offre la possibilité de remplacer ou ajouter un composant sans impacter le reste du système. Deuxièmement, elle permet de réutiliser un ou plusieurs composants pour développer d'autres systèmes ou logiciels [Balci, 1998].

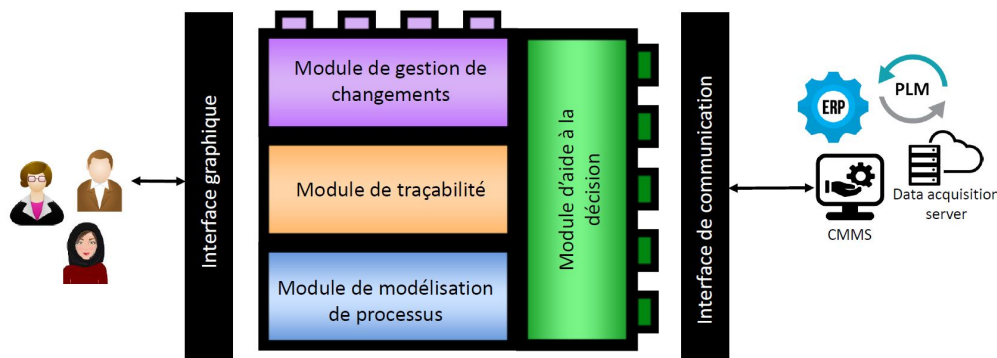


FIGURE 2.10 – Architecture modulaire du système recherché

- **Open source** : le code source du système doit être disponible gratuitement et doit pouvoir être modifié et/ou redistribué,
- **Générique** : le système doit être utilisable pour tous les scénarios possibles de notre contexte de recherche, à savoir, la conception et la supervision des systèmes,
- **Indépendant** : le système doit être indépendant des autres systèmes d'informations existants dans une organisation, tout en offrant la possibilité de communication avec eux (Figure 2.10),
- **Semi-Autonome/Intelligent** : le système doit être apte à modéliser, visualiser, tracer et proposer des décisions sans aucune intervention humaine. En revanche, les décisions ne doivent pas s'exécuter automatiquement sans être validées par l'utilisateur. Le système doit, ensuite, apprendre des décisions passées pour améliorer l'efficacité des décisions qui seront proposées dans le futur.

- **Capacité de support** : sans remplacer les décideurs, le système doit les supporter à n'importe quel niveau hiérarchique (concepteur, responsable de ventes, chef de département, etc.) et à n'importe quelle phase lors de la conception et supervision des systèmes.
- **Facile d'utilisation** : le système doit être facile à comprendre et doit avoir une interface graphique interactive, permettant un flux bidirectionnel d'informations entre le système et l'utilisateur,
- **Extensible** : le système doit permettre la modification et/ou l'ajout de nouvelles fonctionnalités,
- **Maintenable** : le système doit être flexible et doit permettre d'effectuer des changements pour répondre aux nouvelles exigences ou pour corriger les lacunes [Balci, 1998]. Cette caractéristique se voit très importante vu qu'elle représente presque 70% du coût du cycle de vie du logiciel d'après Schach [1999]. En outre, elle est impactée par la fiabilité et la complexité du code source du système [ISO/IEC, 2001],
- **Portable** : le système doit avoir la capacité d'être utilisé dans un environnement matériel dont la configuration est différente du sien, cette caractéristique est intéressante pour deux raisons. Selon Schach [1999], un bon logiciel doit avoir une durée de vie de 15 ans ou plus, tandis qu'un matériel est changé au moins chaque 4 ou 5 ans. Ainsi, un bon logiciel peut être implémenté, durant sa durée de vie, sur au moins trois configurations matérielles différentes. La deuxième raison est que c'est nettement moins cher de déployer un système existant dans un nouvel environnement que de développer un nouveau système en partant de rien.
- **Efficient** : c'est le degré avec lequel le logiciel remplit ses objectifs sans gaspillage de ressources [Balci, 1998]. Comme métriques de cette caractéristique nous citons, par exemple, l'espace disque/mémoire utilisé [ISO/IEC, 2001] et le temps pris pour proposer une décision.
- **Efficace** : c'est le pourcentage des décisions validées par rapport aux décisions proposées. Un système est dit efficace si le taux d'acceptation des décisions proposées dépasse un seuil défini par l'utilisateur.
- **Exact** : cette caractéristique mesure le pourcentage auquel le système adhère à ses exigences spécifiées [Balci, 1998]. Dans notre contexte, les exigences sont les quatre objectifs ou critères auxquels nous devons impérativement répondre.
- **Fiable** : cette caractéristique mesure la fréquence et la criticité de la défaillance du système, sachant que la défaillance est un comportement inacceptable du système [Balci, 1998; ISO/IEC, 2001].
- **Orienté activité et processus** : le système doit permettre de traiter les processus et les activités. Comme nous l'avons déjà défini, un processus est un ensemble d'activité inter-connectées, nous souhaitons donc visualiser les deux concepts, ainsi que tracer les informations qui leurs sont relatives. Nous souhaitons aussi que le système soit capable de détecter les changements qui surviennent au niveau des deux concepts et qu'il aide les ingénieurs à décider des choix de conception en termes de meilleurs processus et activités en tenant compte de ces changements contextuels.

État de l'art général : Travaux académiques et outils permettant l'aide à la décision

"Nothing is more difficult, and therefore more precious, than to be able to decide."

Napoleon Bonaparte

3.1	Introduction : la décision sous le microscope	29
3.2	État de l'art académique : travaux de recherche concernant la prise de décision	30
3.2.1	Problème de décision	31
3.2.2	Théorie de décision	32
3.2.3	Quelques approches de la prise de décision	34
3.2.4	Quelques méthodes et techniques de la prise de décision lors de la conception et supervision des systèmes	37
3.2.4.1	Méthodes dédiées à la conception des systèmes	37
3.2.4.2	Méthodes dédiées à la supervision des systèmes	40
3.2.5	Processus de prise de décision	42
3.2.6	Quelques systèmes intelligents supportant la prise de décision	44
3.2.6.1	Les systèmes experts	44
3.2.6.2	Les systèmes de recommandation	47
3.2.6.3	Les systèmes de support de prise de décision (DSS)	50
3.2.7	Synthèse et discussion de l'état de l'art académique	53
3.3	État de l'art commercial : outils supportant la prise de décision	55
3.3.1	Blueworks Live d'IBM	55
3.3.2	SAS Event Stream Processing	57
3.3.3	Autres outils supportant la prise de décision	58
3.3.4	Synthèse et discussion de l'état de l'art commercial	59
3.4	Synthèse et conclusion de l'état de l'art académique et commercial	60

La prise de décision est un concept très (très) ancien qui remonte à l'époque où l'être humain était guidé par les étoiles. Depuis ces jours, des mécanismes et outils ont été développés; allant des systèmes de numérotation et algèbre indo-arabe, à l'empirisme systématique d'Aristote²³, au raisonnement déductif de Platon²⁴, au raisonnement inductif de Francis Bacon²⁵, à l'application des théories scientifiques de

23. <https://fr.wikipedia.org/wiki/Aristote>

24. <https://fr.wikipedia.org/wiki/Platon>

25. [https://fr.wikipedia.org/wiki/Francis_Bacon_\(philosophe\)](https://fr.wikipedia.org/wiki/Francis_Bacon_(philosophe))

Descartes²⁶. Les progrès qu'a connus la technologie ainsi que la compréhension de plus en plus approfondie du comportement humain ont amélioré la prise de décision. Cette science qui va de la gestion d'une simple décision comme sortir ou pas avec les copains un soir, à la gestion d'une décision à gros risques comme chuter ou pas (et quand) d'un avion qui est en train de subir une explosion en plein air.

Dans ce deuxième chapitre, qui présente la première partie de l'état de l'art général sur la prise de décision, nous allons d'abord définir ce qu'est la notion de décision et les éléments qui la caractérisent, avant d'aller découvrir les travaux de recherche, les approches, les techniques, les méthodes et les outils proposés dans les milieux académiques pour soutenir la prise de la décision. Ensuite, nous présentons quelques outils, existants dans le marché, qui permettent de supporter l'utilisateur au cours de son processus décisionnel. Le chapitre est ensuite clôturé par une synthèse où les travaux étudiés sont discutés.

26. https://en.wikipedia.org/wiki/Ren%C3%A9_Descartes

3.1 Introduction : la décision sous le microscope

L'être humain est amené à prendre des décisions tous les jours, qu'il en soit conscient ou inconscient, à la fois dans sa vie personnelle et professionnelle. Dans les deux cas, une décision est définie comme étant un ou plusieurs choix qui aboutissent à un résultat. Dans une entreprise, une décision est présente durant tout le cycle de vie du produit et/ou service (le Tableau 3.1 présente quelques activités nécessitant la prise de décision lors de la conception d'un produit), et est définie comme étant une allocation de ressources irréversible qui dépend de la nature de l'activité. Plusieurs types de décision sont proposés dans la littérature dont neufs sont cités. [Katona \[1946\]](#) distingue deux types de décisions. Le premier type concerne les **décisions basiques** qui sont cruciales et nécessitent une délibération. Le deuxième concerne les **décisions de routine** qui se répètent fréquemment, nécessitant ainsi peu de considération. [Simon \[1960\]](#) distingue deux types de décisions. Le premier type concerne les **décisions programmées**, qui sont considérées comme des décisions de routine de nature répétitive. Le deuxième type concerne les **décisions non programmées** qui sont plus complexes et qui nécessitent une étude approfondie. [Stoner \[1961\]](#) distingue deux autres types de décisions. Les **décisions individuels** qui sont prises par une seule personne et qui sont de nature simple ou routinière, souvent supportées par des procédures ou des règles prédéterminées ne requérant, ainsi, pas des compétences particulières. L'autre type concerne les **décisions de groupe** qui sont prises par un groupe de deux ou plusieurs personnes, il s'agit des décisions complexes qui nécessitent d'être bien discutées avant d'être prises parce qu'elles ont des conséquences à moyen/long terme. [Anthony \[1965\]](#) et [Dale \[1973\]](#) distinguent trois types d'activités dans une entreprise donnant lieu à trois types de décisions :

- Les activités de planification stratégique : où les activités engendrent des décisions majeures ayant des conséquences à long terme. Ces **décisions stratégiques** concernent l'interaction de l'entreprise avec son environnement, comme par exemple la décision de développer un nouveau produit,
- Les activités de contrôle et planification managériaux : engendrant des décisions avec des conséquences à moyen terme. Ces **décisions tactiques**, appelées aussi administratives, sont relatives à la gestion effective des ressources,
- Les activités de contrôle opérationnel : elles concernent la gestion opérationnelle des ressources et la supervision des activités. Ces **décisions opérationnelles**, dont les conséquences sont à court terme, concernent les activités quotidiennes de l'entreprise, comme la supervision de la qualité des produits, entre autres.

Phase	Situation
Conception	Évaluer l'opportunité, Choisir la ou les technologies à utiliser, Chercher les ressources logicielles et matérielles qui peuvent être réutilisées, Étudier la faisabilité technique du concept en se basant sur les exigences sur le coût, le planning et la performance, Analyser les alternatives pour décider, Choisir quelles ressources physiques allouer pour les fonctions, Sélectionner l'architecture du système.
Développement	Sélectionner les éléments et sous-éléments du système, Sélectionner les méthodes de test et d'évaluation, Sélectionner les personnes qui devraient tester le système et les ressources à utiliser.
Production	Exécuter, faire ou acheter une décision, Sélectionner le processus et le lieu de production, Sélectionner le planning le plus rentable pour la production.
Support	Sélectionner une approche de maintenance
Fin de vie	Sélectionner une approche d'élimination ou de recyclage

TABLE 3.1 – Situations de décision durant le cycle de vie d'un produit (adapté de l'[INCOSE \[2015\]](#); [Buede \[2011\]](#))

Une décision, indépendamment de son type, a typiquement au moins une des caractéristiques suivantes [Dezfuli et al., 2010; Corner and Corner, 1995] :

- L'enjeu : comme le coût significatif, l'impact sur la sécurité, la priorité, etc.
- La complexité : cette caractéristique fait référence aux éléments de la décision et l'interaction avec son environnement. Les décisions complexes ne sont pas simples à faire et nécessitent une analyse plus ou moins approfondie,
- L'incertitude : contrairement aux décisions certaines, les décisions incertaines ont des conséquences dont les probabilités sont complètement inconnues, et nécessitent ainsi des approches particulières,
- Le risque : c'est l'ensemble des événements connus mais non désirés, auxquels la décision peut donner lieu.
- Le nombre des attributs de la décision,
- La diversité des décideurs et/ou partie prenantes : puisqu'il est nécessaire de clarifier les objectifs et formuler les performances attendues au cas où la décision est prise par plusieurs personnes pouvant avoir plusieurs perspectives et/ou des objectifs parfois contradictoires,

! Les notions de risque et d'incertitude sont différentes. En effet, dans un environnement incertain, la probabilité de l'événement, qu'il soit désiré ou non désiré, est inconnue et ne peut pas être mesurée ou estimée. Le risque est, quant à lui, décrit par le triplet (S_i, P_i, U_i) sachant que S_i est un scénario indésirable, P_i est la probabilité de S_i , et U_i mesure la sévérité qualitative ou quantitative de la dégradation de la performance qui pourrait résulter si ce scénario indésirable a lieu [Kaplan and Garrick, 1981]. En outre, le risque peut être positif ou négatif selon la façon comment il affecte le projet et/ou les objectifs [PMI, 2018].

3.2 État de l'art académique : travaux de recherche concernant la prise de décision

Afin de générer la ou les bonnes décisions possibles, il est important de considérer plusieurs éléments. Zachary et al. [1982] identifient trois composants essentiels de la prise de décision : la situation de décision, le décideur et le processus de décision. L'axiome de choix (appelé aussi axiome de sélection) [Lipschutz, 1998], affirme que les trois essences de la prise de décision sont : les objectifs de décision, un ensemble d'alternatives ou choix, et un ensemble de critère de sélection ou stratégies. Dans la suite de ce document, nous caractérisons la prise de décision par les quatre éléments suivants [Chai et al., 2013] :

- **Le décideur** : en spécifiant s'il s'agit d'une seule ou d'un groupe de personnes par lesquelles la décision sera prise,
- **L'environnement de décision** : en spécifiant son type, qui peut être soit certain (le décideur connaît avec certitude les conséquences de chaque décision ainsi que leurs probabilités), incertain, ou en situation de risque, comme déjà décrit,
- **Le problème de décision** : en identifiant sa nature comme présenté dans la Section 3.2.1,
- **La théorie, l'approche, les méthodes/les techniques/les outils et le processus de décision** : qui décrivent, ensemble, la façon comment le problème de décision va être abordé et analysé ainsi que les étapes à suivre pour prendre la bonne décision. Dans la littérature, nous rencontrons quelques concepts clés, qui sont sémantiquement différents mais parfois utilisés interchangeablement ou d'une façon contradictoire. Un exemple de ces concepts est : la théorie, l'approche, la stratégie, la méthode, la technique et le processus. Avant de détailler cet aspect et parcourir les travaux de recherche qui concernent la prise de décisions (Section 3.2), nous souhaitons présenter les définitions que nous gardons pour ces concepts (adaptées de McKechnie et al. [2005] et Mintzberg et al. [2005]) :

- Une **théorie** est un système d'hypothèses ou principes posés pour expliquer un problème.
- Une **approche** est la façon comment un problème est abordé ou abordé.
- Une **méthode** est l'ensemble des étapes suivies pour traiter un problème.
- Une **stratégie** est une méthode à laquelle s'ajoute la notion de planification. De plus, elle est utilisée pour les nouveaux problèmes.
- Une **technique** est l'application des compétences pour résoudre un problème.
- Un **processus** est un ensemble d'étapes bien définies, qui met en œuvre des méthodes et/ou techniques pour exécuter une tâche spécifique.

Par conséquent, pour identifier la technique de prise de décision la plus adaptée à un problème de décision donné, il faut d'abord identifier la nature du problème de décision, la théorie adoptée pour expliquer ce dernier et l'approche comment ce problème est abordé (Figure 3.1). Ensuite, il faut définir un processus de prise de décision, qui met en œuvre la technique choisie et dont l'exécution résout le problème de décision (Figure 3.1). Notons que l'environnement de décision dans notre contexte est supposé être certain, parce que nous nous basons sur les expériences passées dont les résultats sont connus.

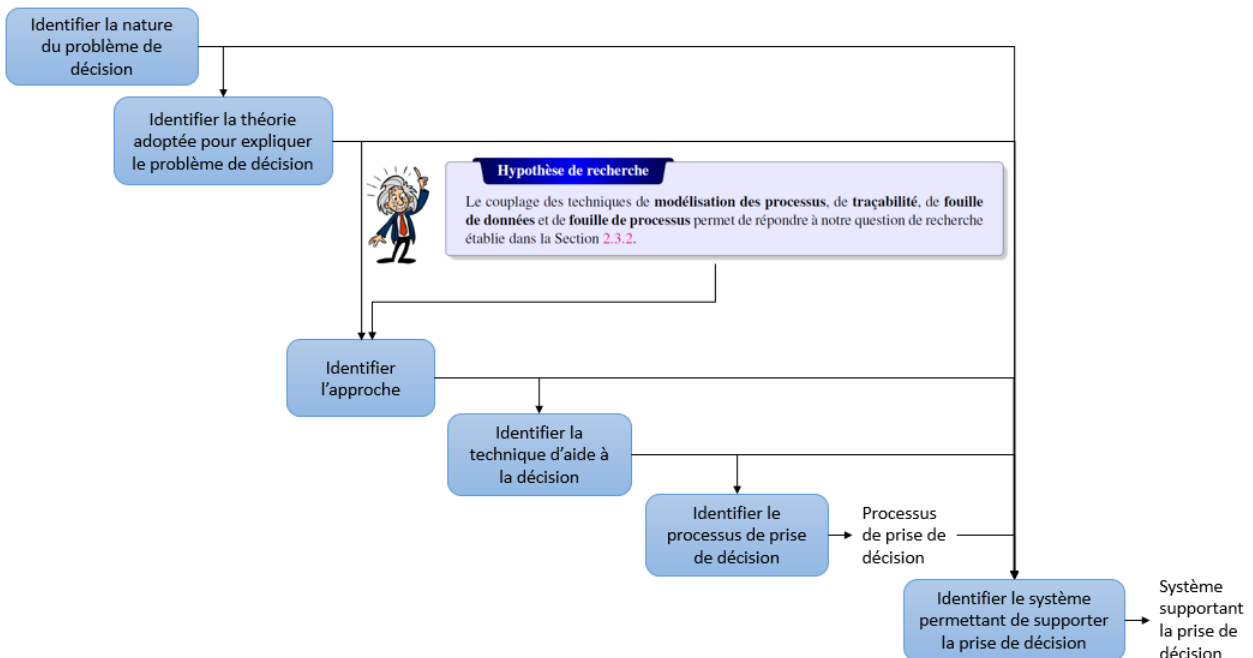


FIGURE 3.1 – Représentation IDEF0 des étapes de résolution d'un problème de décision donné.

3.2.1 Problème de décision

Selon Hammond et al. [1999], les managers ont souvent tendance à considérer peu d'alternatives quand ils prennent des décisions difficiles. Ceci est dû à une mal formulation du problème de décision ; ce qui a un effet significatif sur la qualité des décisions prises. Selon Chai et al. [2013], il existe trois grands types de problèmes de prise de décision :

- **Problèmes structurés** : c'est le type de problème qui admet un processus de prise de décision dont toutes les activités font appel à des procédures normalisées, et ayant des objectifs clairs et des entrées/sorties bien spécifiées. Dans cette catégorie, il existe des méthodes permettant d'obtenir la solution optimale.
- **Problèmes non structurés** : dans cette catégorie, toutes les activités du processus décisionnel ne sont pas structurées, et le jugement et/ou intuition humaine jouent un rôle très important lors de la prise de décision.

- **Problèmes semi structurés** : dans ce type de problèmes, le processus décisionnel comporte quelques activités structurées où les procédures normalisées peuvent être combinées au jugement et/ou intuition humaine.

En plus de ces trois types de problèmes, [Clark et al. \[2009\]](#) identifient d'autres critères qui leur sont associés (Tableau 3.3). En effet, il existe des problèmes de décision simples et d'autres complexes comme le montre la Figure 3.2.a. Dans le premier type, les décisions à prendre sont d'ordre simple (comme le choix entre préparer une tarte au chocolat, ou plutôt une tarte au yaourt) et le problème est souvent structuré (ajouter une quantité de chocolat en poudre aux ingrédients s'il s'agit d'une tarte au chocolat, sinon, ajouter plutôt une quantité de yaourt). Dans le deuxième type de problèmes, les décisions sont complexes et le problème a tendance d'être un problème de conception non ou semi structuré. Cette complexité de décision est, selon [Clark et al. \[2009\]](#), due à la (Figure 3.2.b) :

- Présence de plusieurs objectifs qui ne sont parfois pas bien définis et/ou contradictoires,
- Présence d'un nombre important de variables de décision dont chacune peut avoir plusieurs caractéristiques. Ceci donne lieu à plusieurs options de résolution de problème qui doivent être évaluées par le décideur afin de sélectionner la meilleure.
- Nature des conséquences des décisions qui sont parfois incertaines et/ou imprévisibles. Ceci pousse le décideur à évaluer la décision en se basant sur le maximum de scénarios possibles.

Les problèmes de décision sont aussi caractérisés par leur autonomie, en terme de dépendance ou pas à d'autres processus décisionnels, comme par exemple le processus ayant pour but le choix d'un matériau pour concevoir une voiture qui dépend bien du processus de choix de la technologie de fabrication. Une autre caractéristique du problème de décision est le nombre de décideurs, qui peut varier selon les domaines, la complexité des décisions à prendre, etc.

Paramètre	Valeur		
Degré de difficulté	Simple	Complexe	
Structure du problème	Structuré	Non structuré	
Nature du problème	Problème de choix	Problème de conception	
Lien avec les autres problèmes de décision	Indépendant	Dépendant d'autres processus décisionnels	
Type de décideur	Un seul décideur	Groupe de décideurs	
Nombre d'objectifs à réaliser	Un seul	Plusieurs	
Prédiction des conséquences du problème de décision	Possibilité de prévoir les conséquences avec certitude	Plusieurs conséquences possibles avec des probabilités d'occurrence prévisibles	Plusieurs conséquences possibles avec des probabilités d'occurrence non prévisibles

TABLE 3.3 – Dimensions et caractéristiques d'un problème de décision (adapté de [[Clark et al., 2009](#)])

3.2.2 Théorie de décision

La prise de décision est le processus dans lequel une option ou une suite d'actions parmi plusieurs autres sont choisies selon quelques critères ou stratégies bien définies [[Edwards, 1954](#); [Wilson and Keil, 2001](#); [Wang and Ruhe, 2007](#); [Wang, 2008](#)]. [Stanovich \[2010\]](#) voit que "prendre de bonnes décisions" revient à être "instrumentalement rationnel", c'est à dire se concentrer sur le moyen le plus efficace ou rentable

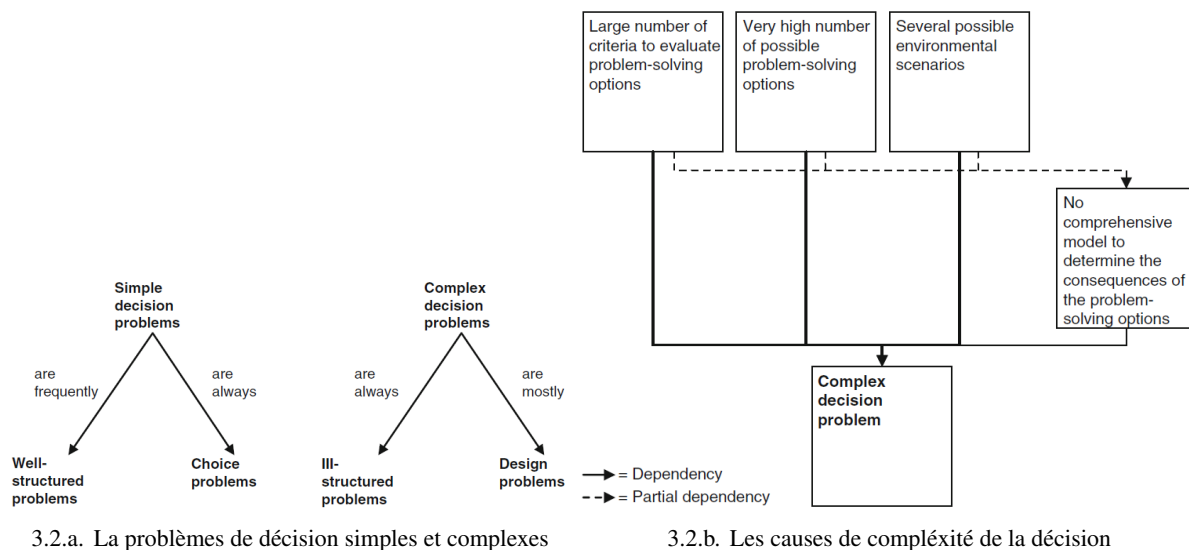


FIGURE 3.2 – Les caractéristiques des décisions simples et complexes [Clark et al., 2009]

pour réaliser son objectif. Cependant, la rationalité totale des décisions, qui voudrait que la décision soit faite en étudiant toutes les informations disponibles, n'est quasiment pas possible comme l'a bien souligné Simon [1967, 1990], lauréat du prix Nobel en économie, en introduisant le concept de la rationalité limitée. Ce dernier suppose que le décideur sélectionne le choix satisfaisant, et non pas optimal, en se basant sur les informations disponibles et ses jugements et émotions. En effet, parfois nous laissons le choix à notre "système I" - émotion - quand nous faisons face à un problème de décision, comme a décrit Kahneman [2011], lauréat du prix Nobel en économie. Prenons par exemple le processus d'achat d'une voiture, le choix final dépend de plusieurs paramètres comme le type de la voiture, sa vitesse maximale, le revenu du décideur, etc. Entre ces deux voitures de luxe : "Land Rover Range Rover Evoque black matte" et "BMW X6 jet black", que nous supposons avoir le même prix et mêmes caractéristiques techniques, une personne "W" peut décider d'acheter la première voiture parce que sa couleur "black matte" est plus mystérieuse et sa silhouette est plus agressive que celle de la deuxième voiture. Il existe, en effet, plusieurs façons de raisonner quant au choix d'une option [Clark et al., 2009] :

- Raisonnement intuitif ou jugement subjectif,
- Raisonnement par un recours systématique aux décisions et procédures utilisées dans le passé,
- Raisonnement par l'adoption des solutions suggérées par les experts,
- Choix aléatoire sans aucun raisonnement,
- Choix en se basant sur un raisonnement rationnel admettant des preuves pertinentes.

Ces raisonnements, parfois subjectifs, qui mènent à la décision finale, sont étudiés par ce qu'on appelle la *théorie de décision*.

La théorie de décision est une science interdisciplinaire, définie par Hansson [2005] comme étant la science qui concerne les activités orientées vers un objectif et dans lesquelles le choix de l'option est fait d'une manière non aléatoire, en étant le résultat d'un raisonnement rationnel [Bazerman and Watkins, 2004]. Par ailleurs, elle est définie par Parmigiani [2001] comme étant la science qui étudie la logique et les propriétés mathématiques de la prise de décision dans l'incertitude (*decision making under uncertainty* en Anglais). Cette science n'a par conséquent pas de définition unique parce que plusieurs chercheurs (comme le politicien qui étudie les règles de vote, le psychologue qui étudie le comportement des individus lors de la prise des décisions, le philosophe qui étudie les exigences des décisions rationnelles, etc.) appartenant à différentes disciplines y contribuent [Hansson, 2005]. Ceci donne lieu à plusieurs façons de théorisation de la décision, qui malgré leur diversité, sont divisées en trois grandes théories : normative, descriptive et prescriptive [Takemura, 2014; Clark et al., 2009] :

- La **théorie normative** vise à exprimer, d'une façon bien définie, comment les décideurs (qui sont supposés être rationnels) doivent réagir lorsqu'ils prennent des décisions risquées [Suhonen et al., 2007]. Les exemples les plus populaires de l'approche nominative largement utilisés en économie, sont la théorie de l'utilité (*utility theory* en Anglais) qui vise à mesurer les préférences sur certains produits ou services [Fishburn, 1970], et la théorie de SEUT (Subjective Expected Utility) dans laquelle un décideur doit maximiser le profit en choisissant la meilleure opportunité sous la présence de risque [Barbera et al., 1998; Fishburn, 1981].
- La **théorie descriptive** vise à décrire comment les décisions (rationnelles ou irrationnelles) sont réellement faites, elle est basée sur les observations empiriques et les études expérimentales du comportement de choix. Nous citons comme exemple, la théorie de décision comportementale [Edwards, 1961] qui inclue la théorie des perspectives²⁷ (Prospect Theory) [Kahneman and Tversky, 2013] et la théorie de regret²⁸ (Regret Theory) [Loomes and Sugden, 1982].
- La **théorie prescriptive** vise à supporter la prise de décision selon le contexte (conditions) du problème [Bell et al., 1988] en présentant un ou plusieurs choix d'aide pratique aux décideurs qui ne sont pas suffisamment rationnels [Suhonen et al., 2007]. Un exemple d'approches de cette théorie est la recherche opérationnelle [Pokrovsky, 2009a] et la science de gestion.

Bouyssou [2000], Belton and Stewart [2002] et Genard and Pirlot [2002] ajoutent une quatrième théorie *constructive*, dans laquelle le problème et sa solution sont construits en même temps. En effet, dans cette théorie, le décideur commence d'abord par formuler un problème formel avec le consensus du client et ensuite il aborde la construction du modèle de prise de décision approprié.

3.2.3 Quelques approches de la prise de décision

La prise de décision est un sujet de recherche auquel s'intéressent plusieurs disciplines comme l'informatique, l'informatique cognitive, la gestion, l'économie, la sociologie, la science de décision, la médecine, etc. [Wald, 1992; Berger, 2013; Pinel and Barnes, 2017; Matlin and Farmer, 2017; Edwards and Fasolo, 2001; Hastie, 2001; Wilson and Keil, 2001]. Il s'agit, selon Wang [2008], d'un processus cognitif faisant partie des 37 processus cognitifs fondamentaux qui sont modélisés dans le modèle LRMB (Layered Reference Model of the Brain) où chacune des disciplines explore l'aspect de la prise de décision connecté à un problème particulier qui lui est lié [Wang et al., 2006; Wang, 2008]. Cette présence de plusieurs disciplines ainsi que la différence dans la nature des problèmes de décision (structuré, non structuré, semi structuré), de l'environnement de décision (certain, incertain, sous risque) et le décideur (personnalité, situation organisationnelle, etc.), sont à l'origine de la grande variation de la façon dont la prise de décision est conceptualisée. De ce fait, plusieurs approches de décisions ont été proposées dans la littérature, qui malgré leur diversité, prennent généralement une (ou plusieurs) des formes suivantes :

- **Approches rationnelles** [Doyle, 1999] : ce sont des approches structurées et séquentielles, ayant pour but de chercher la solution précise à un problème bien défini en utilisant des données obtenues "scientifiquement" par le biais de l'observation, les analyses statistiques, la modélisation, etc. Ces approches réduisent, entre autres, les chances des erreurs et la subjectivité.
- **Approches canoniques** (ou normatives) [Roy, 2016] : dans ce types d'approches, les décideurs prennent leurs décisions d'une façon logique et rationnelle en suivant une norme. Les standards ou principes de cette approche sont fournis par un modèle normatif qui montre la façon correcte ou normale de la prise de décision.
- **Approches informatives** [Roy, 2016] : elles s'opposent aux approches canoniques en considérant les données de type descriptives, comme les recommandations, l'historique, les informations supplémentaires, etc.

27. C'est une théorie qui montre comment les décideurs choisissent les alternatives qui impliquent à la fois le risque et l'incertitude.

28. C'est une théorie qui incorpore le regret, qui pourrait se produire en prenant une mauvaise décision, dans la phase de choix de l'alternative pour l'éliminer ou au moins réduire sa probabilité.

- **Approches axiomatiques** (ou déterministes) [Roberts, 1979] : elles formulent et caractérisent les principes fondamentaux de la prise de décision humaines en termes d'axiomes déterministes qui ne tiennent pas compte des variations stochastiques qui affectent les données.
- **Approches bayésiennes** [Wald, 1950] : ces approches permettent de prédire le résultat d'un événement futur sachant les événements passés. En d'autres mots, elles permettent de trouver les solutions possibles d'un problème de décision, en soulignant la solution la plus probable.
- **Approches stochastiques** [Carbone and Hey, 2000] : ces approches s'intéressent à l'étude des problèmes de décision caractérisés par des facteurs aléatoires comme le comportement aléatoire lors du choix d'une solution parmi d'autres.
- **Approches basées sur la théorie de la différence d'information** (Information-Gap Decision Theory) [Ben-Haim, 2001; Duncan et al., 2008] : ces approches, non probabilistes, mesurent la différence entre un paramètre et sa valeur estimée. Elles ont pour but d'optimiser la robustesse de la décision dans un environnement caractérisé par une incertitude sévère.
- **Approches basées sur la théorie des jeux** (game theory) [Lewis and Mistree, 1998] : ce sont des approches destinées à la prise de décisions logiques dans un environnement compétitif. Dans le contexte de la conception des produits, le processus de prise de décision est considéré comme un jeu qui inclue plusieurs équipes appelées joueurs dont chacune contrôle un sous ensemble de variables de conception en cherchant à minimiser sa fonction de coût. Ainsi, les joueurs coopèrent pour réaliser un ensemble d'objectifs globaux.
- **Approche à un seul niveau** (Single-level approaches) [Sobieszcanski-Sobieski and Haftka, 1997] : dans ces approches, toutes les responsabilités de prise de décision sont confiées à une seule équipe, généralement l'équipe des concepteurs dans le cas de la conception de produit. Cette approche a été critiquée par plusieurs travaux dont les plus remarquables sont les travaux de Balling and Sobieszcanski-Sobieski [1996] et Sobieszcanski-Sobieski and Haftka [1997]. Ces critiques ont donné lieu aux approches à plusieurs niveaux.
- **Approches à plusieurs niveaux** (Multi-level approaches) [Zhang et al., 2015] : ce type d'approches aborde les problèmes qui exigent un compromis entre les objectifs de deux ou plusieurs entités en interaction, qui sont organisées dans une structure hiérarchique et ayant des objectifs indépendants, parfois contradictoires.
- **Approches dynamiques** (Dynamic Decision-Making) [Edwards, 1962] : ces approches concernent les problèmes de décision évolutifs et dynamiques, dans lesquels le décideur prend une séquence de décisions, dont la $décision_n$ dépend des conséquences de la $décision_{(n-1)}$.
- **Approches naturalistes** (Naturalistic Decision Making -NDM) [Zsombok and Klein, 2014] : ce sont des approches qui étudient comment les gens prennent les décisions et exécutent des tâches complexes dans des situations réelles, caractérisées par un temps limité, un environnement dynamique, une incertitude, des contraintes organisationnels, etc. Le modèle RPD (Recognition-Primed Decision Making) [Zsombok and Klein, 2014], issu de l'approche NDM, décrit comment les gens utilisent leur expérience sous forme de patrons (patterns), permettant aux décideurs de comparer le problème de décision actuel avec ces patrons pour prendre la bonne décision.
- **Approches basées sur la logique floue** [Zimmermann, 1986; Zadeh, 1975] : l'environnement de décision ainsi que les sources de données ont souvent plusieurs facteurs d'incertitude. De plus, les données peuvent aussi être incertaines, comme par exemple le jugement humain d'une alternative exprimée par des termes linguistiques "bonne" et "mauvaise" qui sont des expressions incertaines. Les approches mathématiques et précises ne sont pas efficaces face à ce genre d'incertitude. Par conséquent plusieurs techniques de traitement de l'information incertaines ont été développées en utilisant les approches basées sur les ensembles flous, les nombres flous et la logique floue; permettant de traiter ce genre d'informations "floues".
- **Approches basées sur la recherche opérationnelle** (Operations research) [Pokrovsky, 2009b] : ce sont des approches systématiques qui utilisent des techniques mathématiques pour trouver la solution optimale ou presque optimale pour un problème de décision complexe. De cette approche déroule

l'approche multi-critères qui concerne les problèmes de décision en présence de critères multiples et contradictoires (ou en conflit) par rapport aux alternatives disponibles [Mardani et al., 2015a].

Pour classer les approches de prise de décision, Wang and Ruhe [2007] adoptent la théorie des choix (appelée également axiome de sélection ou encore axiome de choix), développée par Lipschutz [1998], qui considère que les éléments, les plus essentiels, de la prise de décisions sont au nombre de trois : les objectifs de décision, l'ensemble d'alternatives ou choix, et l'ensemble des critères de choix (ou de sélection). De cette théorie, Wang and Ruhe [2007] dégagent cinq catégories d'approches de la prise de décision (Tableau 3.4). Les deux catégories intuitive et empirique sont en phase avec les capacités cognitives humaines et n'adoptent donc pas de modèle rationnel expliquant les critères de choix. La prise de décision rationnelle est décrite par les deux catégories statique et dynamique qui favorisent la logique, les analyses et l'objectivité lors de la prise de décision. La cinquième catégorie heuristique s'avère très pratique lorsque le décideur ne dispose pas des informations suffisantes pour mener l'analyse. Une heuristique est définie par Gigerenzer and Gaissmaier [2011] comme étant une technique cognitive efficace qui tient compte à chaque pas, lors de la résolution de problème, des résultats précédents et en déduit la stratégie à adopter par la suite. Elle permet, selon Shah and Oppenheimer [2008], de simplifier le processus de prise de décision en diminuant, entre autres, le travail de récupération et stockage des informations nécessaires pour faire le choix.

Catégorie	Stratégie	Critère de choix
Intuitive	Arbitraire	Basée sur le choix le plus familier
	Préférence	Basée sur les attentes et les préférences
	Bon sens	Basée sur les axiomes et le jugement
Empirique	Essai et erreurs	Basée sur des essais exhaustifs
	Expérimentation	Basée sur les résultats de l'expérience
	Expérience	Basée sur les connaissances existantes
	Consultation	Basée sur des consultations professionnelles
	Estimation	Basée sur une évaluation approximative
Heuristique	Principes	Basée sur des théories scientifiques
	Éthiques	Basée sur le jugement philosophique et la croyance
	Représentative	Basée sur des règles communes
	Disponibilité	Basée sur des informations limitées
Rationnelle statique	Ancrage	Basée sur la présomption et sa justification
	Coût minimal	Basée sur la minimisation de l'énergie, du temps, de l'argent
	Bénéfice maximal	Basée sur la maximisation du gain, de fonctionnalité, de fiabilité et de qualité
	Utilité maximale certaine	Basée sur le rapport coût-bénéfice et sur la probabilité maximale et les données statistiques
	Utilité maximale incertaine optimiste (pessimiste)	Basée sur le maximax (maximin)
Rationnelle dynamique	Utilité maximale sous risque	Basée sur le rapport coût-bénéfice et sur la perte ou le regret minimaux
	Événement interactif	Basée sur des automates
	Jeux grille de décision	Basée sur les conflits Basée sur une série de choix dans une grille de décision

TABLE 3.4 – Catégorie d'approches de prise de décision selon Wang and Ruhe [2007]

Plusieurs autres approches sont récemment développées pour la prise de décision, quelques unes se basent sur une des approches listées ci-haut, tandis que d'autres combinent plusieurs approches. Nous citons par exemple l'approche "Fuzzy multi-criteria decision making", développée par [Mardani et al. \[2015b\]](#), qui couple l'approche basée sur la logique floue et l'approche multi-critères.

Une fois le décideur a étudié et choisi la ou les approches qui lui permettront de mieux aborder son problème de décision, il est temps qu'il sélectionne les méthodes et/ou techniques à utiliser ainsi que le processus à suivre, qui quand il est exécuté, génère la ou les décisions à prendre en compte. Vu la diversité des approches et des problèmes de décision, plusieurs méthodes, techniques et processus sont proposés dans la littérature ; nous souhaitons par la suite lister, brièvement, ceux liés aux processus de conception et de supervision des systèmes, qui délimitent l'horizon de ce présent travail de recherche.

3.2.4 Quelques méthodes et techniques de la prise de décision lors de la conception et supervision des systèmes

3.2.4.1 Méthodes dédiées à la conception des systèmes

Dans la conception des produits, la combinaison des outils/des techniques/des méthodes est un moyen pour exploiter les compétences des concepteurs et des équipes de conception afin d'examiner un chemin possible [[Lutters et al., 2014](#)]. Plusieurs travaux de recherche ont étudié et discuté l'application des différentes méthodes et techniques de la prise de décision dans la conception et/ou supervision des systèmes. En outre, plusieurs autres travaux ont introduit de nouvelles méthodes/techniques regroupant plusieurs techniques de prise de décision existantes. [Krishnamurty \[2006\]](#) a étudié l'application de la théorie de l'utilité espérée (expected utility theory), qui a pour objectif de déterminer l'alternative de conception qui a une valeur utile maximale. Il a aussi étudié la méthode de l'évaluation des préférences en se basant sur la théorie des valeurs (Value Theory Based Preference Assessment) qui se voit comme une théorie déterministe qui permet d'exprimer quantitativement le degré des préférences des alternatives de conception et détermine ainsi ce qui est désirable pour un processus de conception donné. [Aruldoss et al. \[2013\]](#) et [Zavadskas et al. \[2014\]](#) ont revu et discuté les techniques/méthodes/modèles de la prise de décision multi-critères (MCDM). Cette dernière permet de choisir la meilleure solution en évaluant les critères qui sont souvent contradictoires comme le coût et la qualité. [Mardani et al. \[2015c\]](#) ont revu les techniques de prise de décision multi-critères floue (FMCDM) qui est une extension de la MCDM, permettant de donner de meilleurs résultats lorsque les données existantes, concernant les solutions, sont vagues ou incomplètes [[Kahraman et al., 2015](#)]. [Mardani et al. \[2016\]](#) ont étudié l'application de la méthode VIKOR (VIseKriterijumska Optimizacija I Kompromisno Resenje, en Anglais : Multicriteria Optimization and Compromise Solution) dans plusieurs domaines, la conception de produit y comprise. [Renzi et al. \[2017\]](#) ont étudié les méthodes de prise de décision utilisées lors des différentes étapes de la conception dans l'industrie automobile, en spécifiant pour chaque phase la ou les méthodes/techniques/outils utilisés.

La diversité des méthodes de conception a poussé quelques travaux à proposer des classifications, qui varient selon la perspective considérée par les auteurs, permettant de regrouper ces méthodes dans des groupes homogènes facilitant ainsi la compréhension et la quête de la bonne méthode. Une des plus anciennes classifications est celle de [Finger and Dixon \[1989a,b\]](#), elle se base sur la théorie et méthodologie de conception (Design Theory and Methodology - DTM) qui se voit comme un domaine riche des résultats des études sur comment les ingénieurs conçoivent leurs produits en matière de processus et activités [[Tomiyama et al., 2009](#)]. [Finger and Dixon \[1989a,b\]](#) distinguent six classes de méthodes/techniques/outils de prise de décision :

- **Modèles descriptifs** : comme le modèle qui se base sur les données empiriques, développé par [Ullman et al. \[1988\]](#), les modèles cognitifs [[John S. Gero and Coyne, 1987](#)] et les études de cas [[Hales, 1987](#)] entre autres.
- **Modèles prescriptifs** : comme le processus de conception canonique [[French, 1971](#)], l'analyse morphologique [[Pahl et al., 2007](#)], GDT (General Design Theory)[[Yoshikawa, 1981](#)], l'Axiomatic Desin (AD) de [Suh \[2001\]](#) et la méthode de [Taguchi \[1987\]](#).

- **Modèles basés sur l'ordinateur** : comme la conception paramétrique [Monedero, 2000], les méthodes basées sur l'intelligence artificielle [Sriram, 1997; Sriram and Adey, 1987; Gero, 1985] et la conception basée sur les agents distribués [Cutkosky et al., 1993].
- **Langages et représentations pour la conception** : comme la modélisation du comportement et de la fonction [Umeda and Tomiyama, 1997], la modélisation basée sur les caractéristiques (feature based modelling) [Dong et al., 1991] et la modélisation du produit [Krause et al., 1993].
- **Techniques analytiques** : comme les méthodes d'optimisation [Roy et al., 2008], et l'aide à la décision.
- **Conception axée sur la fabrication** : comme l'ingénierie concurrente [Sohlenius, 1992], DFX (Design For X) [Huang, 1996] et l'ingénierie de cycle de vie [Alting, 1995], entre autres.

Renzi et al. [2017] proposent une classification qui se base sur la nature des problèmes de décision (problèmes multi-critères, problèmes non ou semi structurés ou complètement structurés comme discuté dans la Section 3.2.1) et distinguent ainsi trois grandes classes :

- **Méthodes de prise de décision multi-critères** (MCDM) : cette classe se compose de deux sous-classes [Belton and Stewart, 2002]. La première est la prise de décision multi-attributs (Multi Attribute Decision-Making MADM) qui traite les problèmes ayant un nombre fini d'alternatives et un espace de solution discret. La deuxième classe est la prise de décision multi-objectif (Multi Objective Decision-Making - MODM), elle traite les problèmes de décision ayant un nombre infini de solutions dans un espace continu.
- **Méthodes de structuration des problèmes** (Problem Structuring Methods - PSM) : cette classe traite les problèmes non ou semi structurés dans un contexte incertain [Rosenhead, 2006]. Ce type de problème est, selon Mingers and Rosenhead [2001], caractérisé par la présence de plusieurs acteurs, de plusieurs perspectives, des intérêts conflictuels, et par la présence d'une grande incertitude. Dans ce cas, les méthodes de PSM permettent de représenter la situation sous la forme de modèles pour permettre aux participants de comprendre la situation et de converger vers un problème défini pouvant potentiellement donner lieu à une action.
- **Méthodes de résolution de problèmes** (Decision-Making Problem solving - DPS) : cette classe adresse les problèmes relatifs aux premières étapes de conceptions et se divise en trois sous-classes. La première concerne la transformation des exigences des clients en spécifications des produits à concevoir, elle contient les méthodes QFD (Quality Function Deployment) [Ginn and Zairi, 2005] et AD (Axiomatic Design) [Suh, 2001]. La deuxième sous-classe concerne la résolution des problèmes et/ou la génération des alternatives et contient, entre autres, la méthode TRIZ [Altshuller, 1999]. La troisième sous-classe concerne la sélection (et parfois la génération) des alternatives de conception et contient, entre autres, la méthode de PuCC (Pugh's Controlled Convergence) [Pugh, 1981].

Chai et al. [2013] proposent une classification basée sur la nature de l'algorithme utilisé pour prendre la décision :

- **Méthodes multi-critères** : détaillées précédemment.
- **Méthodes de programmation mathématique** (Mathematical Programming - MP) : cette catégorie fait référence aux méthodes analytiques algébriques classiques, qui permettent de prescrire la meilleure façon pour atteindre un objectif donné tout en respectant un ensemble de contraintes. Les modèles MP sont particulièrement utiles lors de la prise des décisions tactiques qui concernent, entre autres, l'allocation des ressources à moyen terme (comme discuté dans la Section 3.1). Cette sous-classe contient les méthodes telles que DEA (Data Envelopment Analysis) [Cooper et al., 2004], LP (Linear Programming) [Li et al., 2010], IP (Integer Programming) [Schrijver, 1998], NLP (Non Linear Programming) [Li, 2011], MOP (Multi-Objective Programming) [Cohon, 2004], GP (Goal Programming) [Tamiz et al., 1998] et la méthode SP (Stochastic Programming) [Birge and Louveaux, 2011].
- **Méthodes de l'intelligence artificielle** (Artificial Intelligence - AI) : cette catégorie contient les

méthodes qui permettent de reproduire les fonctions cognitives de l'être humain en ayant la capacité d'apprendre et de résoudre les problèmes. Comme exemple de ces méthodes, nous citons le GA (Genetic Algorithm) [Goldberg, 1989], GST (Grey System Theory) [Liu et al., 2012], NN (Neural Networks) [Hill et al., 1994], RST (Rough Set Theory) [PAWLAK, 1998], BN (Bayesian Networks) [Jensen, 1996], DT (Decision Tree) [Z and Lior, 2014], CBR (Case-Based Reasoning) [Kolodner, 2014], PSO (Particle Swarm Optimization) [Clerc, 2010], SVM (Support Vector Machine) [Steinwart and Christmann, 2008] et la méthode AR (Association Rule) [Hipp et al., 2000].

Tomiya et al. [2009] proposent une classification basée sur la notion de GDT (General Design Theory) [Reich, 1995], qui se voit comme une théorie abstraite permettant d'expliquer comment la conception est conceptuellement exécutée, à l'aide de la manipulation des connaissances en se basant sur la théorie axiomatique des ensembles (Axiomatic Set Theory) [Suppes, 2012]. Trois classes sont proposées :

- Techniques pour **générer une nouvelle solution de conception** : cette classe contient trois sous-classes qui dépendent de l'objectif de la conception. La première sous-classe est la conception basée sur la créativité, elle contient les méthodes telles que le GA (Genetic Algorithm) [Goldberg, 1989], méthodes de simulation, brainstorming [Putman and Paulus, 2009], etc. La deuxième sous-classe concerne la conception basée sur la combinaison qui contient les méthodes appartenant à l'approche systématique de Pahl and Beitz [2013]. La dernière sous-classe concerne la conception basée sur la modification, elle contient les méthodes telles que la conception paramétrique, CBR [Kolodner, 2014], TRIZ [Altshuller, 1999], etc.
- Techniques pour **enrichir les informations concernant les solutions de conception** : comme l'Axiomatic Design (AD) [Suh, 2001], QFD [Ginn and Zairi, 2005], FMEA (Failure Mode and Effects Analysis) [Stamatis, 2003].
- Technique pour **gérer la conception et représenter ses connaissances** : comme la modélisation des connaissances de conception [Wielinga et al., 1993], ingénierie concurrente [Hartley, 1990], DSM (Design Structure Matrix) [Eppinger and Browning, 2012], etc.

Lutters et al. [2014] proposent une classification des méthodes de prise de décision qui se base sur l'étude de leurs caractéristiques selon plusieurs axes, comme le positionnement dans le cycle de conception, le type du résultat souhaité, le rôle de l'utilisateur, les objectifs, etc.

En plus des classifications présentées, d'autres travaux ont développé de nouvelles méthodes se basant sur une ou plusieurs méthodes existantes. Parmi ces travaux, nous citons deux que nous supposons être les plus intéressants. Le premier travail est celui de Kumar and Tandon [2015] qui a discuté l'imprécision et l'incertitude qui régissent la conception de produit. Il a introduit une nouvelle technique permettant de prendre des décisions créatives, faisables et valides quand les données d'entrée du processus de conception sont imprécises ou contradictoires. Le deuxième travail concerne celui de Montagna [2011] qui s'est intéressé au processus de développement de nouveaux produits (New Product Development - NPD) qui contient, entre autres, le processus de conception. Montagna [2011] a introduit une approche de prise de décision hybride qui permet d'analyser les décisions tout au long du NPD en mettant en place différentes méthodes et outils dérivés de trois domaines (gestion du cycle de vie produit - PLM, science de gestion/recherche opérationnelle - OR, gestion de connaissances - KM), en un seul framework intégré. L'approche hybride se compose de quatre étapes, devant être exécutées par un analyste, comme le montre la Figure 3.3 :

- **Étape 1** : cette étape comprends trois sous-étapes. La première est l'identification des opportunités ou situations de décision au cours du processus de NPD. La deuxième est la représentation des acteurs et de l'environnement dans lequel la décision doit être prise. La troisième étape est la description de la nature des contraintes et du processus opérationnel suivi.
- **Étape 2** : cette étape concerne l'identification des processus de décision au sein du NPD. Elle utilise le modèle de Mintzberg et al. [1976] comme référence pour définir la typologie des processus de décision au niveau de chaque opportunité de décision.

- **Étape 3** : cette étape concerne l'identification des typologies des processus d'aide à la décision dont la nature, l'objectif et les activités et/ou procédures sont déterminés par les caractéristiques du problème de décision donné.
- **Étape 4** : dans cette étape sont intégrés les différents outils dont chacun supporte une action technique spécifique.

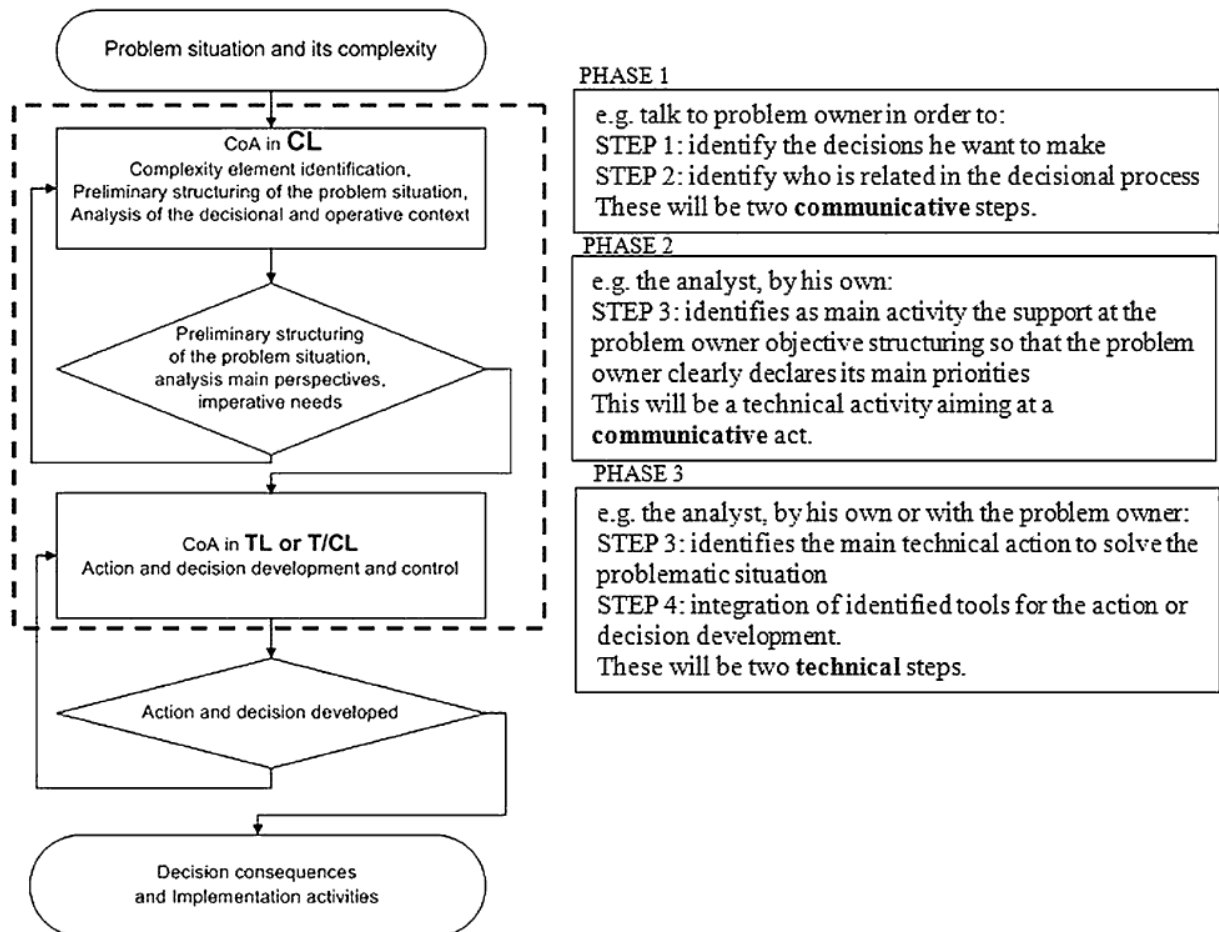


FIGURE 3.3 – Étapes de l'approche hybride pour la prise de décisions [Montagna, 2011]

Nous venons de découvrir les méthodes/techniques/outils de prise de décision lors de la conception des systèmes, étant la première tranche de notre contexte de travail. Dans la sous-section suivante, nous allons découvrir les méthodes/techniques/outils utilisés lors de la supervision des systèmes.

3.2.4.2 Méthodes dédiées à la supervision des systèmes

La supervision des systèmes ou des processus industriels, tout comme leur conception, joue un rôle décisif puisqu'elle affecte le système ou processus, positivement ou négativement, selon l'exactitude des décisions prises. Plusieurs chercheurs se sont intéressés au processus de la supervision et ont proposé des méthodes/outils/techniques permettant au superviseur (ou opérateur) de savoir ce qui est en train de se passer et qu'est ce qu'il doit faire pour éliminer les anomalies et amener le système à l'état stationnaire afin de garder le fonctionnement normal. Avant d'exposer quelques uns de ces travaux, nous souhaitons reprendre la définition de quelques mots-clés du processus de supervision [Cuyvers et al., 1990; Isermann and Ballé, 1996; Sohlberg, 1998] :

- **Surveillance** (Monitoring) : c'est une tâche continue en temps réel, ayant pour but de déterminer les conditions d'un système physique en enregistrant les informations permettant de reconnaître les erreurs et/ou les anomalies de comportement du système.
- **Supervision** : c'est la surveillance d'un système physique en prenant les mesures appropriées (décisions) pour maintenir le fonctionnement normal en cas de mal fonctionnement ou d'anomalie provenant d'une perturbation interne ou externe du système pour minimiser le risque de défaillance.
- **Anomalie** (Fault) : c'est une déviation non autorisée d'au moins d'une caractéristique, propriété ou variable du système.
- **Défaillance** (Failure) : c'est l'incapacité permanente du système à effectuer une fonction requise sous des conditions de fonctionnement spécifiques.
- **Mal fonctionnement** (Malfunction) : c'est une irrégularité intermittente d'un système dans l'accomplissement d'une fonction souhaitée.
- **Erreur** : c'est la déviation entre une valeur mesurée ou calculée d'une variable de sortie et sa valeur spécifiée ou considérée théoriquement correcte.
- **Perturbation** : c'est une entrée agissant sur le système et qui donne lieu à un départ temporaire de l'état stationnaire.

Selon [Haber et al. \[2002\]](#), les tâches de base pouvant être accomplies au niveau du processus de la supervision sont décrites par la boucle : surveillance des paramètres estimés, détection du degré de correspondance entre les valeurs mesurées et celles estimées et prise de décision. [Sohlberg \[1998\]](#) distingue trois tâches dans le processus de supervision, dont chacune fait appel à des méthodes de prise de décision différentes :

- **Détecter l'anomalie** (Fault detection) : il s'agit ici de détecter la présence d'une anomalie. Cette tâche est souvent probabiliste puisqu'elle se base sur les données ou mesures passées, qui sont sujets parfois à des perturbations, ce qui influence l'évaluation de l'anomalie (vraie ou fausse). [Miljković \[2011\]](#) expose les méthodes de prise de décision utilisées dans cette tâche.
- **Diagnostiquer l'anomalie** (Fault diagnosis) : dans cette tâche sont déterminés la localisation de l'anomalie au sein du processus supervisé et sa grandeur (anomalie affectant un composant du processus, plusieurs, etc.). Un exemple de méthodes utilisées dans cette tâche est les méthodes basées sur les modèles [[Ding, 2008](#)].
- **Identifier l'anomalie** (Fault identification) : dans cette tâche sont prises les décisions à la suite du diagnostic de l'anomalie, comme continuer l'exécution du processus industriel malgré la présence de l'anomalie, planification d'une maintenance, déclencher l'exécution d'un autre processus permettant de corriger cette anomalie, etc. Un exemple de méthodes utilisées dans cette tâche est les méthodes basées sur la reconstruction [[Yue and Qin, 2001](#)].

[Willsky \[1976\]](#) a présenté les méthodes de détection des anomalies dans le contexte des systèmes linéaires²⁹ et a discuté les avantages et inconvénients de chacune. [Isermann \[1984\]](#) a présenté les méthodes de détection des anomalies qui se reposent sur les techniques de modélisation et d'estimation des paramètres et des états. [Sohlberg \[1998\]](#) considère la modélisation des processus comme méthode de prise de décision lors de la supervision des systèmes puisqu'un modèle décrit les entrées et sorties du système et peut être simulé. Deux types de modèles sont étudiés par le même auteur, le premier concerne les modèles boîte noire qui décrivent linéairement le système et qui sont souvent des modèles de régression linéaire ou des fonctions de transfert séparant les parties déterministe et stochastique (ces deux notions ont été présentées dans la Section 3.2.3). Le deuxième type concerne les modèles boîte grise qui se basent sur les paramètres qui ont un sens physique et qui permettent une description explicite du système [[Tulleken, 1993](#)]. [Sohlberg \[1998\]](#) présente les méthodes de l'estimation en ligne qui se voient très utiles pour les systèmes dynamiques

29. Il s'agit de tout système pouvant s'écrire sous la forme d'une équation matricielle : $AX = K$ où M est la matrice des coefficients, Y est le vecteur des constantes et X est le vecteur des inconnus.

changeant de paramètre dans un laps de temps très court. La maintenance préventive est, selon le même auteur, une méthode pratique pour prendre la décision lors de la supervision puisqu'elle permet d'avoir des informations permettant de prédire quand est ce que le système va mal fonctionner et de laisser le temps à l'opérateur pour exécuter les actions correctives avant même que le mal fonctionnement ait lieu.

Selon Åström and Hägglund [1995], les systèmes de supervision classiques n'évaluent pas correctement l'état du processus ou système supervisé parce que l'information n'est parfois pas disponible ou incomplète, ou parce que leurs stratégies ne sont pas aussi sophistiquées pour prendre en compte un grand nombre d'événements. A la suite de cette déclaration, le concept de supervision intelligente a commencé d'être développé, il inclue les techniques/méthodes/modèles intelligents permettant de faire une meilleure approximation des relations de correspondance entre les entrées et sorties d'un système dynamique, en tenant compte du comportement non linéaire [Takagi and Sugeno, 1993; Alique et al., 2000]. Haber et al. [2002] introduit les techniques de prise de décision basées sur la logique floue et l'analyse des partitionnements de données (clustering analysis) qui fait partie des techniques principales de fouille de données. [Venkatasubramanian et al., 2003] s'est opposé à l'utilisation des méthodes basées sur les modèles qui nécessitent des connaissances a priori, et a étudié l'application des techniques basées sur les données tracées du processus, appelées aussi historique. Ces dernières techniques sont classées selon leur nature (qualitative, quantitative, statistique) comme le montre la Figure 3.4. La recherche récente de Sobie et al. [2017] considère que les techniques basées sur l'historique sont limitées parce que les informations tracées ne sont valides que pour un contexte spécifique et introduit une nouvelle méthodologie d'apprentissage basée sur la simulation, qui permet d'entraîner les algorithmes d'apprentissage en utilisant des données générées par des simulations du système supervisé.

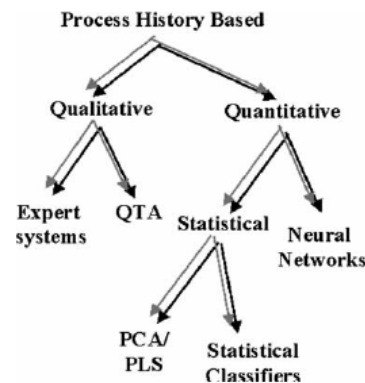


FIGURE 3.4 – Quelques méthodes de supervision basées sur l'historique [Venkatasubramanian et al., 2003]

3.2.5 Processus de prise de décision

Dans la Section 3.1, nous avons caractérisé la prise de décision par les quatre éléments suivants : le décideur, l'environnement de décision, le problème de décision et la théorie, l'approche, les méthodes et le processus de décision. Dans les sections précédentes nous avons introduit les six premiers éléments ; cette section a pour but d'introduire la notion de processus de prise de décisions et d'explorer quelques uns qui existent dans la littérature.

Hansson [2005] distingue deux types de processus de prise de décision, qui ont émergé après le modèle de Condorcet, le philosophe français ayant été le premier à définir les activités du processus de prise de décision. Le premier type est le modèle séquentiel, qui a été défini par Dewey [1910] comme étant l'ensemble des activités suivantes, exécutées séquentiellement :

- Identifier la difficulté,
- Définir les caractéristiques de cette difficulté,
- Suggérer les solutions possibles,
- Évaluer ces suggestions,
- Réaliser des expériences pour accepter ou rejeter la suggestion.

Ce processus a été modifié par [Simon \[1960\]](#) qui l'a adapté pour qu'il soit exploitable dans le milieu organisationnel. Le processus modifié comporte trois phases principales qui sont : Trouver les occasions qui nécessitent une décision, trouver les options possibles et puis choisir une de ces options. [Brim \[1962\]](#) reprend cette dernière version du processus et la redéfinit en intégrant les notions de la recherche d'information et de la performance :

- Identifier le problème,
- Chercher les informations nécessaires,
- Produire la liste des solutions possibles,
- Évaluer chaque solution,
- Sélectionner une stratégie pour la performance,
- Implémenter la décision.

[Witte \[1972\]](#) a critiqué le mode séquentiel du processus de prise de décision et a empiriquement prouvé que les activités peuvent être exécutées en parallèle et non pas nécessairement d'une façon consécutive. Quatre ans plus tard, [Mintzberg et al. \[1976\]](#) ont proposé un modèle circulaire du processus décisionnel comme le montre la Figure 3.5. Ce modèle a servi de base pour les processus de prise de décision actuels. Ces derniers, dont chacun dépend de l'environnement de décision, de la nature du problème, des caractéristiques du décideur, entre autres, ont fait l'objet de plusieurs travaux de recherche. Nous nous contentons, par la suite, de présenter le processus que nous supposons être le plus générique.

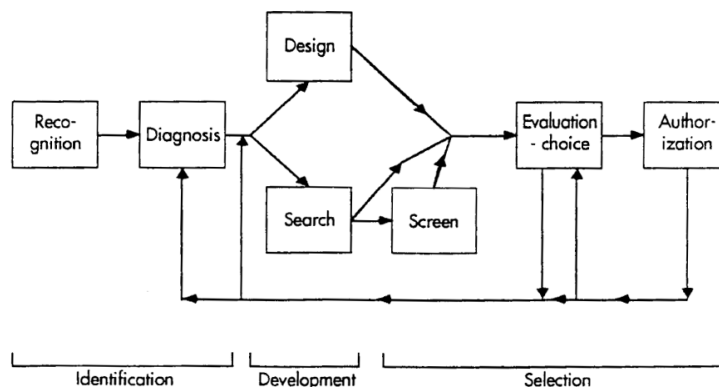


FIGURE 3.5 – Processus de prise de décision selon [Mintzberg et al. \[1976\]](#)

[Hammond et al. \[1999\]](#) introduit l'approche PROACT qui définit les huit éléments devant être pris en compte dans un processus de prise de décision pour faire le choix *intelligent* :

- **PROblème** : il s'agit ici de bien définir le problème de décision. En effet, la façon comment le problème est définie, dès le départ, a un grand impact sur la décision. Pour faire un choix intelligent, le problème doit être bien énoncé et formulé en identifiant son degré de complexité.
- **OBJECTIFS** : dans cette étape, le décideur doit se demander ce qu'il veut accomplir, ses intérêts, préoccupations, peurs et les aspirations les plus pertinentes pour atteindre ses objectifs.
- **ALTERNATIVES** : il s'agit ici de penser à créer des alternatives faisable, désirables et créatives. C'est une étape intéressante puisque la décision finale ne sera autre qu'une des alternatives identifiées ou créées.
- **CONSÉQUENCES** : dans cette étape, le décideur doit comprendre les conséquences de chaque alternative. Cette étude permet de cibler les alternatives qui répondront mieux aux objectifs fixés.
- **COMPROMIS (Tradeoffs)** : les objectifs sont parfois en conflit les uns avec les autres (comme par exemple : acheter une voiture de luxe, comme la *Lamborghini urus 2018*, à un prix inférieur à 30.000€) et les alternatives peuvent ne pas satisfaire tous les objectifs. Le décideur doit donc trouver un compromis en définissant ses priorités ou en choisissant l'alternative qui répond le mieux à tous les objectifs, ou la plupart d'entre eux.
- **INCERTITUDE, RISQUE, CONSIDÉRATION DES DÉCISIONS INTERDÉPENDANTES** : Ces trois éléments permettent de clarifier la décision dans un environnement volatile ou évolutif. En effet, une décision efficace

exige que le décideur soit confronté à l'incertitude en étudiant la probabilité des différents résultats possibles (auxquels une décision peut donner lieu) et en évaluant leurs impacts. Quand une décision inclue l'incertitude, la conséquence désirée peut ne pas être celle qui en résulte finalement (la greffe de cellules souches permet au patient de survivre à une leucémie comme elle peut ne pas être efficace), le décideur doit donc prendre en compte le niveau de risque de chaque alternative pour rendre le processus de prise de décision plus fluide et efficace. En outre, les décisions prises à l'état présent pourront influencer l'espace des choix dans le futur, pareillement, les objectifs du futur peuvent contraindre les décisions devant être prises à l'état présent, ainsi, les décisions sont liées les unes aux autres. Pour gérer ce type de décisions, il est nécessaire d'isoler et résoudre le problème à court terme tout en capitalisant les informations qui seront nécessaires pour résoudre les problèmes qui surviendront plus tard.

Ces éléments communs à tout processus de prise de décision, et plusieurs autres sont mis en œuvre dans un processus SDP (Systems Decision Process) présenté par Parnell et al. [2011] (Figure 3.6.a), qui a la particularité de pouvoir intervenir tout au long du cycle de vie de tout système (Figure 3.6.b). La première étape du SDP concerne la définition du problème comme l'indique l'élément *PROblème* de l'approche PROACT, cette étape inclut la définition des fonctions que doit exécuter le système et les exigences qu'il doit respecter ainsi que l'identification d'une méthodologie permettant d'évaluer les solutions candidates (alternatives). La deuxième étape du SDP est la conception des solutions qui commence d'abord par la génération des idées en utilisant plusieurs techniques de créativité comme le brainstorming, le brainwriting, la méthode delphi, etc. Une fois les idées générées, elles doivent être transformées en alternatives raisonnables et faisables en utilisant plusieurs approches de génération d'alternatives dont quelques unes sont exposées par Nutt [2001]. Le choix de la ou les solutions candidates est basé sur une étude quantitative et qualitative incluant entre autres le coût de l'alternative. Le décideur, ayant les solutions candidates entre les mains, doit affecter un score à chacune en utilisant des méthodes telles que la modélisation, les tests, les simulations, etc. L'ingénieur doit ensuite analyser la sensibilité de ses hypothèses de modélisation et des scores des solutions candidates avant de passer à l'étude des incertitudes et des risques régissant chacune de ces solutions. Après une étude de compromis qui tient compte des résultats des études précédentes, la décision finale sera générée et va ensuite être implémentée et surveillée pour s'assurer du bon fonctionnement du système après l'implémentation de la décision.

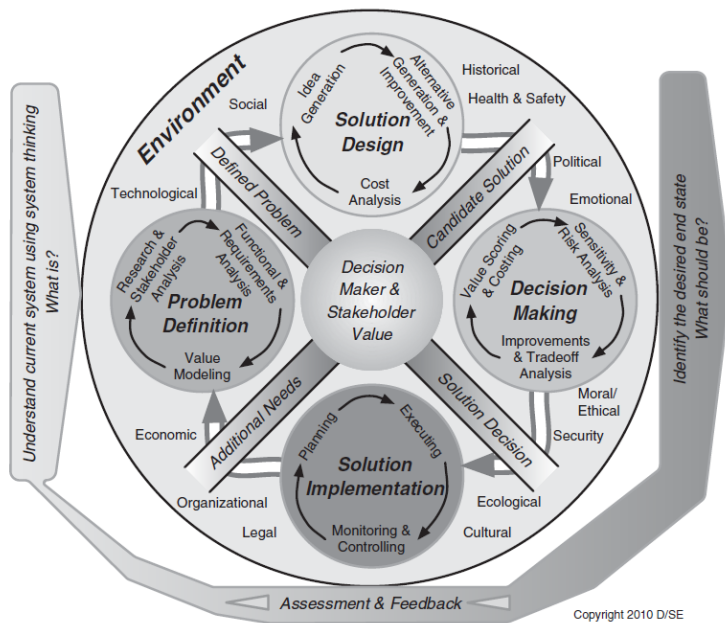
3.2.6 Quelques systèmes intelligents supportant la prise de décision

Initiée en 1956 par Pr. John McCarthy³⁰, l'intelligence artificielle (IA) est une discipline qui inclue les machines qui peuvent exécuter des tâches caractérisant l'intelligence humaine. Cette discipline, qui a reçu beaucoup d'intérêt par les scientifiques et les industriels, a connu un progrès significatif au cours des deux dernières décades. Aujourd'hui, les applications de l'IA couvrent plusieurs domaines y compris la prise de décision lors de la conception et la supervision des systèmes. En effet, les techniques de l'IA permettent, selon leurs caractéristiques, soit de supporter l'utilisateur au cours de son processus décisionnel (donnant lieu aux systèmes de recommandation et aux systèmes d'aide à la décision), soit de prendre la décision d'une façon plus autonome sans ou avec très peu d'intervention de la part du décideur (donnant lieu aux systèmes experts) [Ruhe, 2003; Ruhe and Ngo-The, 2004; Wang, 2007; Ford, 1985a]. L'objectif de cette sous-section est de découvrir ces systèmes en présentant un exemple illustrant l'utilisation de chacun d'eux lors de la conception ou supervision des systèmes.

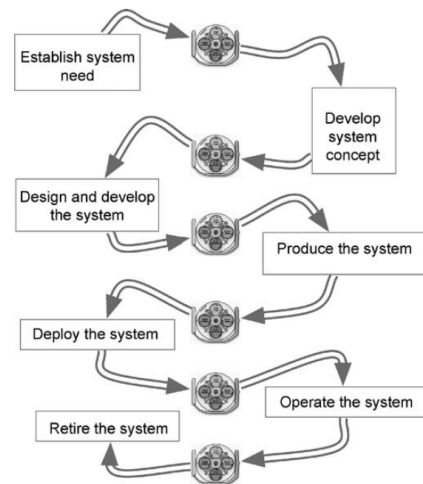
3.2.6.1 Les systèmes experts

La notion de système expert a vu le jour dans les années 1960 et le tout premier système expert était le programme *Dendral* [Lindsay et al., 1993], qui a pour but d'identifier les constituants chimiques d'un matériau à partir de la spectrométrie de masse et de la résonance magnétique nucléaire. Il n'existe pas une seule définition du système expert, mais la plus générique est donnée par Durkin [1996], qui le définit

30. C'est un ex scientifique chercheur en sciences informatiques à l'université de Stanford : <http://jmc.stanford.edu/>



3.6.a. Processus de prise de décision des systèmes (SDP)



3.6.b. Intervention du processus durant le cycle de vie du système

FIGURE 3.6 – Processus de prise de décision générique pouvant être exécuté à n'importe quel phase du cycle de vie du système [Parnell et al., 2011]

comme étant un programme informatique ou un logiciel qui capture les connaissances d'un expert humain concernant un problème appartenant à un domaine spécifique, et qui utilise ces connaissances pour résoudre des problèmes similaires. Lors de la conception des produits, par exemple, plusieurs ingénieurs collaborent pour atteindre l'objectif de la conception, ici, l'approche de système multi-expert, qui se voit comme étant une fédération de petits systèmes experts dont chacun résout un problème appartenant à un domaine spécifique, se voit plus appropriée [David, 1987].

Un système expert est constitué généralement par trois composants essentiels (Figure 3.7) : la base de connaissance, le moteur d'inférence et l'interface de communication [Hopgood, 2001; Nikolopoulos, 1997]. La base de connaissances est une collection organisée de faits concernant le domaine du système expert. Le moteur d'inférence interprète et évalue ces faits afin de fournir une réponse au problème de décision. Les faits relatifs à une base de connaissances doivent être acquis auprès d'experts humains au moyen d'entrevues et d'observations. Cette connaissance est alors généralement représentée sous la forme de règles "si-alors" (par exemple, si une condition c_1 est vraie, alors l'inférence i_1 peut être prise en compte). Puisque la base de connaissance d'un grand système expert comprend des milliers de règles, un facteur de probabilité devrait être souvent attaché à la décision finale relative à chaque règle, car la conclusion n'est pas à 100% certaine. Prenons comme exemple un système de surveillance d'un processus industriel qui pourrait indiquer, selon les informations qui lui sont fournies à l'instant t , une probabilité de 80% que le processus dévie de son fonctionnement normal. Un système expert peut afficher la séquence des règles par lesquelles il est arrivé à sa conclusion; tracer ce flux aide l'utilisateur à évaluer la crédibilité de sa décision et est utile comme outil d'apprentissage pour les nouveaux utilisateurs. Il existe un autre système d'aide à la décision qui se voit plutôt un type particulier des systèmes expert, il s'agit du système de conseil (Advisory Systems - AS) [Beemer and Gregg, 2008]. Les systèmes experts et de conseil partagent la même architecture, le seul point de différence est que les ASs guident le décideur dans son processus décisionnel mais ne présentent aucune décision Turban et al. [2005]. Comme exemple de système de conseil en conception, nous citons le "Design advisor", introduit par Chau and Albermani [2004], qui a pour objectif de guider les ingénieurs lors de la conception des structures de retenue de liquide (liquid-retaining structures).

Comme exemple d'utilisation des systèmes experts lors de la supervision, nous citons deux travaux de recherche. Le premier est le travail de Nabende and Wanyama [2008] qui ont développé le système expert

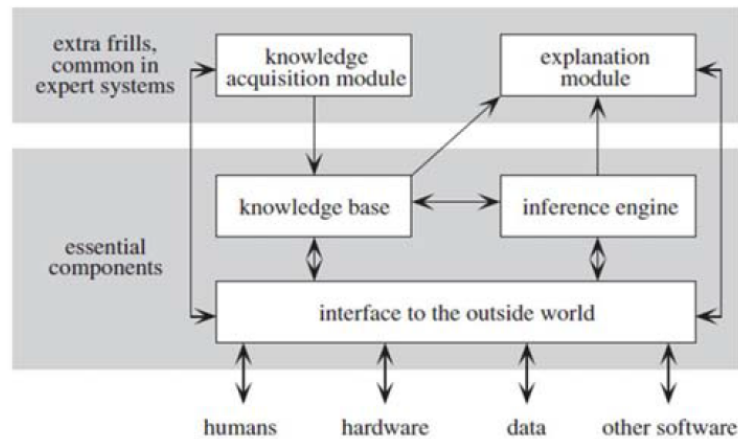
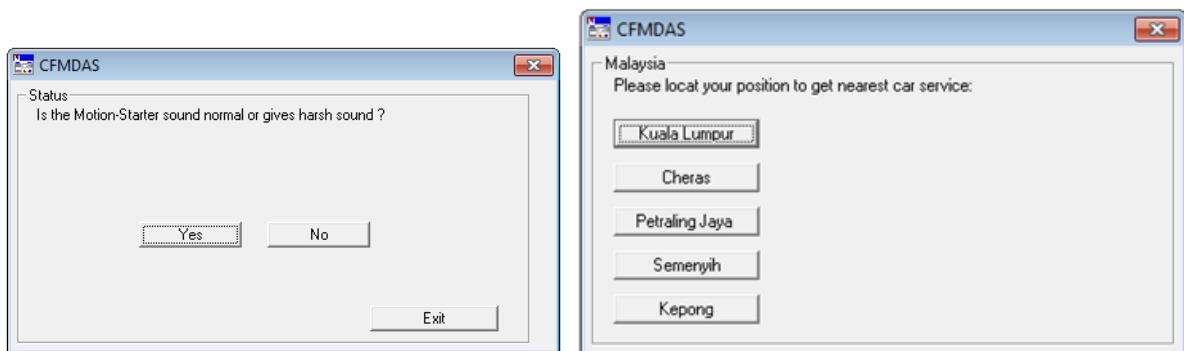


FIGURE 3.7 – Composants du système expert selon Hopgood [2001]

nommé *HDDE-ES* (Heavy Duty Diesel Engine Expert System). Le but principal de ce système est de fournir des conseils d'experts instantanés en traitant les problèmes des moteurs diesels de grande puissance, quand le temps est limité et quand le mécanicien spécialisé, qui doit prendre les décisions, n'est pas disponible. Le système *HDDE-ES* est basé sur la technologie BBN (Bayesian Belief Network) qui est un outil de raisonnement causal présentant le problème sous la forme d'un ensemble de nœuds inter-connectés avec des arcs où les nœuds représentent les hypothèses ou quelque chose qui a au moins deux valeurs possibles et les arcs représentent les probabilités conditionnelles entre les nœuds [Grover, 2012]. Le deuxième travail est celui de Mostafa et al. [2012] qui ont proposé un système expert nommé *CFMDAS* qui diagnostique les pannes de voiture et les classe. La première classe de problème est "l'état de démarrage", elle concerne les problèmes qui peuvent survenir lorsqu'une personne essaie de démarrer la voiture. La deuxième classe est "l'état de stabilité", elle concerne les problèmes pouvant survenir après le démarrage de la voiture. La troisième classe est "l'état du mouvement" qui concerne les problèmes pouvant survenir lorsque la voiture est en mouvement. Le système expert *CFMDAS* possède 150 règles pour différents types de défaillances et de causes et peut détecter plus de 100 types de défaillances. Grâce à l'interface utilisateur, la communication entre le système et l'utilisateur est effectuée sous la forme d'un menu qui affiche des questions aux utilisateurs et, ces derniers répondent par oui, non ou sélectionnent un choix parmi d'autres (Figure 3.8.a et Figure 3.8.b). Les connaissances sont ensuite stockées sous la forme de règles dans le système CLIPS (Système de production intégré en langage C) introduit par la NASA [Riley et al., 1989].



3.8.a. Boîte de dialogue de l'état de démarrage

3.8.b. Fenêtre de localisation

FIGURE 3.8 – Quelques fenêtres du système expert *CFMDAS* [Mostafa et al., 2012]

Dans le contexte de la conception, London et al. [1992] proposent un système expert appelé "guide expert des coûts et de la fabricabilité" (Expert Cost and Manufacturability Guide - ECMG), ce système fournit aux ingénieurs mécaniques l'estimation des coûts de fabrication, entre autres, très tôt lors du processus de conception. L'ECMG est un outil applicable à une variété de processus de fabrication. le Guide se

compose de deux logiciels, le premier, ECMG, est utilisé par un ingénieur en mécanique pour calculer les estimations de coûts de la conception préliminaire, pour comprendre les facteurs de fabricabilité qui influencent la conception et pour examiner les effets des compromis de conception sur les coûts et la fabricabilité. Le deuxième logiciel, ECMGEdit, fournit les outils nécessaires pour que les ingénieurs responsables peuvent faire des changements au niveau de la base de connaissances utilisée par ECMG. Lai [1993] introduit le système de conception expert appelé "la conception basée sur la connaissance pour l'assemblage" (knowledge Based Design for Assembly - KBDA), pour l'analyse et le raffinement de l'assemblage de produits. L'assemblage influence le cycle de développement lorsqu'il n'est pas pris en compte très tôt dans le processus de conception. KDBA effectue des analyses concernant les produits conçus et fournit des conseils pour le raffinement de la conception afin de tenir compte de leur assemblage. Le système est écrit en PROLOG³¹. Il implémente des règles de conception heuristiques acquises à partir des bonnes pratiques et des connaissances tirées des livres d'assemblage. Plusieurs travaux de recherche récents proposent des systèmes experts hybrides et/ou basés sur la logique floue et sur d'autres techniques récentes. Sahin et al. [2012] proposent une classification des systèmes experts hybrides selon plusieurs critères. Milanović et al. [2010] proposent un système hybride constitué par un système expert ayant pour but de diagnostiquer le composant défectueux et dont la sortie représente l'entrée du système d'aide à la décision qui permet de choisir le fournisseur pour le remplacement des composants défectueux en utilisant des analyses multi-critères. Les différences entre un système expert et un système d'aide à la décision sont clarifiées par Ford [1985b] et sont de plusieurs natures (différences au niveau des objectifs, des opérations, des utilisateurs, méthodologie de développement). Dans la suite du manuscrit, nous gardons les points de différence suivants :

- Un **système d'aide à la décision** permet d'extraire les connaissances et/ou apprendre pour permettre à l'utilisateur de traiter les problèmes non ou semi-structurés d'une manière flexible et personnelle dans la manipulation des données et des modèles. Ce système ne remplace pas le décideur.
- Un **système expert**, après l'injection des connaissances dans la base par un expert permet de proposer des décisions à l'utilisateur qui a peu ou pas de flexibilité lors du traitement des problèmes structurés puisque la décision est plutôt automatisée. Ce système peut remplacer le décideur.

3.2.6.2 Les systèmes de recommandation

Avec la croissance continue du Web pour le business et les transactions de commerce électronique, les utilisateurs peuvent de plus en plus exprimer leurs goûts et avis concernant les produits, à travers les notations et/ou les commentaires positifs ou négatifs. Ceci a servi de moteur principal au développement de la technologie des systèmes de recommandation (Recommender System - RS) qui sont apparus comme un domaine de recherche indépendant au milieu des années 1990 [Aggarwal et al., 2016; Anand and Mobasher, 2003] et qui ont été adopté initialement par Amazon.com [Linden et al., 2003], puis par plusieurs compagnies tels que LinkedIn [Rodriguez et al., 2012], Netflix [Steck, 2013], Google [Liu et al., 2010a], etc. Les RSs sont définis par Rashid et al. [2002] comme une stratégie de prise de décision pour les utilisateurs dans des environnements d'information complexes et par Portugal et al. [2017] comme des systèmes qui utilisent les techniques de l'intelligence artificielle pour proposer des recommandations aux utilisateurs. Selon Ricci et al. [2011], Les RS visent principalement les personnes qui n'ont pas suffisamment d'expérience ou de compétences personnelles pour évaluer le nombre potentiellement élevé d'autres éléments qu'un site Web, par exemple, pourrait offrir [Resnick and Varian, 1997]. Les RSs sont aussi définis comme étant des outils logiciels ou des techniques fournissant des suggestions d'éléments (items) qui pourraient être utiles pour un utilisateur donné [Resnick and Varian, 1997; Burke, 2007]. Ces suggestions concernent plusieurs processus décisionnels tels que les articles à acheter d'un site de commerce électronique, la musique à écouter, les films à regarder, etc. Un RS se concentre sur un type spécifique d'item et par conséquent, sa conception, son interface graphique et les techniques de recommandation sont toutes

31. PROgrammation en LOGique, c'est un langage de programmation logique : <https://en.wikipedia.org/wiki/Prolog>

spécifiques à ce type d'items pour fournir des suggestions utiles et plus efficaces à l'utilisateur [Ricci et al., 2011].

Il existe différentes façons comment un RS peut être formulé, ces formulations suivent généralement les deux approches suivantes [Aggarwal et al., 2016] :

- **Approche prédictive du problème** : Elle consiste à prédire la valeur d'évaluation pour une combinaison "utilisateur-article". Il est supposé que les données d'apprentissage sont disponibles, indiquant les préférences pour les articles. Avoir m utilisateurs et n éléments, correspond à une matrice $m \times n$, où les valeurs spécifiées (ou observées) sont utilisées pour entraîner un modèle de prédiction, et les valeurs manquantes (ou non observées) sont prédites en utilisant ce modèle de prédiction.
- **Approche évaluative du problème** : cette approche vise à recommander les éléments top-k pour un utilisateur particulier, ou déterminer les utilisateurs top-k à cibler pour un article particulier, sachant qu'en pratique, la détermination des éléments top-k est plus populaire que la détermination des utilisateurs top-k, et que les méthodes dans les deux cas sont exactement les mêmes.

Le processus de recommandation est constitué par trois phases :

- **Phase de collecte de l'information** : ici sont collectées les informations pertinentes sur les utilisateurs, à travers des feedbacks, pour générer un profil utilisateur ou un modèle qui contiennent, entre autres, les attributs de l'utilisateur, ses capacités intellectuelles, ses intérêts, ses préférences, son style, son comportement et les ressources auxquelles il accède. Des modèles/profils précis sont indispensables pour générer des recommandations pertinentes. Trois types de feedback sont présents dans la littérature. Le premier est explicite et concerne l'intérêt de l'utilisateur pour un élément. Il nécessite la participation de l'utilisateur puisque le RS l'invite, à travers son interface, à fournir des notes pour l'élément. Le deuxième feedback est implicite, il infère automatiquement les préférences de l'utilisateur en surveillant son comportement et ses différentes actions telles que l'historique des achats, l'historique de navigation, le temps passé sur certaines pages Web, les clics sur les boutons, etc. Le dernier feedback est hybride et peut être obtenu grâce à la combinaison des deux feedbacks explicite et implicite. La combinaison peut être réalisée en utilisant une donnée implicite comme une vérification de l'évaluation explicite [Isinkaye et al., 2015].
- **Phase d'apprentissage** : cette phase applique un algorithme d'apprentissage pour filtrer et exploiter les caractéristiques de l'utilisateur à partir des informations recueillies au cours de la phase précédente.
- **Phase de prédiction ou recommandation** : en utilisant le modèle développé dans la phase précédente, cette phase prédit le type d'éléments qu'un utilisateur peut préférer.

Il existe plusieurs types de RS dans la littérature dont la différence clé est la technique utilisée (Figure 3.9) :

- **Technique de filtrage basée sur le contenu** (Content-based filtering technique) : c'est un algorithme qui met l'accent sur l'analyse des attributs des éléments afin de générer des prédictions. Cette technique de filtrage basée sur le contenu est la plus efficace lorsqu'il s'agit de recommander des documents tels que des pages Web, des publications, etc. La recommandation, ici, est faite sur la base des profils d'utilisateurs en utilisant des caractéristiques extraites du contenu des éléments évalués par l'utilisateur dans le passé. Les articles qui sont principalement liés aux articles positivement évalués sont recommandés à l'utilisateur. Cette technique utilise différents types de modèles pour trouver la similarité entre les documents afin de générer des recommandations significatives, ces modèles sont présentés par Isinkaye et al. [2015]. Cette technique a l'avantage d'ajuster les recommandations dans un délai très court si le profil de l'utilisateur change. En revanche, son principal inconvénient est la nécessité d'avoir une connaissance approfondie et une description des caractéristiques des éléments dans le profil utilisateur.
- **Technique de filtrage collaboratif** (Collaborative filtering technique) : c'est une technique de prédiction indépendante du domaine pour les éléments qui ne peuvent pas être décrits facilement par des méta-données tels que les films et la musique. Elle commence par créer une base de données de préférences des items par utilisateur (matrice utilisateur-item), après elle regroupe les utilisateurs

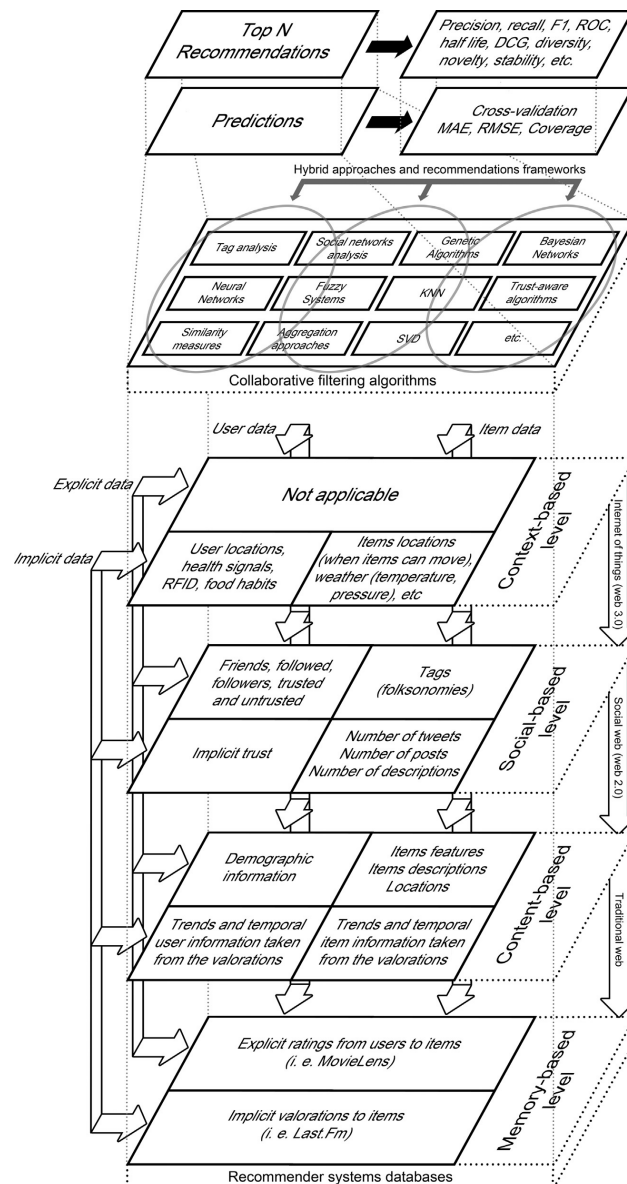


FIGURE 3.9 – Les différentes techniques de recommandation selon Bobadilla et al. [2013]

avec des intérêts/préférences pertinents en des groupes, appelés voisinages, en calculant les similarités entre leurs profils. Un utilisateur obtient, donc, des recommandations pour les éléments qu'il n'a pas notés auparavant mais qui étaient déjà positivement notés par les utilisateurs de son voisinage. Une étude comparative des algorithmes utilisés dans cette technique est faite par Lee et al. [2012]. Deux catégories de cette technique sont présentes dans la littérature : la première est basée sur la mémoire et la deuxième est basée sur les modèles [Aggarwal et al., 2016] :

- Dans la catégorie des techniques basées sur la mémoire, les éléments qui ont déjà été évalués par l'utilisateur avant jouent un rôle important dans la recherche d'un voisin qui partage l'appréciation avec lui, et elle peut être réalisée de deux manières selon la technique utilisée qui peut être basée sur l'utilisateur ou sur les items. La technique de filtrage collaboratif basée sur l'utilisateur calcule la similarité entre les utilisateurs en comparant leurs évaluations sur le même article, puis calcule la note prédite pour un article par l'utilisateur actif comme moyenne pondérée des évaluations de l'article par des utilisateurs similaires à l'utilisateur actif. La technique de filtrage collaboratif basée sur l'item calcule des prédictions en utilisant la similarité entre les items et non entre les utilisateurs. Et sur la base de ces similarités, la préférence de l'utilisateur pour un item non évalué par lui est calculée.

- Les techniques basées sur un modèle utilisent les évaluations précédentes pour développer un modèle en utilisant les techniques d'apprentissage ou de fouille de données et ce modèle est ensuite utilisé pour produire des prédictions (recommandations). Les différentes méthodes d'apprentissage utilisées par les RS sont présentées par [Zhang et al. \[2017\]](#) et [Nunes and Jannach \[2017\]](#).
- **Technique de filtrage hybride** : Elle combine différentes techniques de recommandation pour obtenir un RS optimal et pour surmonter à quelques limitations des RS [[Stern et al., 2009](#); [Adomavicius and Zhang, 2012](#)]. Il est aussi noté qu'une combinaison d'algorithmes fournira des recommandations plus précises et efficaces qu'un seul algorithme [[Schafer et al., 2007](#)].

[Liu et al. \[2017\]](#) discute l'applicabilité des RS dans la conception de produits et introduit le FRS (Function Recommender System), un système basé sur l'approche hybride, qui se voit comme un nouveau paradigme de la recommandation des nouvelles fonctions pour les produits ou services cibles. Le système estime le classement d'une fonction candidate pour un produit cible, sachant que la fonction est extraite d'un produit qui est similaire au produit cible. [Pereira et al. \[2017\]](#) ont développé un RS qui permet de comprendre les besoins et préférences des clients pour guider les décideurs lors du processus de configuration de produit. Par conséquent, au lieu de considérer toutes les solutions de l'espace de configuration lors du processus décisionnel, les décideurs ne considèrent que certaines caractéristiques, recommandées par le RS, pour configurer un produit. [Moon et al. \[2008\]](#) présente un RS permettant de supporter la conception des familles de produits en se basant sur l'étude des préférences des clients dans un environnement dynamique.

3.2.6.3 Les systèmes de support de prise de décision (DSS)

Depuis le début des années 1960, de nombreux aspects opérationnels au sein des organisations étaient informatisés, pour exécuter les applications telles que le traitement des commandes, la facturation, le contrôle des stocks, etc. De cette mobilisation, le concept de DSS, qui a été basé sur l'idée d'utiliser les ordinateurs pour supporter le processus de prise de décision [[Bonini, 1963](#)], a vu le jour. Il a été initialement mis en œuvre par [Gorry and Scott Morton \[1971\]](#), qui avaient pour objectif d'améliorer les systèmes d'information de gestion, en introduisant un DSS qui se base sur les catégories de l'activité managériale de [Anthony \[1965\]](#) et la taxonomie des types de décisions de [Simon \[1960\]](#). [Gorry and Scott Morton \[1971\]](#) ont défini leur DSS comme un système qui soutient toute activité managériale lors du traitement des problèmes de décision non ou semi-structurés. Dans les décennies qui ont suivi, les recherches concernant les DSSs ont connu un progrès important et ont divergé vers un domaine multidisciplinaire incluant les domaines des systèmes d'information, de la recherche opérationnelle, de la théorie de la décision et des études organisationnelles, entre autres [[Merkert et al., 2015](#)]. Suite aux déclarations de [Simon \[1980\]](#) qui suggère d'inclure les techniques de l'IA aux DSSs, les chercheurs de l'IA ont joint la communauté de recherche des DSSs, favorisant ainsi l'émergence de nombreuses approches et catégories des DSSs; ces systèmes qui peuvent être définis comme des systèmes d'informations interactifs basés sur l'ordinateur et ayant pour objectif de supporter la prise de décision dans plusieurs domaines (médecine, agriculture, ingénierie, etc.) [[Shim et al., 2002](#)].

Pour gérer cette diversité, plusieurs travaux de recherche ont essayé de classer les DSSs selon plusieurs perspectives. [Donovan and Madnick \[1977\]](#) proposent deux classes des DSSs selon la nature des décisions à prendre. Dans la catégorie **institutionnelle**, les DSSs supportent les décisions continues et récurrentes. La deuxième catégorie, **Ad hoc**, concerne les DSSs qui supportent les décisions ponctuelles. [Hackathorn and Keen \[1981\]](#) divisent les DSSs en trois classes : Les DSSs **personnels**, les DSSs de **groupe** et les DSSs **organisationnels**. Selon [Alter \[1980\]](#), les DSSs se divisent en sept classes selon le type de leurs opérations comme le montre le Tableau 3.5. [Carlsson et al. \[1998\]](#) considèrent que les DSSs se divisent en deux classes dont la différence dépend du niveau de support fourni :

- **Les DSSs passifs** : ces systèmes ne produisent pas des suggestions de décision explicites parce que leur objectif est plutôt de simplifier et transformer les problèmes non structurés en un ensemble de tâches bien définies et non ambiguës avant de commencer le processus de résolution de problème.

- **Les DSSs Actifs** : ce type de systèmes sont capables de produire des suggestions ou solutions explicites après l'analyse du problème non ou semi structuré.

La classification moderne proposée par Power [2002], divise les DSSs en cinq grandes classes :

- DSSs **dirigés par les modèles** : ils utilisent des modèles quantitatifs basés sur l'heuristique, l'optimisation, la simulation, entre autres, pour obtenir des solutions aux problèmes de décisions [Power and Sharda, 2007].
- DSS **dirigés par les données** : ils mettent l'accent sur l'accès et la manipulation d'une série chronologique de données internes, parfois externes, à l'entreprise [Power, 2008b].
- DSSs **dirigés par les communications** : ils utilisent les technologies de réseau et de communication pour prendre en charge la collaboration et la communication décisionnelles. Dans ces systèmes, les technologies de communication sont l'élément le plus important [Power, 2008a].
- DSSs **dirigés par les documents** : ils utilisent les techniques de stockage et de traitement informatique pour récupérer et analyser les documents [Power, 2008a].
- DSSs **dirigés par les connaissances** : ils capturent, stockent et analysent l'expertise pour proposer les décisions.

Orientation	Type de système	Description
Orienté données	File Drawer Systems	Permettent un accès immédiat à la donnée.
	Systèmes d'analyse de données	Permettent la manipulation des données grâce à l'utilisation de paramètres ou d'outils informatisés spécifiques ou génériques.
	Systèmes d'analyse d'informations	Ce type de DSSs donne accès à des ensembles de bases de données orientées vers la décision et de modèles simples.
Orienté modèles	Modèles compatibles et financiers	Ce type de DSS peut effectuer l'analyse "what if" et calculer les résultats de différents chemins de décision.
	Modèles représentationnels	Ce type de DSS peut également effectuer l'analyse "what if" et calculer les résultats de différents chemins de décision, en se basant sur des modèles simulés.
	Modèles d'optimisation	Ce type de DSS fournit des solutions grâce à l'utilisation de modèles d'optimisation.
	Modèles de suggestions	Ce type de DSS fonctionne lorsque la décision à prendre est basée sur des tâches bien structurées.

TABLE 3.5 – Les classes des DSSs (adapté de Alter [1980])

Plusieurs autres variantes de DSSs existent dans la littérature [Burstein and Holsapple, 2008], dont la plus populaire concerne les DSSs basés sur le web [Bhargava et al., 2007]. Ces derniers permettent de fournir des outils d'aide à la décision à un gestionnaire ou à un analyste commercial à l'aide d'un navigateur Web (client léger). Le serveur informatique hébergeant l'application DSS est relié à l'ordinateur de l'utilisateur client par un réseau avec le protocole TCP/IP. Les DSSs basés sur le Web peuvent être dirigés par les communications, les données, les documents, les connaissances, les modèles ou hybride combinant deux ou plusieurs classes.

Un DSS est généralement constitué par trois composants essentiels comme le montre la Figure 3.10. Le premier concerne les entrées à analyser, comme les bases de données, des connaissances, des modèles, entre autres. Le deuxième composant permet de transférer les décisions générées par le DSS à l'utilisateur, c'est généralement une interface graphique où sont affichées les décisions proposées sous la forme d'un texte,

de tables, de diagrammes, de graphiques, etc. Le troisième composant concerne le moteur de traitement qui adopte une ou plusieurs techniques d'analyse. Plusieurs techniques peuvent être utilisées par un DSS, [Shim et al. \[2002\]](#) présente l'évolution de ces technologies, notamment l'utilisation des techniques de fouille de données et de l'intelligence artificielle, qui donnent lieu aux systèmes d'aide à la décision intelligents (Intelligent Decision Support Systems - IDSS). Les techniques de fouille de données se voient comme un moyen efficace pour améliorer la qualité des décisions prises selon [Liu et al. \[2010b\]](#) (Figure 3.11). En effet, elles permettent d'analyser les données, en grande quantité, pour identifier les relations éventuelles qui les lient et en extrayant les connaissances pertinentes qui pourront être utilisées pour proposer les décisions qui seront selon [Liu et al. \[2010b\]](#) d'une qualité meilleure que celles générées par les DSSs classiques qui se basent plutôt sur les informations. Notons que le débat "Donnée-Information-Connaissance" est loin d'être l'objectif de ce travail de recherche. Il existe plusieurs travaux, traitant ce sujet, auxquels le lecteur peut se référer (comme par exemple celui de [Liew \[2007\]](#)).

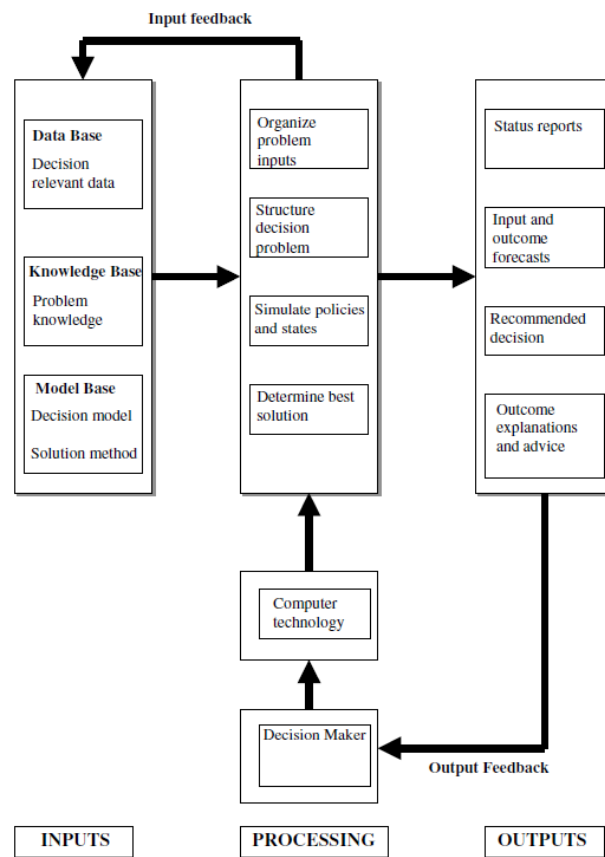


FIGURE 3.10 – Architecture générale d'un DSS [[Phillips-Wren et al., 2009](#)]

Comme illustration de l'utilisation des DSSs dans la conception des systèmes, nous citons le travail récent de [Seletos and Salmon \[2017\]](#) qui a présenté le système intégré : "System DSS". Ce dernier se compose d'un DSS (partie en bas de la Figure 3.12) et d'un PDG (Product Design Generator) [[Roach et al., 2005](#)] qui occupe la partie haute de la Figure 3.12. Le système intégré a pour objectif d'analyser les effets de la conception d'un produit sur le business. En effet, le PDG permet d'assister le concepteur en générant des milliers de conceptions potentielles, et le DSS permet de supporter les investigations concernant ces conceptions, de présenter les données commerciales et d'optimiser les pratiques actuelles de l'entreprise (données d'entrée du DSS de la Figure 3.12)

L'utilisation des DSSs dans la supervision des systèmes peut être illustrée par le travail de [Guo et al. \[2015\]](#) où est présenté le système pilote ayant pour objectif la supervision et la planification des processus de production des entreprises manufacturières distribuées comportant plusieurs usines de production. Ce dernier est basé sur la technologie RFID (Radio Frequency IDentification) qui, selon les auteurs, permet d'améliorer la visualisation et la traçabilité des informations de la chaîne logistique. En supervisant le

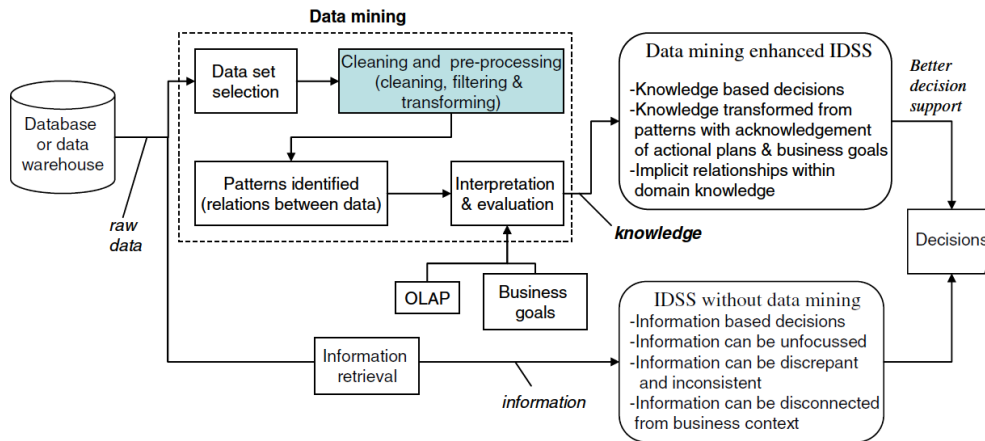


FIGURE 3.11 – Amélioration des IDSS avec les techniques de fouilles de données [Liu et al., 2010b]

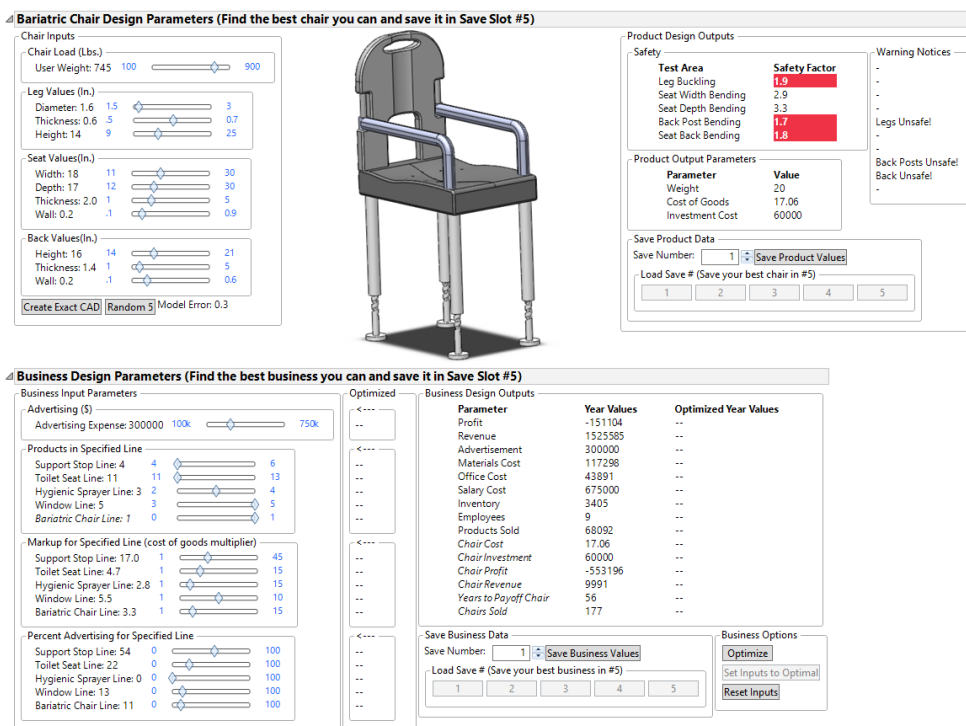


FIGURE 3.12 – Système DSS intégrant un DSS (en bas) et un PDG (en haut) [Seletos and Salmon, 2017]

processus de production, le système a la capacité de comparer l'efficacité du processus et le progrès des différentes opérations dans une unité de production, et permet ainsi d'identifier les opérations qui retardent le processus (Open 11 dans la Figure 3.13) et qui nécessitent d'être ajustées pour équilibrer le processus.

3.2.7 Synthèse et discussion de l'état de l'art académique

Dans cette première partie de l'état de l'art, nous avons présenté les éléments essentiels qui caractérisent la décision. Pour identifier la technique, le processus et le système de prise de décision, nous avons suivi la logique présentée dans la Figure 3.1. Maintenant, nous allons procéder à la projection de ces éléments sur notre contexte de travail (établi dans la Section 2.2.3), pour identifier les travaux qui nous semblent les mieux adaptés pour répondre à notre problématique.

Un problème de décision lors de la conception et/ou la supervision des systèmes est de nature non ou semi structurée. En effet, lors de la conception d'une voiture par exemple, un ingénieur peut choisir une certaine forme parce qu'il l'a trouvée originale et attractive. Ces deux caractéristiques ne sont pas décrites



FIGURE 3.13 – Interface du système Pilote [Guo et al., 2015]

(ou partiellement décrites) par des règles parce qu'elles sont souvent subjectives. Pareillement, lors de la supervision d'un processus industriel, l'opérateur compte sur son expertise pour résoudre un problème d'anomalie qui n'a jamais eu lieu auparavant. Par conséquent, les autres caractéristiques de la décision (théorie, approche, techniques/méthode et le processus de décision) sont contraintes par la nature du problème, qui est dans notre contexte, **semi ou non structuré**.

Nous allons maintenant identifier la théorie de décision qui nous paraît la plus appropriée, parmi celles présentées dans la Section 3.2.2 (théories normatives, descriptives, prescriptives et constructives), pour supporter l'utilisateur pendant son processus décisionnel. Ce dernier se compose des décisions automatisées (comme le calcul du périmètre d'un disque, de rayon R , qui consiste à lancer la procédure permettant de calculer le résultat de $2 \times \pi \times R$) et non automatisées (comme les décisions subjectives qui dépendent du comportement de l'utilisateur et/ou le contexte). Par conséquent, nous gardons les **théories normatives, descriptives et prescriptives**.

Le choix précédent, notre contexte de recherche et notre hypothèse contraignent le choix de la ou les approches à choisir comme le montre la Figure 3.1. En effet, notre contexte de travail est caractérisé par sa complexité, son aspect dynamique et la présence des décisions non automatisées et celles pouvant faire appel à des procédures comme expliqué dans le paragraphe précédent. De plus, notre hypothèse consiste à considérer l'historique et les anciennes expériences pour proposer les nouvelles décisions. De ces faits, nous considérons une **approche hybride qui couple les approches rationnelles, informatives, bayésiennes, dynamiques et naturalistes**. Par conséquent, nous gardons la classe des **techniques basées sur l'intelligence artificielle** qui sont à la fois descriptives, prescriptives, analytiques et sont conformes à l'approche hybride que nous avons considérée ainsi qu'à la question de recherche. De plus ces techniques peuvent être appliquées aussi bien dans la conception que dans la supervision des systèmes.

Concernant le choix du processus de décision, qui dépend bien des choix précédents, nous voyons qu'aucun des processus présentés ne permet de lever les verrous scientifiques que nous avons identifiés dans la question de recherche. En effet, aucun des processus étudiés ne considère les activités de modélisation et de traçabilité des processus et celles de l'aide à la décision et la gestion des changements contextuels.

La technique et processus de prise de décision doivent, ensuite, être implémentés sous la forme d'un système permettant de supporter la prise de décision. Pour identifier la nature de ces systèmes et tenant compte des

choix précédents, nous avons présenté quelques **systèmes intelligents** permettant de supporter la prise de décision (systèmes experts, systèmes de recommandation et les systèmes d'aide à la décision). Le choix d'un de ces systèmes est, en plus de nos choix précédents, contraint par les caractéristiques recherchées, que nous avons établies dans la Section 2.3.4. Le résultat de la comparaison de ces systèmes par rapport à ces caractéristiques est présenté dans le Tableau 3.6.

	Systèmes experts	Systèmes de recommandation	Systèmes d'aide à la décision
Modéliser et visualiser le processus	-	-	-
Tracer le processus	-	-	-
Aider à décider	~	~	✓
Identifier et propager les changements contextuels	-	-	-
Open source			
Modulaire d'architecture			
Générique			
Indépendant			
Semi-Autonome/Intelligent	-	~	~
Capacité de support			
Facile			
Extensible			
Maintenable			
Portable			
Efficient			
Efficace			
Fiable			
Orienté activité et processus	-	-	-

TABLE 3.6 – Comparaison des systèmes intelligents par rapport aux caractéristiques recherchées. "✓" (resp. "-") veut dire que le système répond parfaitement (resp. ne répond pas) à l'exigence, "~" veut dire qu'il répond en partie à l'exigence et la case vide veut dire qu'il n'a pas été testé.

Le Tableau 3.6 montre que les systèmes d'aide à la décision présentés ont la possibilité d'**aider à décider** exactement comme nous l'avons décrit et sont semi-autonomes mais pas tout à fait intelligents pour apprendre des décisions passées (d'où le symbole "~" sur le Tableau 3.6). Aucun des systèmes présentés n'a la capacité de modéliser et visualiser les processus ni de tracer les informations et identifier et propager les changements et, de plus, ne sont pas orientés activités et processus. Les autres caractéristiques ne peuvent, malheureusement, pas être évaluées puisque les systèmes présentés ne sont pas publiquement accessibles.

3.3 État de l'art commercial : outils supportant la prise de décision

3.3.1 Blueworks Live d'IBM

L'International Business Machines³², connue sous son acronyme IBM, est une multinationale américaine caractérisée par sa forte présence dans le domaine informatique et dont les produits incluent les logiciels, les intergiciels, le matériel informatique et les services comme les prestations de conseil et l'hébergement des données. IBM reconnaît que les organisations sont amenées à prendre rapidement les bonnes décisions pour s'adapter aux conditions actuelles du marché (discutées plus en détails dans le Chapitre 2), et affirme que la capacité de découvrir et automatiser le processus de prise de décision permet aux entreprises non seulement

32. <https://www.ibm.com/us-en/>

de survivre mais aussi de gagner un avantage compétitif (Chapitre 2).

IBM voit la découverte de décision (Figure 3.14) comme étant le processus dans lequel sont identifiées et documentées les décisions métier clés qui sont régulièrement prises au sein de l'organisation [Thorpe et al., 2014]. La première activité de ce processus est de comprendre les décisions, puisque ces dernières ne sont pas claires pour toute l'équipe vu que les connaissances sont dissimulées dans les têtes des experts, dans des codes informatiques ou dans des documents et nécessitent, ainsi, d'être découvertes pour dégager les informations qui justifient pourquoi les décisions ont été prises, afin de les analyser, les améliorer et potentiellement les automatiser.

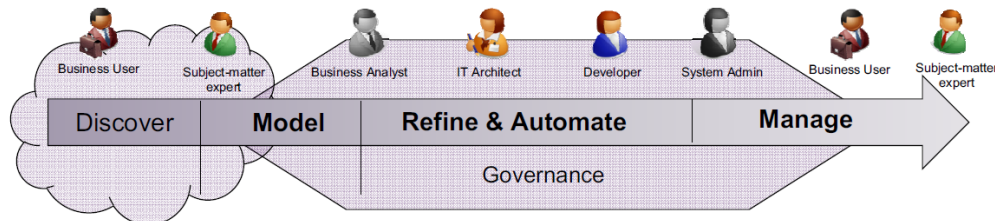


FIGURE 3.14 – Processus de découverte de décision [Thorpe et al., 2014]

IBM commercialise le "Blueworks Live", une plateforme permettant de documenter, graphiquement et textuellement, à la fois le processus métier et les décisions prises au cours de ce dernier. Comme le montre la Figure 3.15, le processus est modélisé graphiquement selon la norme BPMN de l'OMG [2013], et la décision est modélisée sous la forme d'une "Decision Task" associée au concept "Business Rule Task" toujours de la norme BPMN. Chaque tâche décisionnelle est associée à un diagramme de décision où elle est bien détaillée et décomposée selon la norme DMN de l'OMG (Figure 4.13 de la Section 4.3.3). Les caractéristiques principales de chaque décision (comme les raisons qui amènent à sa prise, son ordre d'importance en termes de son effet sur la performance, sa fréquence, etc.) sont documentées dans l'onglet "Details" du logiciel, et les détails additionnels peuvent être capturés dans l'onglet "Documentation". Le logiciel offre la possibilité de joindre des documents dans l'onglet "Attachments" pour bien justifier la décision (Figure 3.16).

Après avoir défini le diagramme de décision (en décomposant cette dernière et en spécifiant ses données d'entrée et de sortie), Blueworks Live génère une table de décision (ce concept a été introduit dans la Section 4.3.4) basée sur le diagramme de décision mais qui reste personnalisable (possibilité de ré-arranger les colonnes, les renommer, etc). Une fois la table est validée, l'utilisateur peut remplir la logique suivie pour prendre la décision. Après la documentation complète de cette dernière, les experts ou les parties prenantes peuvent ensuite la valider. Le logiciel permet de faciliter l'automatisation de la validation des décisions en envoyant automatiquement les notifications quand il y a des décisions en attente de revue et validation.

Le *Blueworks Live* est un outil facile à utiliser, offrant aux experts et aux analystes métiers la possibilité de travailler ensemble pour capturer, raffiner et améliorer les modèles de processus et de décision. Il permet aussi de :

- Centraliser l'information en offrant une base de données partagée, permettant ainsi de faciliter la collaboration et l'amélioration des processus,
- Aider les utilisateurs à mieux comprendre les processus et décisions à travers leur modélisation graphique,
- Tracer, documenter et sauvegarder les informations concernant les processus et les décisions,
- Tracer les changements au niveau des décisions avec la possibilité de restaurer les anciennes décisions,
- Exporter les diagrammes et tables de décisions ainsi que toutes les informations sous la forme de documents Microsoft word et Excel,
- Implémenter, exécuter et optimiser les processus à travers son intégration directe à "IBM Business Process Manager", outils de gestion de processus.

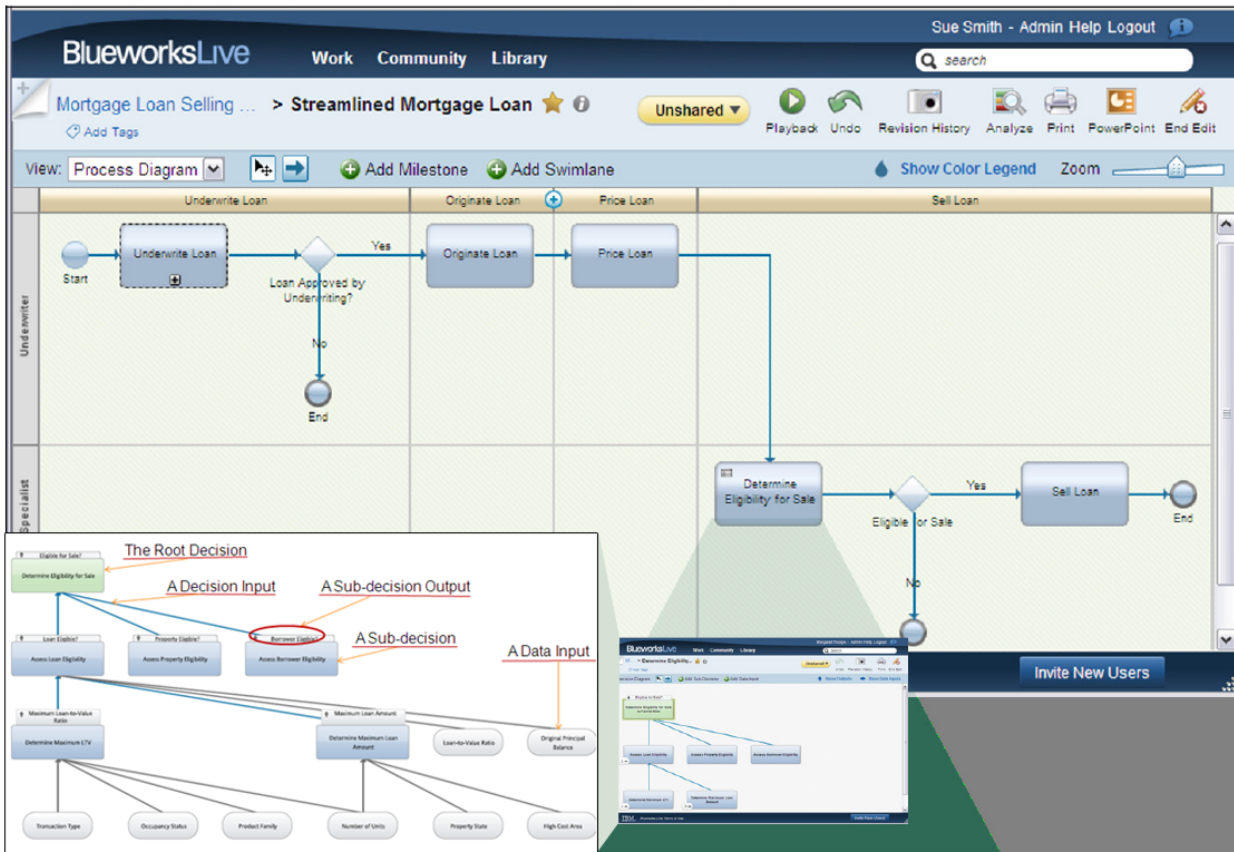


FIGURE 3.15 – Exemple d’une tâche décisionnelle au sein d’un processus métier dans ”Blueworks Live” (adapté de [Thorpe et al., 2014])

The screenshot shows the documentation for the 'Determine Eligibility for Sale' decision. It includes the following sections:

- Business Motivation:** Ensure that those loans are originated in compliance with secondary mortgage mark
- Business Owners:** SVP Mortgage Operations
- Experts:** Credit Policy Analyst (Carol)
- Sources:** Fannie Mae Selling and Servicing Guides
- Key Performance Indicators:** Portfolio Default Rate
- How often is this decision made?:** a couple of thousand times a day (Volume: High)
- How often is this decision updated?:** usually quarterly, sometimes monthly (Frequency: Medium)
- How much time do you have to update the decision?:** usually a month (Latency: Medium)
- Decision Description:** In order to sell this loan to Fannie Mae, the borrower, the property and the loan must all comply with their eligibility requirements.
- Inputs and Output:**
 - Inputs: Loan Eligible?, Property Eligible?, Borrower Eligible?
 - Output: Eligible for Sale?
- Decision Table:** Determine Eligibility for Sale to Fannie Mae

Considerations			Conclusions
Loan Eligible?	Property Eligible?	Borrower Eligible?	Eligible for Sale?
yes	yes	yes	yes
no	-	-	no
-	-	-	no

FIGURE 3.16 – Documentation des caractéristiques clés de la décision et sa logique dans ”Blueworks Live” (adapté de [Thorpe et al., 2014])

3.3.2 SAS Event Stream Processing

Le SAS Institute³³, multinationale américaine ayant pour activité le développement d’une suite de logiciels et systèmes analytiques appelée SAS (Statistical Analysis System), permettant d’aider les entreprises à accéder, analyser et gérer leurs données afin de supporter leurs processus de prise de décision. SAS

33. https://www.sas.com/fr_fr/home.html

commercialise le "SAS Event Stream Processing", une solution d'analyse des flux de données en temps réel (*Streaming analytics* en Anglais) qui inclut un serveur offrant un environnement d'exécution pour exécuter les modèles de flux d'événements³⁴ par rapport à un ou plusieurs flux de données. Cette technique permet d'analyser les données en transit et bien avant qu'elles soient stockées, elle inclut la manipulation, normalisation, agrégation et filtrage de données ainsi que la détection et évaluation des patterns³⁵.

Le SAS se compose de trois éléments clés comme le montre la Figure 3.17 :

- **Interface de publication** : elle assure la connexion en temps réel entre le SAS et les flux de données en lisant ces derniers, qui viennent souvent de différentes sources, et en les transportant au processeur de SAS.
- **Serveur SAS Event Stream Processing** : il exécute les modèles de flux d'événement, qui spécifient, entre autres, les requêtes et opérations devant être appliquées sur les données (comme la transformation et analyse des données) et sous quel format elles vont être exprimées. Le SAS offre des modèles prédéfinis comme il offre à l'utilisateur la possibilité de développer et implémenter ses propres modèles en utilisant l'interface web appelée *SAS Event Stream Processing Studio*.
- **Interface d'abonné** : elle partage les résultats, issus de l'étape précédente, avec d'autres systèmes ou consommateurs d'information.

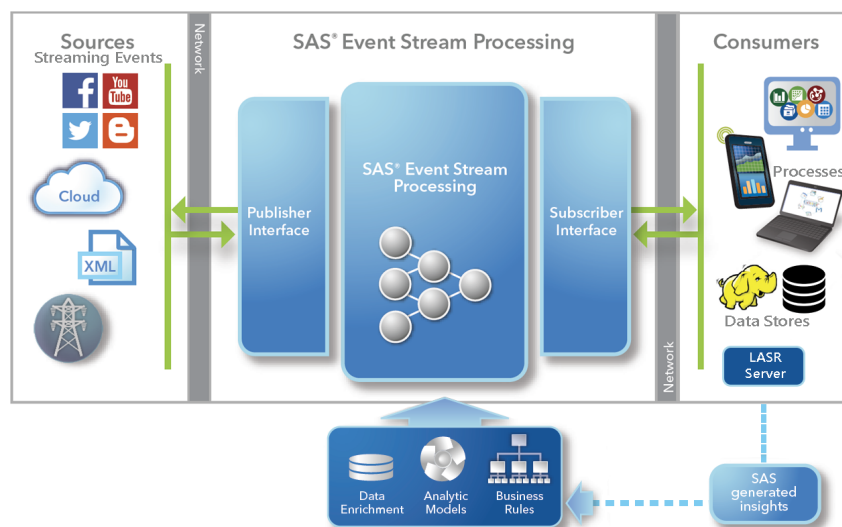


FIGURE 3.17 – Architecture de SAS Event Stream Processing (adapté de [SAS, 2015])

Le SAS Event Stream Processing permet aux entreprises de :

- Examiner, en temps réel, les événements qui se produisent dans les entreprises pour des fins de supervision ou d'audit,
- Améliorer la réactivité, la flexibilité et l'efficacité organisationnelles en agissant immédiatement dès la production des événements,
- Créer plus de valeur en prenant les bonnes décisions qui permettent, entre autres, d'améliorer l'expérience des clients, de réduire les coûts et les risques.

3.3.3 Autres outils supportant la prise de décision

Le marché des logiciels est très riche en outils commercialisés supportant la prise de décision. Plusieurs forums et sites de comparaison des logiciels, comme Capterra³⁶, proposent un classement de ces outils

34. Un événement est défini, selon le SAS [2015], comme étant quelque chose qui se produit à un temps identifiable et qui peut être capturée.

35. C'est une structure ou un motif fréquemment rencontré lors de l'étude des données. Ce concept est détaillé dans les chapitres suivants

36. <https://www.capterra.com/decision-support-software/>

selon plusieurs critères comme le déploiement (basé sur le web ou installé sur la machine), caractéristique (analyse de données ou de performance, outil basé sur la simulation ou sur les arbres de décisions, etc.), note des utilisateurs du logiciel, etc. Nous citons ci-dessous quelques outils que nous supposons être les plus utilisés et les plus innovants :

- **PowerCurve Decision Management**³⁷ : C'est un outil conçu par la société *Experian*. Il permet d'aider les entreprises à adopter la stratégie de gestion de décision la plus efficace, tout au long du cycle de vie du client, à travers ses deux principaux composants : le premier étant dédié à la découverte des règles métier et l'autre à l'analyse prédictive. Cet outil supporte les entreprises lors de la prise des décisions comme l'identification et la quantification des opportunités, supervision et gestion des relations avec les clients, détermination des fraudes et des prix initiaux des nouveaux produits, etc. Cet outil est caractérisé par sa base de données très riche, contenant les données venant de plusieurs sources (internes et externes à l'entreprise) offrant ainsi un moteur de décision capable de prédire plusieurs stratégies et de les tester et les comparer au modèle de décision considéré et exécuter les simulations "What-if" (connues sous le nom *what-if analysis*) à partir des données historiques.
- **TIBCO Systems of Insight**³⁸ : C'est une suite d'outils destinée aux entreprises pour améliorer leurs activités en permettant, entre autres, la détection des fraudes, traçabilité et suivie des opérations logistiques, et l'analyse des clients. Elle englobe à la fois les techniques de l'analyse prédictive des données et celles de l'analyse des flux de données en temps réel (Streaming analytics introduits dans la Section 3.3.2). Il permet de réduire le temps allant de l'analyse des données à la génération des connaissances grâce à ses ressources riches, allant des modèles de solutions prédéfinis, aux astuces proposées par la communauté des utilisateurs de ce logiciel. Ce dernier a contribué à la réussite de Mercedes-AMG Petronas Motorsport, équipe du constructeur Mercedes-Benz, classé le premier³⁹ mondial en remportant le championnat du monde de Formule 1 avec 51 victoires sur un total de 59 courses au cours de ces trois dernières saisons.
- **Microsoft Excel**⁴⁰ : Cet outil très populaire permet de supporter la prise de décisions grâce à ses fonctions comme la fonction logique *if*. Il existe aussi plusieurs macros⁴¹ et plug-in (comme T4, 3rd party, etc) qui viennent s'ajouter pour étendre les fonctionnalités de l'outil, en introduisant de nouvelles fonctions comme les fonctions *Goal Seek* (permettant de trouver une valeur inconnue qui permet de produire un résultat souhaité), *Solver* (permettant de calculer la solution optimale en se basant sur plusieurs variables et contraintes), *Scenario Manager* (permettant de créer et évaluer une collection de scénarios *What-if* contenant plusieurs valeurs d'entrée), *simulations de Monte Carlo*, *arbres de décisions*, *prévision* (en utilisant le Data Analysis toolpack), *régression* (pour estimer la relation entre les variables), etc. De plus, il existe aussi plusieurs solveurs⁴², comme ceux de Frontline solvers⁴³, permettant de résoudre différents problèmes sur Excel, comme l'optimisation sans incertitudes (en utilisant Analytic Solver - Optimization), ou encore le problème de visualisation de données, fouille de texte, les analyses prédictives pour les bases de données géantes (en utilisant le Analytic Solver - Data Mining).

3.3.4 Synthèse et discussion de l'état de l'art commercial

Dans cette troisième et dernière partie de l'état de l'art, nous avons présenté quelques outils commercialisés, permettant de supporter la prise de décision. La comparaison de ces outils par rapport aux caractéristiques que nous cherchons est présentée dans le Tableau 3.7, dans lequel nous voyons clairement que les outils commerciaux répondent à un nombre de caractéristiques plus intéressant que les outils proposés dans le

37. <http://www.experian.com/strategy-management/powercurve.html>

38. <https://www.tibco.com/solutions/systems-of-insight>

39. Classement des constructeurs de Formule 1 : <https://www.lequipe.fr/Formule1/f1-classement-ecuries.html>

40. <https://products.office.com/en/excel>

41. C'est un outil intégré dans Excel, qui se voit comme un programme codé en utilisant le langage de programmation VBA (Visual Basic for Applications), visant à automatiser un ensemble de décisions en se basant sur certaines conditions

42. C'est une partie de logiciel permettant de résoudre un problème mathématique donné.

43. <https://www.solver.com/products-overview>

milieu académique. De plus, l'outil qui permet de répondre au maximum de caractéristiques est le Blueworks Live d'IBM. Les principaux inconvénients de cet outil est que, malgré le fait qu'il permette la modélisation et la traçabilité des processus ainsi que l'aide à la décision relative au processus et l'identification des changements, il n'a pas la capacité d'assurer l'aide à la décision au niveau des activités (en proposant les paramètres des activités les plus pertinents) et la **propagation** des changements contextuels identifiés, en plus d'être payant et n'ayant pas la capacité d'apprendre pour inférer (comme le mentionne notre hypothèse).

	Blueworks Live d'IBM	SAS Event Stream Processing	Power Curve Decision Management	TIBCO Systems of Insight	Microsoft Excel
Modéliser et visualiser le processus	✓	-	-	-	-
Tracer le processus	✓	-	-	-	✓
Aider à décider	✓	✓	✓	✓	✓
Identifier et propager les changements contextuels	~	-	-		~
Open source	-	-	-	-	-
Modulaire d'architecture					
Générique					
Indépendant	✓				
Semi-Autonome \ Intelligent	~	~	~	~	~
Capacité de support	~	~	~	~	~
Facile	✓	✓	✓	✓	✓
Extensible					
Maintenable					
Portable					
Efficace					
Fiable					
Orienté activité et processus	~	~	~	~	~

TABLE 3.7 – Comparaison des outils commercialisés présentés par rapport aux caractéristiques recherchées. "✓" (resp. "-") veut dire que l'outil répond parfaitement (resp. ne répond pas) à l'exigence, "~" veut dire qu'il répond en partie à l'exigence et la case vide veut dire qu'il n'a pas été testé.

3.4 Synthèse et conclusion de l'état de l'art académique et commercial

Nous venons de choisir la classe des techniques que nous considérons les plus appropriées pour répondre à nos objectifs, après avoir identifié la théorie et l'approche les plus adaptées à notre contexte (en suivant la logique tracée dans la Figure 3.1). De plus, il se trouve que le système qui répond au maximum de caractéristiques est le système d'aide à la décision et que aucun des processus de prise de décision ne nous semble capable de permettre la réalisation de nos objectifs principaux. De plus, aucun des outils commercialisés ne répond parfaitement à nos exigences. Ainsi, et avant de prononcer le verdict, nous souhaitons étendre notre champ de vision en explorant quelques pratiques et processus adoptés dans le milieu industriel. Ceci fera l'objet du chapitre suivant.

4

CHAPITRE

État de l'art général : processus de prise de décision dans les milieux industriels

"The only proven way to raise your odds of making a good decision is to learn to use a good decision-making process - one that can get you the best solution with a minimal loss of time, energy, money, and composure."

John Hammond

4.1	Dans les industries aéronautiques et spatiales	63
4.1.1	Prise de décision lors de la gestion de risque à la NASA	63
4.1.2	Prise de décision lors de la conception des systèmes à la NASA	66
4.1.3	Prise de décision pendant le vol (lors de la supervision)	68
4.2	Dans les industries nucléaires et de l'énergie	70
4.2.1	Prise de décision à l'U.S.NRC	70
4.2.2	Prise de décision à Schneider Electric	73
4.3	Selon quelques organisations et associations contribuant à la prise de décision	75
4.3.1	L'INCOSE	75
4.3.2	Le BKCASE	76
4.3.3	L'OMG	78
4.3.4	L'IIBA	81
4.3.5	L'ISO	82
4.4	Synthèse et discussion de l'état de l'art industriel	83
4.5	Prononciation du verdict : adopter une solution existante ou concevoir une autre ?	83

La réussite d'un produit n'est pas le seul facteur permettant de créer la richesse d'une entreprise. En effet, les dépenses engagées au cours du développement de produit, pouvant avoir un impact important sur le profit de l'entreprise, varient en fonction de plusieurs paramètres comme le choix de la meilleure alternative⁴⁴, entre autres. Ces paramètres peuvent être maîtrisés en suivant un processus décisionnel, où sont explicitement décrites les activités nécessaires à exécuter pour générer la décision la plus satisfaisante possible dans un contexte donné.

Dans le chapitre précédent, nous avons présenté comment est prise la décision dans les milieux académiques et commerciaux, nous avons identifié les techniques et la nature des systèmes d'aide à la décision qui sont les plus adaptés à notre contexte. De plus, nous avons conclu qu'aucun des processus de prise de décision ne

44. En effet, la sélection d'une mauvaise alternative aura un impact majeur sur la qualité du produit, le temps et le coût de son développement, etc.

répond à nos besoins. Ainsi, et avant de chercher à proposer notre propre processus de prise de décision, il est intéressant de découvrir les pratiques et les processus adoptés dans les milieux industriels. Par conséquent, nous allons étudier, dans le présent chapitre, les processus adoptés dans les domaines aéronautiques, spatiales, nucléaires et de l'énergie, que nous supposons être les plus critiques. Nous présentons, ensuite, les différents processus de prise de décision vus par quelques organisations caractérisées par leurs contributions actives dans le domaine de la prise de décision. Le chapitre est ensuite clôturé par une synthèse où est discutée la possibilité d'adopter un des processus étudiés pour répondre à notre question de recherche, avant de prononcer le verdict : ré-utiliser une solution existante ou concevoir une autre pour répondre à nos objectifs.

4.1 Dans les industries aéronautiques et spatiales

4.1.1 Prise de décision lors de la gestion de risque à la NASA

L'administration nationale de l'aéronautique et de l'espace⁴⁵ (NASA) a connu trois accidents mortels dont la cause principale est la prise d'une "mauvaise décision". En début de l'année 1967, la mission AS-204 - qui devait constituer le 4ème vol du programme Apollo - n'a jamais eu lieu parce qu'un incendie s'est déclenché pendant un essai au sol provoquant la mort de trois astronautes. Suite à cette catastrophe, la NASA a convoqué le *Apollo 204 Accident Review Board* pour déterminer la cause de l'incendie⁴⁶. Il s'est avéré que le problème était dû au *mauvais choix* des matériaux lors de la *conception* (utilisation des matériaux inflammables) et à une *mauvaise supervision* du câblage électrique du circuit de refroidissement (présence d'un fil électrique dénudé).

En début de l'année 1986, la NASA a vécu une deuxième catastrophe : celle de l'explosion de la navette spatiale *Challenger* (lors de la mission STS-51-L), 75 secondes après son décollage, provoquant la mort de sept astronautes. Les rapports d'enquête⁴⁷ ont montré que l'incident était dû à un *défaut de conception* (perte d'étanchéité d'un joint torique à cause du froid) ainsi qu'à la *culture de l'entreprise* (définie dans la Section 2.1.2) et à son *processus décisionnel* à la suite duquel la navette a été lancée bien que le comportement des joints au froid soit connu et sous-estimé. En effet, la navette a été certifiée prête pour le vol, deux semaines avant l'accident, par les participants à la réunion "Flight Readiness Review (FRR)". Ces derniers n'ont pas pensé à prévoir les phénomènes météo et la décision a été faite bien que les spécialistes de Morton Thiokol⁴⁸, le fabricant des moteurs, n'aient pas recommandé le lancement de la navette pendant les périodes à basse température. Le processus décisionnel qui a mené à cette tragédie est discuté, plus en détails, par Morel [2014].

17 ans plus tard, en début 2003, la NASA a connu un troisième drame : la destruction de la navette spatiale *Columbia* (mission STS-107), au cours de son retour sur terre lors de sa rentrée dans l'atmosphère provoquant la mort de sept astronautes. L'accident était dû au détachement de la mousse de protection du réservoir suite à sa collision avec le bord d'attaque de l'aile gauche de la navette⁴⁹. Bien que ce détachement ait lieu 82 secondes après décollage, aucune décision n'a été prise de la part de la NASA parce que l'incident avait été considéré comme *non risqué*. Michael D. Griffin⁵⁰, physicien américain et administrateur à la NASA, a affirmé pendant une interview que la situation la plus difficile qu'il a vécu pendant toute sa carrière était la perte de crédibilité de la NASA suite au résultat de l'investigation qui a trouvé de grosses similarités entre les processus décisionnels qui ont causé la destruction des navettes Challenger et Columbia [NASA, 2010a].

À la suite de ces accidents et pour restaurer sa crédibilité, la NASA a apporté plusieurs modifications à la méthode d'ingénierie de ses systèmes, notamment le passage de la gestion de sûreté basée sur le risque (qui peut être définie comme étant le processus dans lequel les ressources sont allouées de façon à ce que le risque prévu soit minimisé [Farhang Mehr and Tumer, 2006]) à la gestion de sûreté tenant compte du risque (qui est définie comme un processus délibératif dans lequel les informations sur le risque sont formellement utilisées comme une partie d'un processus de prise de décision intégré [Zio and Pedroni, 2012; Dezfuli et al., 2010]). L'ancien processus de gestion de risque (*Risk Management - RM en Anglais*) à la NASA, se basait uniquement sur l'approche de la gestion de risque continue (*Continuous Risk Management - CRM en Anglais*) comme présenté dans la Figure 4.1.a [Dezfuli et al., 2010]. Le CRM est définie par Alberts et al. [1996] comme étant une approche qui aide l'organisation à déterminer et évaluer - en continu - les risques les plus importants, et à implémenter les stratégies qui permettent de les traiter. C'est le processus qui a pour but d'assurer que l'implémentation de l'alternative sélectionnée répond aux exigences [Dezfuli et al., 2010],

45. <https://www.nasa.gov/>

46. <https://history.nasa.gov/Apollo204/content.html>

47. <https://history.nasa.gov/rogersrep/genindex.htm>

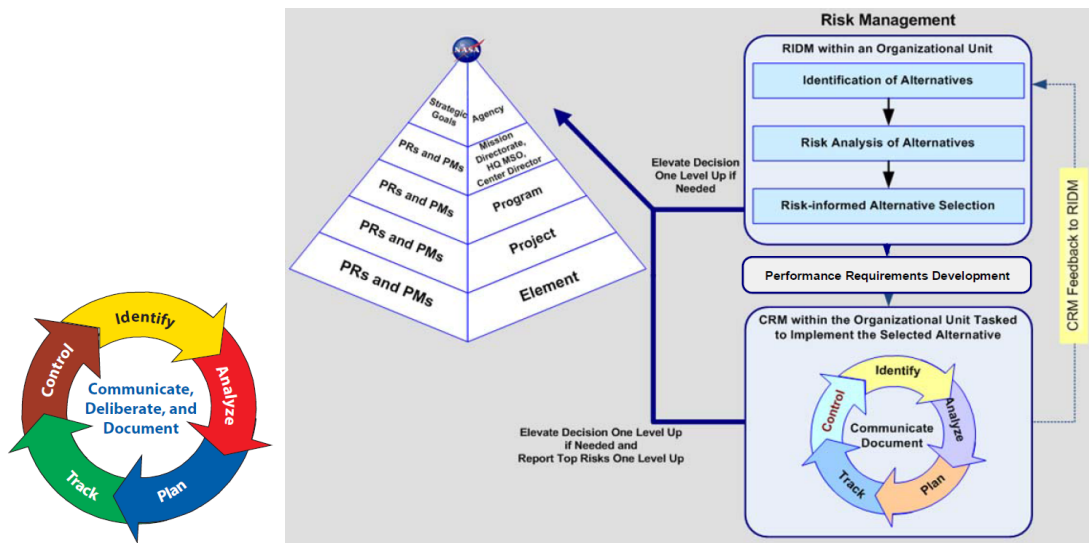
48. <https://en.wikipedia.org/wiki/Thiokol>

49. <https://history.nasa.gov/columbia/CAIB.html>

50. https://en.wikipedia.org/wiki/Michael_D._Griffin

il consiste à [NASA, 2010b] :

1. **Identifier** les risques par l'identification des scénarios ayant des conséquences défavorables (qui ont abouti à des résultats autres que ceux attendus),
2. **Analyser** et estimer la probabilité et les conséquences du risque et la période dans laquelle l'atténuation du risque doit être effectuée,
3. **Planifier** le suivi et le contrôle des actions correctives,
4. **Suivre** les données relatives aux paramètres de performance techniques,
5. **Contrôler** le risque émergent en exécutant l'action de contrôle appropriée et vérifier son efficacité,
6. **Communiquer** toutes les informations du risque et **délibérer** des décisions durant l'exécution du processus. **Documenter** le risque, le plan de contrôle du risque, etc.



4.1.a. Ancien RM : CRM [NASA, 2010b]

4.1.b. Nouveau RM : CRM + RIDM [Dezfuli et al., 2010]

FIGURE 4.1 – Évolution du processus de gestion de risque à la NASA

En fin 2008, le processus de RM a été revisité et modifié. Il se compose dorénavant de deux composants complémentaires (Figure 4.1.b) : le CRM (déjà défini dans le paragraphe précédent) et le processus de prise de décisions tenant compte du risque (*Risk-Informed Decision Making - RIDM* en Anglais). Le RIDM est appelé pour choisir la décision qui permet de réaliser les objectifs et le CRM est appliqué pour gérer les risques associés à l'implémentation de cette décision.

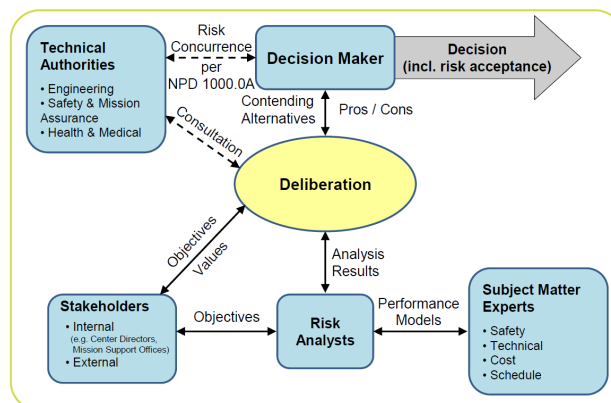


FIGURE 4.2 – Flux d'information pendant l'exécution du processus RIDM [Dezfuli et al., 2010]

Le RIDM s'intéresse à la sélection des décisions tenant compte de leurs risques pour assurer une réalisation effective des objectifs [Dezfuli et al., 2010], en d'autres mots, il produit une base justifiable pour la prise de décisions et aide à identifier les plus grands risques en priorisant les efforts pour les minimiser ou les éliminer. Le RIDM reconnaît que les informations techniques ne constituent pas les seules données d'entrée pour le processus de prise de décision et que le jugement humain joue un rôle important dans ce processus. En effet, pendant l'exécution de ce processus, plusieurs personnes (parties prenantes, analyste des risques, spécialistes du domaine, autorités techniques et le décideur) interagissent afin d'assurer que les objectifs, valeurs et connaissances sont intégrées lors de la prise de décision (Figure 4.2). Il est impératif que l'analyste - qui conduit l'analyse de risque des alternatives - tienne compte des objectifs de tous les participants. L'analyse de risque complète doit ensuite être délibérée avant que le décideur sélectionne (avec l'accord de l'autorité technique appropriée) l'alternative qui va être implémentée. L'exécution du processus RIDM passe par trois étapes comme le montre la Figure 4.1.b :

Étape 1 : identification des alternatives Dans cette étape, les objectifs - qui sont souvent qualitatifs et multidimensionnels - sont décomposés en donnant lieu à des objectifs dérivés dont chacun reflète un problème individuel significatif. Au plus bas niveau de cette décomposition se trouvent les objectifs de performance dont chacun est associé à une mesure de performance qui quantifie le degré de son atteinte par l'alternative. Les objectifs dont les mesures de performances doivent osciller à l'intérieur de limites définies, donnent lieu aux contraintes imposées (reflétant ces limites et/ou les exigences de performance comme présenté dans la Figure 4.1.b). Les objectifs ainsi que les contraintes sont les paramètres autour desquels les alternatives sont évaluées, et les mesures de performances sont les paramètres qui permettent de quantifier la capacité de ces alternatives à vérifier les contraintes imposées et à répondre aux objectifs.

Étape 2 : analyse de risque des alternatives Dans cette étape, les mesures de performance de chaque alternative sont quantifiées en tenant compte des incertitudes qui tiennent entre la sélection d'une alternative et la réalisation des objectifs. Par conséquent, le résultat d'une alternative ne sera qu'un élément de l'ensemble des résultats prévus en fonction de l'occurrence, absence ou qualité d'occurrence des événements intervenants (incertitudes). Le processus RIDM est exécuté d'une manière progressive, la rigueur de l'analyse doit, en effet, être proportionnelle à la complexité du contexte de la décision. L'analyste doit exécuter le RIDM à un niveau suffisant pour supporter la sélection robuste d'une alternative. Si l'incertitude sur un ou plusieurs mesures de performance empêche le décideur d'évaluer avec confiance les différences importantes entre les alternatives, l'analyse du risque doit être itérée pour réduire l'incertitude.

À l'issue de cette partie, est produit le document technique pour la délibération (*Technical Basis for Deliberation - TBfD* en anglais), qui documente l'ensemble des alternatives, résume les méthodologies d'analyse utilisées pour quantifier les mesures de performance et présente les résultats. Ce document constitue l'entrée qui assure que les délibérations qui supportent la prise de décision (étape suivante) tiennent compte des risques.

Étape 3 : sélection de l'alternative tenant compte du risque Dans cette étape, la délibération a lieu entre les parties prenantes et le décideur. Ce dernier exécute une des trois tâches suivantes (selon le résultat de la délibération) : sélectionner une alternative pour l'implémentation ou éliminer un ensemble d'alternatives et demander un examen plus approfondi des alternatives restantes (retour à l'étape 2) ou demander de nouvelles alternatives (retour à l'étape 1). La justification de la sélection d'une alternative est documentée dans un "rapport de sélection tenant compte du risque" (*Risk-Informed Selection Report - RISR en anglais*).

D'après les retours d'expériences à la NASA, il s'est avéré que le nouveau RM a permis de [Dezfuli et al., 2010] :

- Favoriser la gestion proactive des risques,
- Prendre de bonnes décisions suite à la bonne utilisation de l'information sur les risques,
- Gérer, d'une façon efficace, les risques liés à l'implémentation en concentrant le processus CRM sur les exigences de performance générées par le processus RIDM (lors de l'étape 1).

4.1.2 Prise de décision lors de la conception des systèmes à la NASA

Les systèmes sont conçus à la NASA en suivant un processus fortement itératif et récursif (Figure 4.3). La première étape est la collection et l'analyse des exigences (de haut niveau) et attentes des parties prenantes comme les objectifs des systèmes, les critères pour définir leur réussite, les contraintes, etc. Ces derniers constituent l'entrée de la boucle de conception itérative (Iterative Design Loop dans la Figure 4.3) dans laquelle sont développées les trois produits suivants qui doivent être cohérents entre eux : un premier design du système, les concepts d'opérations et les exigences dérivées. L'atteinte de cette cohérence nécessite des itérations au niveau de la boucle de conception ainsi que la prise de décisions de conception. Une fois la cohérence est atteinte, l'équipe projet analyse et valide (ou pas) le design par rapport aux attentes des parties prenantes. Au cours du processus de validation sont posées des questions du type : Est ce que le système conçu fonctionne comme prévu? Est ce qu'il est sûr et fiable? Est ce qu'il est réalisable en respectant le budget imparti et le temps prévu? Si la réponse d'une de ces questions est "non" alors le design ou les attentes des parties prenantes doivent être modifiés et le processus doit recommencer en tenant compte de ces changements. Ce processus continue jusqu'à ce que le design du système, les concepts d'opérations et les exigences respectent les attentes des parties prenantes [NASA, 2010b].

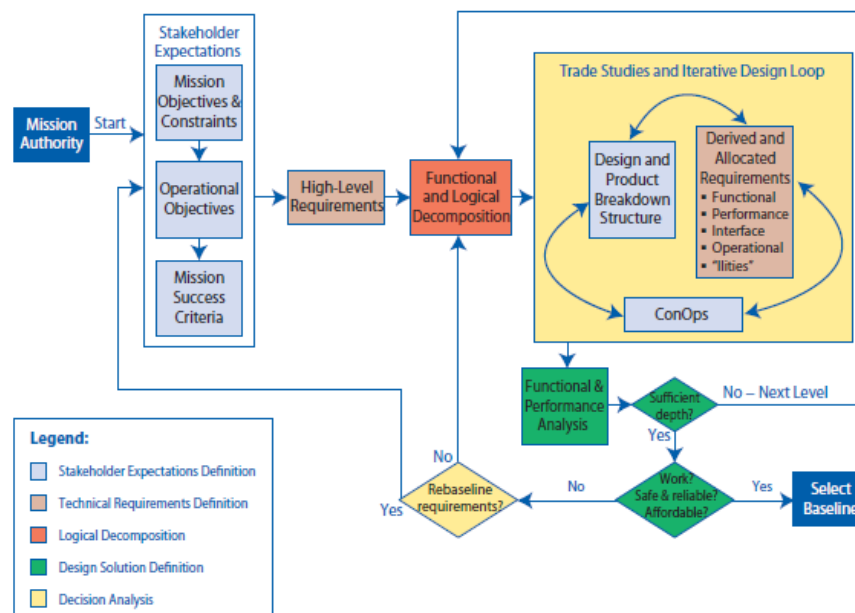


FIGURE 4.3 – Processus de conception des systèmes à la NASA [NASA, 2010b]

Le processus d'analyse de décision présenté dans la Figure 4.4, permet d'évaluer les enjeux techniques, les alternatives et leurs incertitudes afin de supporter la prise de décision lors du processus de conception des systèmes décrit dans le paragraphe précédent. Il nécessite comme entrée NASA [2010b] :

- Le **problème**, le **besoin de décision**, les **alternatives identifiées** (qui font l'objet du paragraphe suivant) et toute autre donnée qui permettra de supporter la prise de décision,
- La **demande de support** qui permet de faire appel à l'exécution de ce processus,
- Les **objectifs** et les **contraintes** à la base desquelles les alternatives sont évaluées.

La NASA assume que la prise de bonnes décisions commence par l'identification rigoureuse des alternatives. En effet, si ces dernières sont rigoureusement définies et bien comprises, le décideur peut facilement choisir une ou plusieurs alternatives avec un certain degré de certitude. Pour atteindre cet objectif, la NASA assume qu'il est nécessaire de creuser profondément l'espace des solutions possibles [NASA, 2010b], et introduit le processus de raffinements successifs présenté dans la Figure 4.5. Durant le premier tour du processus, les approches ou stratégies générales et parfois les concepts architecturaux sont étudiés. Au niveau du tour suivant, l'aspect fonctionnel est étudié, puis vient la conception détaillée et ainsi de suite.

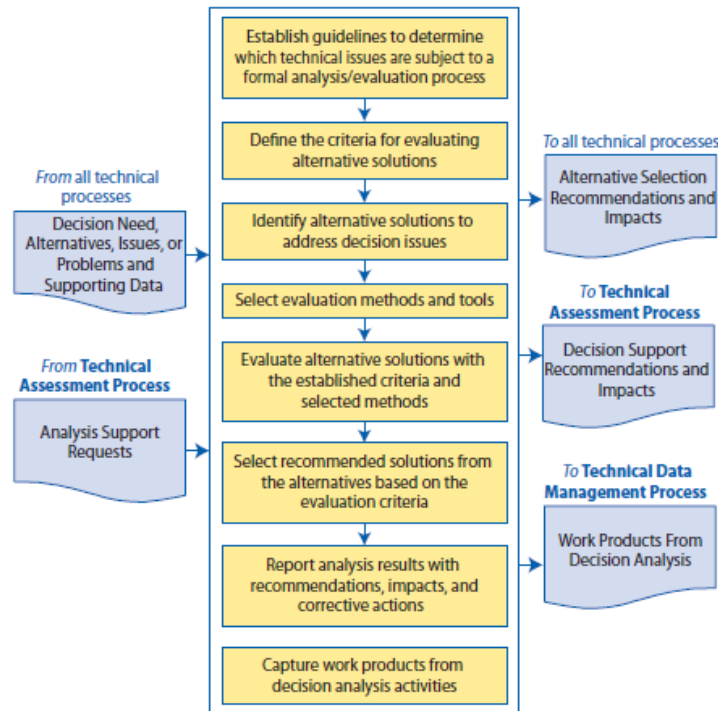


FIGURE 4.4 – Processus d’analyse de décisions à la NASA [NASA, 2010b]

Le but de ce processus est d’éviter de donner une importance prématurée à une ou plusieurs alternatives et d’assurer que ces dernières, qui seront évaluées par la suite, prennent en compte toutes les exigences [NASA, 2010b]. En effet, si on prend comme exemple le processus de choix de la date de lancement d’une navette spatiale, il n’est évidemment pas logique d’inclure les jours froids dans l’espace des alternatives si la technologie utilisée (joints toriques dans le cas de la navette Challenger) n’est pas testée (ou ne fonctionne pas) à basses températures.

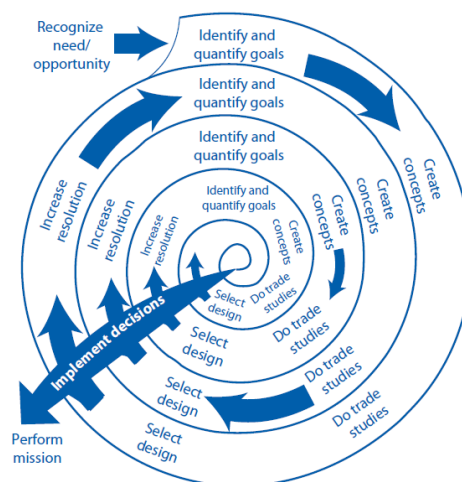


FIGURE 4.5 – Processus de raffinements successifs [NASA, 2010b]

Une fois les alternatives identifiées, les activités du processus d’analyse des décisions (Figure 4.4) sont exécutées :

1. **Établir les directives qui déterminent quelles sont les problèmes techniques qui font l’objet d’une analyse formelle ou d’un processus d’évaluation** : cette activité détermine quand est ce qu’il serait possible d’utiliser une procédure de prise de décision formelle, les concepts qui doivent être documentés, le décideur et ses responsabilités et comment seront traitées les décisions qui ne

nécessitent pas de procédure d'évaluation formelle.

2. **Définir les critères à base desquels les solutions alternatives seront évaluées** : Cette activité détermine le type des critères à considérer (attentes et exigences des clients, limites de la technologie, sûreté, etc.), leurs intervalles de valeurs acceptées, et leur classement par ordre d'importance.
3. **Identifier les solutions alternatives pour traiter les problèmes de décisions** : Cette activité consiste à enrichir l'espace des alternatives en cherchant des alternatives de solutions en plus de celles issues du problème de conception. Ceci consiste à faire un brainstorming des options de décision, et des recherches dans la littérature. Les options de décision, même les mauvaises, sont documentées.
4. **Sélectionner les méthodes et les outils de l'évaluation** : Dans cette activité, les techniques et outils de l'analyse sont sélectionnés en fonction du but de l'analyse de décision et de la disponibilité des informations supportant ces méthodes et/ou outils. Les méthodes d'analyse peuvent inclure les simulations, les matrices de compromis pondérés [Parnell, 2016], les matrices de décision, l'ingénierie, la fabrication, les tests, etc.
5. **Évaluer les alternatives avec les critères établis et les méthodes sélectionnées** : Dans cette activité, sont évaluées les hypothèses liées aux critères d'évaluation ainsi que les preuves qui les supportent, et l'impact des incertitudes sur l'analyse.
6. **Sélectionner les solutions recommandées en se basant sur les critères d'évaluation** : Dans cette activité, les recommandations sont justifiées en documentant l'information, les hypothèses et les limitations des méthodes d'évaluation utilisées.
7. **Rapporter l'analyse, l'évaluation des résultats, les conclusions, les impacts et les actions correctives** : Dans cette activité, l'équipe technique fait des propositions au décideur en documentant toutes les étapes allant de l'analyse aux conclusions.
8. **Capturer les produits issus du processus d'analyse de décision** : Dans cette activité sont capturées : les directives de l'analyse de décisions générée, la stratégie et les procédures utilisées, l'approche d'analyse, les critères et les méthodes et/ou outils utilisés, les résultats de l'analyse, les hypothèses, les incertitudes sur les actions recommandées et les actions correctives, les leçons acquises et les recommandations pour améliorer les futures analyses de décision.

À l'issue de cette analyse, les recommandations qui concernent la sélection des alternatives et leurs impacts sont proposés aux équipes du processus technique, les recommandations qui supportent l'aide à la décision et leurs impacts sont proposées aux équipes du processus d'évaluation technique, et tous les produits issus du processus d'analyse sont envoyés aux équipes du processus de gestion des données techniques.

4.1.3 Prise de décision pendant le vol (lors de la supervision)

Pendant son vol du 15 janvier 2009, l'avion Airbus A320 a été endommagée⁵¹ immédiatement après son décollage suite à une collision en vol avec un oiseau, qui a causé l'arrêt des deux turboréacteurs de l'avion sur le fleuve Hudson à New York. Le pilote Chesley Burnett Sullenberger⁵² a décidé de faire amerrir l'avion sur le fleuve. Cet événement a fait la une de la presse pour deux raisons. La première est que l'amerrissage est une tâche complexe parce qu'elle n'est pas tout à fait simulée car le simulateur ne reproduit pas fidèlement le contact de l'avion avec l'eau. La deuxième est que le pilote a pris la bonne décision au bon moment durant le vol. Cette prise de décision pendant le vol est connue sous le nom de prise de décision aéronautique (*Aeronautical decision-making* - ADM en Anglais). Elle est définie par la Federal Aviation Administration⁵³ (FAA) comme étant une approche systématique du processus mental utilisé par les pilotes pour déterminer le meilleur plan d'action pour répondre à un ensemble de circonstances donné [FAA, 2013].

Malgré les progrès qu'ont connus les systèmes aéronautiques, les techniques d'automatisation et les autres approches supportant l'amélioration de la sécurité des vols ; la prise de décision humaine demeure un enjeu important pour la sécurité des vols [Liu et al., 2013; EHEST, 2012; Hooey and Foyle, 2006; Orasanu et al.,

51. https://en.wikipedia.org/wiki/US_Airways_Flight_1549

52. Ancien pilote américain et expert en sécurité aérienne : https://en.wikipedia.org/wiki/Chesley_Sullenberger

53. Agence chargée des réglementations de tous les aspects de l'aviation civile aux États-Unis : <https://www.faa.gov/>

1998]. La FAA [2013] a lancé de nombreux programmes visant à réduire ou éliminer les facteurs plausibles (dont quelques uns sont présentés dans le rapport de la NTSB⁵⁴ [NTSB, 2000]) qui contribuent aux erreurs de décisions des pilotes, parmi ces programmes nous citons la formation à la prise de décision qui vise à réduire les erreurs des pilotes. Hooey and Foyle [2006] ont montré, en effet, que 42% des accidents sont dûs à de mauvaises décisions. Adams et al. [1993] ont montré que le taux des accidents a été réduit d'environ 50% pour les groupes de pilotes ayant suivi des séances de formation à la prise de décision. Hooey and Foyle [2006] ont montré que 23% des accidents sont dûs à des informations erronées, incomplètes ou ambiguës, ainsi, en plus des formations à la prise de décision, la conception et l'implémentation des outils d'affichages d'information et des interfaces homme-machine sont également importantes puisque qu'ils permettent la représentation complète et non ambiguë de l'information et supportent ainsi le processus ADM. Une autre notion est cruciale pour le processus ADM et donc pour la sécurité des vols, il s'agit de la supervision [Harris and Li, 2017]. Avant le vol, l'entreprise est amenée à vérifier que tout est dans l'ordre (avion bien maintenu) et pendant le vol le pilote est amené à s'assurer que tout fonctionne bien et doit être pro-actif et réactif (ces deux notions sont définies dans la Section 2.1.2). La Figure 4.6 montre qu'un mauvais processus organisationnel, faisant référence aux décisions et règles qui gouvernent les activités quotidiennes d'une entreprise comme l'établissement des procédures opérationnelles et méthodes formelles pour la maintenance et la supervision, a une influence sur le processus de supervision qui à son tour peut conduire à des décisions inappropriées durant le vol conduisant parfois à des accidents mortels. L'évaluation des performances des pilotes ainsi que les méthodes et modèles de prise de décision sont introduits et discutés plus en détails par O'Hare [2003].

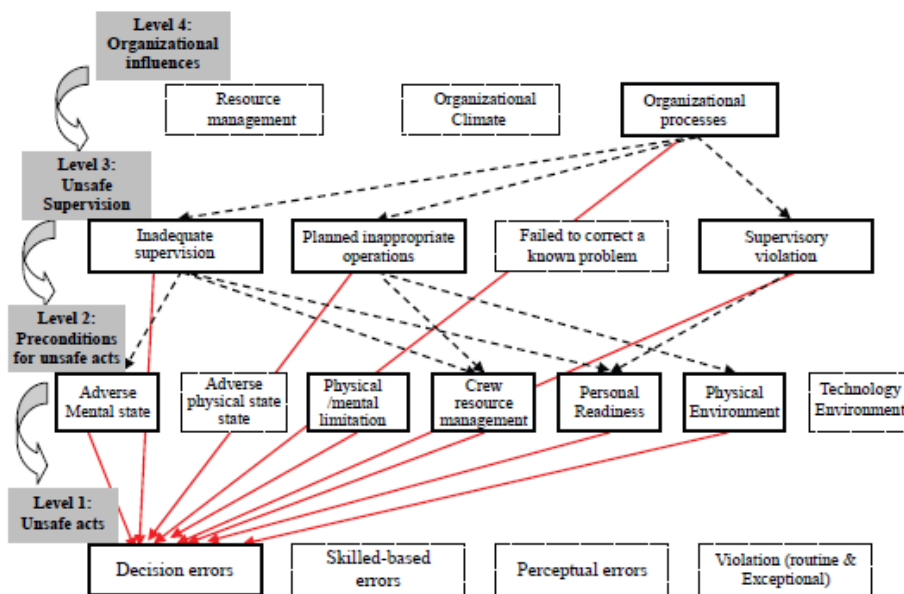


FIGURE 4.6 – Effets d'une mauvaise supervision sur la qualité des décisions [Harris and Li, 2017]

Le processus ADM aborde tous les aspects, de la prise de décision pendant le vol, qui influencent la capacité du pilote à identifier, gérer et minimiser les erreurs. Ce processus comporte, en général, les activités suivantes :

- **Définir le problème** : cette étape a lieu suite à l'occurrence d'un changement non prévu ou à la non occurrence d'un changement prévu. Ceci peut être détecté par le pilote à la suite de son expérience et/ou détecté par les systèmes aéronautiques. Cette étape est considérée comme la plus complexe du processus car le problème peut être incorrectement défini par le pilote, ou non clairement défini donnant lieu ainsi à différentes interprétations qui peuvent requérir des plans d'action différents. L'expertise du pilote et l'utilisation des connaissances et ressources appropriées sont cruciales pour assurer la bonne définition du problème.

54. National Transportation Safety Board (Conseil national de la sécurité des transports en français) est une agence responsable des enquêtes sur les accidents de transport aux États-Unis : <https://www.ntsb.gov/Pages/default.aspx>

- **Choisir le plan d'action** : Avant de choisir le plan d'action, il est nécessaire de l'évaluer. Cette évaluation peut faire appel aux manuels d'utilisation, brainstorming au sein de l'équipe, etc. pour évaluer les actions ainsi que leurs résultats.
- **Implémenter le choix** : Une fois le plan d'action est choisi, le pilote procède à son implémentation et doit maintenir la vigilance à un niveau élevé afin d'évaluer le résultat de l'action et s'assurer qu'elle produit l'effet désiré.

Il existe plusieurs modèles et méthodes mettant en pratique le processus ADM. Nous détaillons ci dessous la méthode la plus connus et citons dans le Tableau 4.1 quelques modèles et méthodes utilisés.

La méthode FORDEC : c'est une méthode de prise de décision très connue par les pilotes [Soll et al., 2016], elle a été développée par la compagnie aérienne Lufthansa⁵⁵ et le centre allemand pour l'aéronautique et l'astronautique, connu sous son acronyme allemand DLR⁵⁶. C'est une méthode adoptée par plusieurs compagnies aériennes, Air France y comprise, ainsi que par plusieurs centrales nucléaires et dans le domaine de la médecine d'urgence. Chaque lettre de l'acronyme FORDEC décrit une activité [Soll et al., 2016] :

- **Fait** : dans cette activité, le problème nécessitant une prise de décision est déterminé et confirmé, puis le bilan des faits (techniques ou pas) est réalisé. Dans le cas où plusieurs problèmes surviennent au même moment, ce qui est généralement le cas, le pilote doit bien les considérer et les traiter par ordre de priorité.
- **Options** : il s'agit ici de rassembler toutes les options possibles, sachant le problème et son contexte, parmi lesquelles le pilote décide de choisir une.
- **Risques** : Chacune des options est étudiée en établissant ses avantages et inconvénients par rapport à la sécurité du vol et des passagers.
- **Décision** : le pilote, qui est le décideur final, choisit une option et établit un plan d'action.
- **Exécution** : le scénario décrit dans le plan d'action est exécuté dans cette étape en prévenant, si le pilote le juge nécessaire, les services de contrôle aérien ainsi que les passagers.
- **Contrôle** : la décision exécutée est supervisée pour deux raisons : éviter toute mauvaise surprise au cours du vol et calculer sa performance pour la classer par rapport aux autres options non choisies.

4.2 Dans les industries nucléaires et de l'énergie

4.2.1 Prise de décision à l'U.S.NRC

La commission de réglementation nucléaire des États-Unis⁵⁷ (*U.S. Nuclear Regulatory Commission - U.S.NRC* en anglais) est l'agence qui se charge de la réglementation de la sûreté nucléaire aux États-Unis. Elle se base sur plusieurs lois, pour réguler les usages civils des matières nucléaires, dont la plus fondamentale est la loi *Act*⁵⁸ (*Atomic Energy Act*). L'implémentation de ces lois a considérablement évolué depuis la loi *Act* originale, ceci a donné lieu à de nouveaux concepts et méthodes. Un des concepts émergés est la défense en profondeur (*defence in depth* en anglais) qui se voit comme un ensemble de couches de défense multiples, indépendantes et redondantes ayant pour but la conception et l'exploitation des installations nucléaires qui empêchent ou atténuent les accidents issus du rayonnement des matières nucléaires [U.S.NRC, 2012]. Une des méthodes émergées est l'évaluation des risques (*risk assessment* en anglais) qui, conjointement avec d'autres techniques d'analyse, renseigne les décisions sur les mesures appropriées de la défense en profondeur [U.S.NRC, 2012].

Pour continuer à assurer l'utilisation sûre et fiable des matières nucléaires, l'équipe RMTF (Risk

55. <https://fr.wikipedia.org/wiki/Lufthansa>

56. <http://www.dlr.de/dlr/en/desktopdefault.aspx/tabid-10002/>

57. <https://www.nrc.gov/>

58. C'est une loi qui déclare que le développement, l'utilisation et le contrôle de l'énergie atomique doivent être orientés de manière à promouvoir la paix dans le monde, améliorer le bien être général, améliorer le niveau de vie et renforcer la libre concurrence entre les entreprises privées : <https://www.nrc.gov/about-nrc/governing-laws.html>

Acronyme	Activités	Référence
DECIDE	Detect, Estimate, Choose, Identify, Do, Evaluate	[Benner, 1975]
DESIDE	Detect, Estimate, Set safety objectives, Identify, Do, Evaluate	[Murray, 1997]
PASS	Problem identification, Acquire information, Survey strategy, Select strategy	[Maher, 1989]
SOAR	Situation, Options, Act, Repeat	[Oldaker, 1995]
SHOR	Stimuli, Hypotheses, Options, Response	[Wohl, 1981]
QPIDR	Questioning, Promoting, Ideas, Decide, Review	[Prince and Salas, 1993]
DODAR	Diagnosis, Options, Decision, Assign Tasks, Review and risk assessment	[Asbury and Jacobs, 2014]
SPORDEC	Situation Catch, Preliminary Action, Options, Rating, Decision, Execution, Controlling	[FOCA, 2016]
3Ps	Perceive, Process, Perform	[FAA, 2013]
NMATE	Navigate, Manage, Alternatives, Take Action, Evaluate	Boeing
SADIE	Share Information, Analyse Information, Develop the Best Solution, Implement your decision, Evaluate the Outcome	Emirates dans les années 90
SACoA	Situation Assessment, Course of Action	[EHEST, 2012]
OODA Loop	Observe, Orientate, Decide, Act	[EHEST, 2012]
GRADE	Gather, Review, Analyze, Decide, Evaluate	Boeing
FATE	Fly the plane, Analyse the problem, Take action, Evaluate	[Aviation-Knowledge, 2012]
SAFE	State the problem, Analyse the options, Fix the problem, Evaluate the result	[Aviation-Knowledge, 2012]

TABLE 4.1 – Quelques méthodes et outils mettant en pratique le processus ADM

Management Task Force) a été chargée de développer une approche de régulation holistique⁵⁹, tenant compte du risque et basée sur la performance. À la suite de plusieurs années de recherche, l'équipe RMTF a proposé le framework présenté dans la Figure 4.7. Il s'agit d'un flux logique allant de la définition de la mission de l'U.S.NRC (qui consiste à protéger le public et l'environnement contre les hasards possibles issus de l'usage des matériaux radioactifs) à la définition de son objectif (qui consiste à gérer le risque) à la définition du but⁶⁰ de la gestion de risque (fourni par le concept de la défense en profondeur tenant compte du risque et basée sur la performance) au processus de prise de décision pour la gestion des risques dont chacune des étapes est décrite comme suit [U.S.NRC, 2012] :

Étape 1 : identifier le problème Le problème peut être signalé par les parties prenantes, les utilisateurs ou par les unités de l'U.S.NRC. La première étape du processus de prise de décision est la clarification et la définition du problème. Ce dernier est ensuite caractérisé pour définir son cadre et ses implications potentielles en termes d'objectifs de gestion de risque (Figure 4.7).

Étape 2 : identifier les options Cette étape dépend du problème. quelques processus de l'U.S.NRC (comme l'évaluation environnementale) comprennent des directives précises pour identifier et évaluer les alternatives (l'alternative "aucune action" y comprise). D'autres processus (comme l'examen des propositions des utilisateurs) ne requièrent pas une identification des options puisque la réponse n'est autre qu'une acceptation ou un refus.

59. C'est une approche qui décrit un système comme étant un ensemble indivisible et le traite d'une manière globale non disjointe pour assurer un bon niveau de cohérence

60. Bien que les notions "objectif" et "but" sont définis de la même façon par les dictionnaires, il existe une grande différence du point de vue business. Un but est l'ensemble des déclarations décrivant les aspirations pour l'avenir d'une entreprise, tandis qu'un objectif est l'ensemble des étapes concrètes que l'entreprise doit suivre pour réaliser ses buts

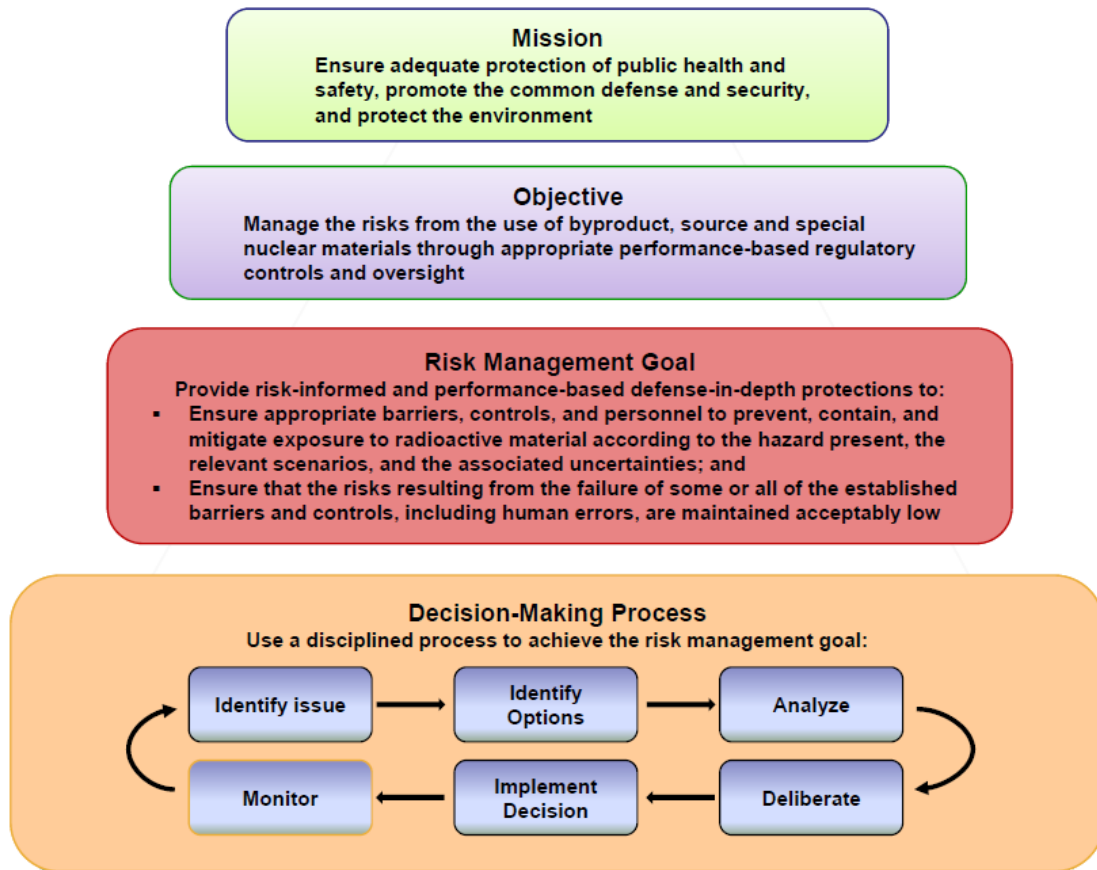


FIGURE 4.7 – Framework réglementaire de la gestion du risque à l'U.S.NRC [U.S.NRC, 2012]

Étape 3 : analyser Il s'agit, ici, d'évaluer les risques associés aux options. Plusieurs types d'analyses peuvent être réalisées : l'analyse technique assure que les événements risqués soient maintenus à un niveau acceptable, l'analyse mécanique comme les calculs thermo-hydrauliques, les évaluations probabilistes ou qualitative des risques, etc. Ces techniques supportent les décisions lors des délibérations.

Étape 4 : délibérer La délibération, aussi appelée processus de prise de décision intégré (Figure 4.8), est définie comme étant tout processus formel ou informel ayant pour but la communication et la considération collective des problèmes [Council et al., 1996]. Elle permet de formuler le problème de décision, guider l'analyse pour améliorer la compréhension des participants, expliciter la signification des résultats des analyses et des incertitudes, améliorer la capacité des participants pour participer efficacement au processus de prise de décision [Council et al., 1996]. Lors des délibérations, les analyses techniques et leur sensibilité, les hypothèses, les incertitudes, et les résultats analytiques (ainsi que le résultat de leur comparaison avec les critères de décision) sont tous considérés comme des facteurs importants influençant la prise de la décision finale. D'autres facteurs peuvent être considérés lors de la délibération, comme les ressources, le planning, les exigences légales, les normes, etc. [U.S.NRC, 2012].

Étape 5 : implémenter la décision Les décisions produites lors de la délibération déterminent si les protections assurées par le concept de défense en profondeur, et associées aux hasards radiologiques, sont adéquates ou si elles doivent être renforcées. Ces décisions sont implémentées en suivant un processus réglementaire qui comprend la préparation des règlements et conseils, la réalisation des examens environnementaux et l'exécution des programmes de surveillance.

Étape 6 : superviser Il s'agit d'une des étapes les plus importantes du processus de prise de décision. Une fois la décision est implémentée, elle doit être supervisée pour permettre à l'agence d'être réactive en identifiant les problèmes et pour mesurer l'efficacité des décisions prises ainsi que l'efficacité du

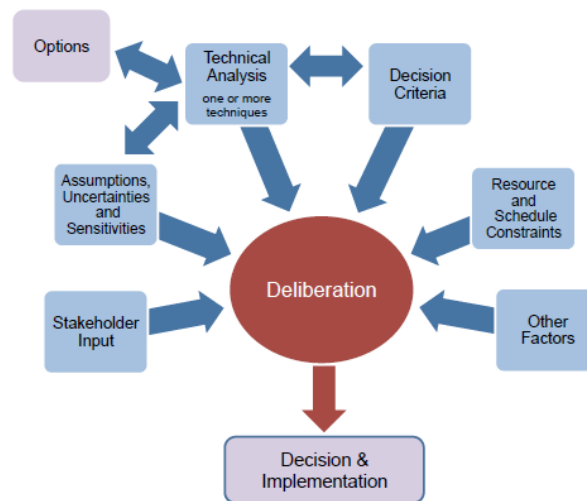


FIGURE 4.8 – Processus de délibérations (aussi appelé processus décisionnel intégré) [U.S.NRC, 2012]

framework proposé. Les outils d'évaluation comprennent les programmes d'inspection et de surveillance, les interactions avec les parties prenantes et les utilisateurs, les évaluations périodiques, etc.

Il s'est avéré que le framework proposé par l'U.S.NRC (présenté dans la Figure 4.7) a de nombreux avantages [U.S.NRC, 2012] :

- Les connaissances issues du processus d'évaluation des risques permettent d'améliorer le réalisme et la base technique des directives et réglementations,
- La cohérence de la communication interne et externe est améliorée,
- Le support lors de la résolution des problèmes est réalisé d'une manière systématique, cohérente et efficace et permet d'améliorer le processus d'allocation des ressources.

4.2.2 Prise de décision à Schneider Electric

Au cours de la dernière décennie, les bouleversements du marché et les progrès technologiques sous-jacents ont perturbé le secteur de l'électricité et ont entraîné un rééquilibrage des sources de production de l'énergie, entre autres. Ceci a conduit les entreprises à exécuter plusieurs tâches (dont trois, entre autres, sont citées) pour faire face à ces perturbations et réduire le coût de l'énergie et l'impact sur l'environnement (en réduisant l'émission du dioxyde de carbone - CO₂). La première tâche est de mettre en question les modèles commerciaux, stratégiques et réglementaires existants, qui se sont devenus incertains. La deuxième est de changer la façon comment elles achètent, vendent et utilisent l'électricité et les autres sources d'énergie. La troisième est de rechercher de nouvelles techniques et outils pour soutenir les processus de prise de décision. Ce dernier concept peut être défini, dans le domaine de l'énergie, comme étant l'ensemble des décisions prises durant le processus de conception ou de sélection des technologies et/ou des systèmes énergétiques pour réaliser un objectif ou un ensemble d'objectifs ; comme la conception d'un compresseur plus efficace, ou encore la sélection d'un type de centrale électrique pour produire une quantité donnée d'électricité à un coût minime [Simpson, 2010].

Schneider Electric⁶¹ (le groupe industriel français à portée internationale, spécialisé dans la gestion de l'électricité, dans la proposition et fabrication des solutions d'automatisation et d'autres services dans le domaine de l'énergie électrique) admet que bien que ces tâches doivent être prises en considération ensembles pour produire le maximum d'avantages, elles sont souvent gérées de manière indépendante en raison des divisions départementales, ce qui favorise l'inefficacité et permet de rater beaucoup d'opportunités [SE, 2017]. En plus de l'absence du parallélisme lors de l'exécution des tâches, Schneider Electric souligne que l'existence de données incohérentes et d'une mauvaise visibilité à

61. <https://www.schneider-electric.fr/fr/>

travers l'organisation sont également considérées comme un obstacle à la compétitivité.

À Schneider Electric, ainsi qu'à plusieurs autres entreprises leader, les fonctions d'énergie et de durabilité (desquelles relèvent les trois tâches que nous avons citées dans le paragraphe précédent) deviennent étroitement intégrées et l'entreprise commence à adopter un processus intégré de capitalisation et de prise de décision (Figure 4.9). Le processus comprend les étapes suivantes [SE, 2017] :

- **Capturer** : Dans cette étape sont tracées les données venant de plusieurs sources d'information, comme les données relatives aux factures des consommateurs et/ou producteurs, les données métier relatives à l'expertise, la performance et le marché, les données relatives à la météo, etc.
- **Combiner** : Les données capturées, issues souvent de plusieurs sources, sont combinées dans une plateforme unique et accessible, tout en les post-traitant pour les rendre homogènes et en ajoutant à chacune son contexte.
- **Valider** : Les données combinées sont ensuite évaluées en identifiant et traitant les anomalies qu'elles présentent pour améliorer leur justesse (*accuracy* en Anglais⁶²) et qualité, puis elles sont validées et mises à disposition pour les analyses.
- **Analyser et exécuter** : Dans cette étape, les données sont analysées en vue de trouver les meilleures options pour économiser de l'argent, améliorer la performance, bien se positionner dans le marché, améliorer les prévisions et décisions d'achat, etc.



FIGURE 4.9 – Processus de prise de décision à Schneider Electric [SE, 2017]

Le processus de prise de décision intégré (Figure 4.9) permet de donner à l'entreprise une vue holistique sur sa performance et lui permet aussi d'améliorer ses stratégies et ainsi innover [SE, 2017]. Il contribue fortement à la découverte des opportunités de la gestion active de l'énergie, ce concept qui se voit comme une approche puissante et efficace pour l'utilisation, basée sur la conservation, des ressources énergétiques [XIANG et al., 2015].

62. Bien que les notions de justesse (*accuracy*) et de précision (*precision*) sont souvent interchangeables, la différence sémantique est loin d'être négligeable. La justesse fait référence à la proximité d'une valeur mesurée par rapport à une valeur standard et/ou connue. En revanche, la précision est la proximité de deux mesures ou plus, les unes aux autres. Si par exemple, les mesures sont proches de la valeur standard, mais sont très loin les unes des autres ; nous pouvons dire qu'elles sont justes mais imprécises [Westgard and Lott, 1981]

4.3 Selon quelques organisations et associations contribuant à la prise de décision

4.3.1 L'INCOSE

L'International Council on Systems Engineering (INCOSE)⁶³ est une organisation ayant pour but le développement et la promotion de l'ingénierie systèmes (*Systems Engineering - SE* en Anglais), cette approche interdisciplinaire qui est définie par l'Association Française d'Ingénierie Système (AFIS)⁶⁴ comme étant "une démarche méthodologique générale qui englobe l'ensemble des activités adéquates pour concevoir, faire évoluer et vérifier un système apportant une solution économique et performante aux besoins d'un client tout en satisfaisant l'ensemble des parties prenantes". L'INCOSE affirme que "all SE activities should be conducted within the context of supporting good decision making", et considère, ainsi, le processus de gestion de décision, dans lequel une situation de décision est transformée en plan d'action, comme le plus critique de l'ingénierie systèmes [INCOSE, 2015]. Le modèle IPO (Input-Process-Output)⁶⁵ du processus de gestion de décision est présenté dans la Figure 4.10. Ce processus est constitué de trois activités :

- **Préparer pour la décision** : la situation de décision (dont quelques exemples sont décrits dans le Tableau 3.1) fait appel à cette première activité dans laquelle la stratégie de gestion de décision est d'abord identifiée. L'énoncé de décision est ensuite établi et clarifié pour éviter que la décision soit mal formulée (ce qui donnera lieu à un système autre que celui ayant été spécifié),
- **Analyser l'information de la décision** : Il s'agit ici de :
 - Cadrer et structurer la décision,
 - Développer les objectifs, générer les alternatives, et les évaluer en utilisant les techniques de l'analyse déterministe,
 - Identifier les incertitudes potentielles et évaluer leur impact,
 - Améliorer les alternatives et faire une étude de compromis,
 - Rédiger un rapport de recommandation détaillant le résultats des recherches menées et la logique derrière la prise de décision, puis identifier un plan d'implémentation.
- **Faire et gérer les décisions** : dans cette activité, la décision finale est prise et est tracée en mentionnant toutes les ressources utilisées.

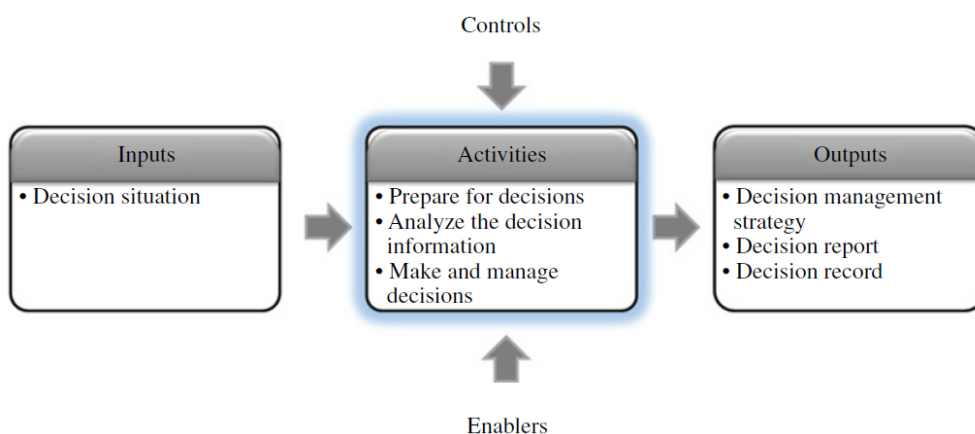


FIGURE 4.10 – Modèle IPO du processus de gestion de décision selon l'INCOSE [2015], où IPO (Input-Process-Output) est un formalisme permettant de modéliser les processus : <https://www.sixsigmadaily.com/input-output-model/>.

63. <https://www.incose.org/>

64. C'est la version Française de l'INCOSE : <http://www.afis.fr>

65. <https://www.sixsigmadaily.com/input-output-model/>

A l'issu de l'exécution du processus de gestion de décision, trois documents sont générés. Il s'agit de la stratégie de gestion de décision, du rapport de recommandation et de la trace ; générés respectivement au cours de la première, deuxième et troisième activité.

4.3.2 Le BKCASE

Le projet "Body of Knowledge and Curriculum to Advance Systems Engineering" (BKCASE)⁶⁶ qui compte trois organisations participantes : l'INCOSE⁶³ (déjà introduite), l'Institute of Electrical and Electronics Engineers Computer Society (IEEE-CS)⁶⁷ et le Systems Engineering Research Center (SERC)⁶⁸. Le BKCASE a pour but de partager les connaissances les plus importantes pour l'avancement de l'ingénierie systèmes, à travers son produit : Guide to the Systems Engineering Body of Knowledge (SEBoK) [BKCASE, 2017].

Le BKCASE [2017] affirme que le processus de gestion de décision est essentiel dans la discipline de l'ingénierie systèmes puisqu'il permet d'identifier, analyser et évaluer un ensemble d'alternatives ; supportant ainsi le processus de prise de décision qui est généralement complexe vû qu'il inclue plusieurs parties prenantes, différents objectifs et d'importantes incertitudes pouvant parfois être associées à des conséquences néfastes. Le processus de gestion de décision, selon le BKCASE, est présenté dans la Figure 4.11. Il est constitué de dix activités [BKCASE, 2017] :

- **Formuler la décision** : cette activité a pour but d'assurer une bonne compréhension de la décision par toute l'équipe. Pour ce, l'analyste doit décrire le système à concevoir, ses limites et ses interfaces. Le **contexte de la décision** doit aussi être explicite en notant la phase du cycle de vie dans laquelle la décision sera prise, la liste des décideurs et des parties prenantes, les ressources disponibles, etc.
- **Développer les objectifs et mesures** : Les décisions sont souvent complexes et impliquent une multitude d'objectifs, ce qui rend l'être humain incapable de les gérer sans qu'elles soient décomposées [Keeney, 2002]. Dans cette activité, les objectifs sont développés en menant des enquêtes et des réunions avec les experts et les parties prenantes, puis en procédant à l'analyse des compromis. En effet, la valeur ajoutée d'un système perçue par les parties prenantes peut comporter des objectifs contradictoires au niveau de la performance, du planning de développement, des coûts unitaire et de support, etc. Il est donc intéressant de s'assurer que les objectifs fondamentaux sont d'abord essentiels, puis décomposer chacun d'eux en sous objectifs compréhensibles, non redondants et concis. Les caractéristiques des objectifs et leur hiérarchie sont discutés plus en détails par Edwards et al. [2007]. Une mesure doit ensuite être définie pour évaluer les alternatives selon leur valeur pour chacun des objectifs déjà définis. La mesure doit être compréhensive, non ambiguë et opérationnelle [Edwards et al., 2007].
- **Générer les alternatives créatives** : dans cette activité sont générées les alternatives qui étendent l'espace de décision. Plusieurs techniques permettent d'assurer cette génération, elles sont présentées par Parnell et al. [2011] et Buede [2011].
- **Évaluer des alternatives** : Une fois les objectifs, mesures et alternatives sont définis, les experts calculent le score de chaque combinaison alternative/mesure en utilisant toutes les ressources nécessaires (données opérationnelles, données de test, simulations, connaissances des experts, etc.)
- **Synthétiser les résultats** : les scores issus de l'activité précédente sont transformés en une table de valeur en utilisant des fonctions de valeur qui permettent la transformation de l'espace de mesure à l'espace de valeur BKCASE [2017]. La table de valeur permet de visualiser les compromis entre les alternatives et de localiser les alternatives qui nécessitent d'être améliorées [Parnell et al., 2011].
- **Identifier les incertitudes** : les experts doivent discuter des incertitudes potentielles des variables pouvant impacter un ou plusieurs scores. La simulation de Monte-Carlo, permettant de calculer une

66. <http://www.bkcase.org/>

67. <https://www.computer.org/>

68. <http://www.sercuarc.org/>

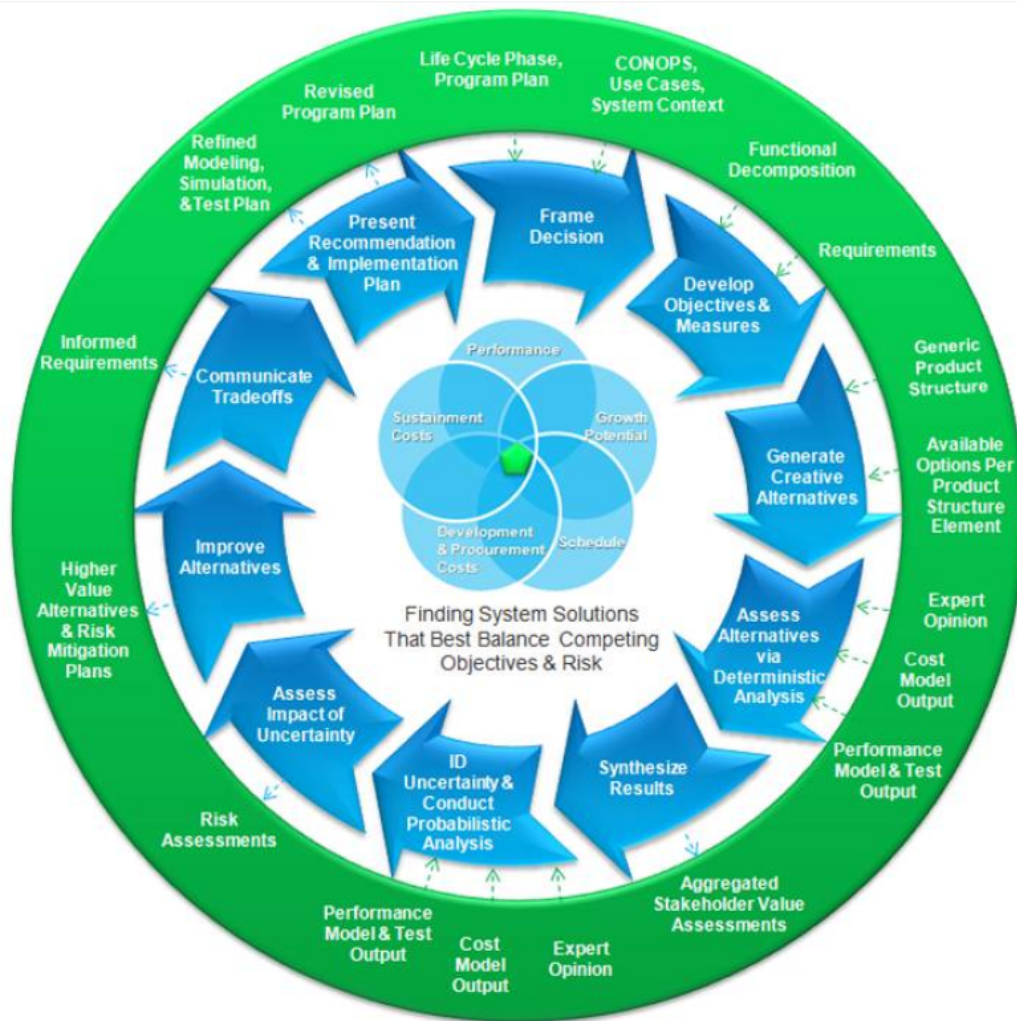


FIGURE 4.11 – Processus de gestion de décision selon le BKCASE [2017]

valeur approchée en utilisant des techniques probabilistes [Parnell et al., 2011], est l'exemple de la technique permettant d'étudier les incertitudes et identifier celles qui ont plus d'impact sur la décision.

- **Évaluer l'impact des incertitudes** : il s'agit ici d'étudier les risques associés aux incertitudes. Plusieurs méthodes permettent d'exécuter cette activité, nous citons par exemple les arbres de décision, les diagrammes d'influence, la simulation de Monte-Carlo, etc. [Parnell et al., 2011].
- **Améliorer les alternatives** : l'étude des données générées pour les alternatives pourrait révéler des opportunités comme la modification de certains choix de conception pour prendre en compte une valeur non exploitée et/ou réduire les risques. Ce processus de raffinement permet de transformer le processus de décision d'un processus basé sur les alternatives à un processus basé sur la valeur, ce dernier étant plus créatif permet de générer de meilleures alternatives [Keeney, 1993].
- **Étudier les compromis** : dans cette activité, les analystes identifient les observations clés concernant les compromis.
- **Présenter les recommandations et le plan d'implémentation** : il s'agit ici de décrire les recommandations sous forme de liste de tâches pour augmenter la probabilité de l'implémentation de la décision. Des rapports décrivant les résultats de toutes les activités précédentes ainsi que la logique derrière chaque décision sont très importants pour garder un historique pouvant être utile lors des prochaines décisions.

Le processus de gestion de décision du BKCASE (présenté dans la Figure 4.11) lui a permis de :

- Chercher les alternatives représentant le meilleur équilibre entre les objectifs à travers l'étude de compromis,
- Travailler dans les limites cognitives humaines à travers la décomposition du problème de décision. Cette décomposition permet aussi à chaque expert d'évaluer les alternatives qui s'inscrivent dans leur domaine d'expertise, ce qui rend fiable le résultat de l'évaluation.

4.3.3 L'OMG

L'Object Management Group⁶⁹ (OMG) est une association internationale ayant pour but de définir des normes communes pour les systèmes distribués⁷⁰ orientés-objet⁷¹. Elle se concentre sur les logiciels de modélisation, les systèmes et les processus métier, ainsi que sur les normes⁷² qui les supportent, dont une des plus récentes est la norme DMN (Decision Model and Notation) qui a été formalisée en partenariat avec la société *Decision Management Solutions*⁷³, société experte dans le domaine de la gestion et modélisation des décisions.

Le processus de prise de décision a été abordé par les normes de modélisation existantes selon deux points de vue :

- **modèles de processus métiers** : qui peuvent décrire la prise de décision au sein des processus métier en définissant les activités spécifiques dans lesquelles cette prise de décision a lieu,
- **Logique de décision** : qui peuvent définir la logique suivie lors de la prise des décisions individuelles comme les règles métier, les tables de décisions ou encore les modèles analytiques exécutables.

Il s'est avéré, selon plusieurs retours d'expérience, que la décision a souvent une structure interne qui n'est pas capturée de manière pratique par les deux points de vue décrits ci-dessus et que l'inclusion des détails, concernant les décisions, dans les modèles de processus rend ces derniers beaucoup plus complexes et rend difficile de faire, détecter ou propager les changements [DMS, 2016a]. La norme DMN vient alors pour combler ces besoins en introduisant un troisième point de vue appelé *diagramme d'exigence de décision* (*Decision Requirements Diagram* en Anglais), qui forme le pont entre les modèles de processus et la logique de décision :

- Les **modèles de processus métier** définissent les activités dans lesquelles la prise de décision doit se produire (par exemple, l'activité *Decide routing* dans la Figure 4.12).
- Les **diagrammes d'exigences de décision** définissent les décisions qui doivent être prises au cours de ces activités, leurs relations, et leurs exigences pour la logique de décision (sur la Figure 4.12, la décision *routing* dépend de deux autres décisions qui sont *Application risk* et *Eligibility*).
- La **logique de décision** définit les décisions de manière suffisamment détaillée pour permettre la validation et/ou automatisation (la décision sur l'éligibilité suit des règles, présentées sous la forme d'une table de décision dans la Figure 4.12). La table de décision est détaillée dans la Section 4.3.4.

Le *diagramme d'exigences de décision* et la *logique de décision* forment ensemble le modèle de décision (DMN), présenté sur la Figure 4.12, dont l'objectif principal est de fournir les concepts nécessaires pour modéliser les décisions afin de représenter les processus de prise de décision sous forme de diagrammes, et fournir ainsi une notation commune, facilement compréhensible par tous les utilisateurs, allant des *analystes métier* ayant besoin de créer d'abord les exigences initiales de décision puis les modèles de décisions plus détaillés, aux *développeurs* chargés d'intégrer et automatiser les décisions au sein des processus de

69. <http://www.omg.org/>

70. Ce sont des réseaux constitués d'ordinateurs autonomes connectés à l'aide d'un intergiciel de distribution. Ils permettent de partager différentes ressources et capacités pour fournir aux utilisateurs un réseau cohérent unique et intégré : https://en.wikipedia.org/wiki/Distributed_computing

71. Dans un système orienté objet, les données sont représentées par des objets réutilisables avec lesquels l'utilisateur et les autres objets du système peuvent interagir. Ce type de systèmes permet à l'utilisateur de se concentrer sur les tâches plutôt que sur les outils d'implémentation (vu qu'il existe plusieurs langages et outils permettant d'implémenter les systèmes adoptant une approche orientée objet) : [https://en.wikipedia.org/wiki/Object_\(computer_science\)](https://en.wikipedia.org/wiki/Object_(computer_science))

72. Toutes les normes OMG sont disponibles sur : <http://www.omg.org/spec>

73. <http://www.decisionmanagementsolutions.com/>

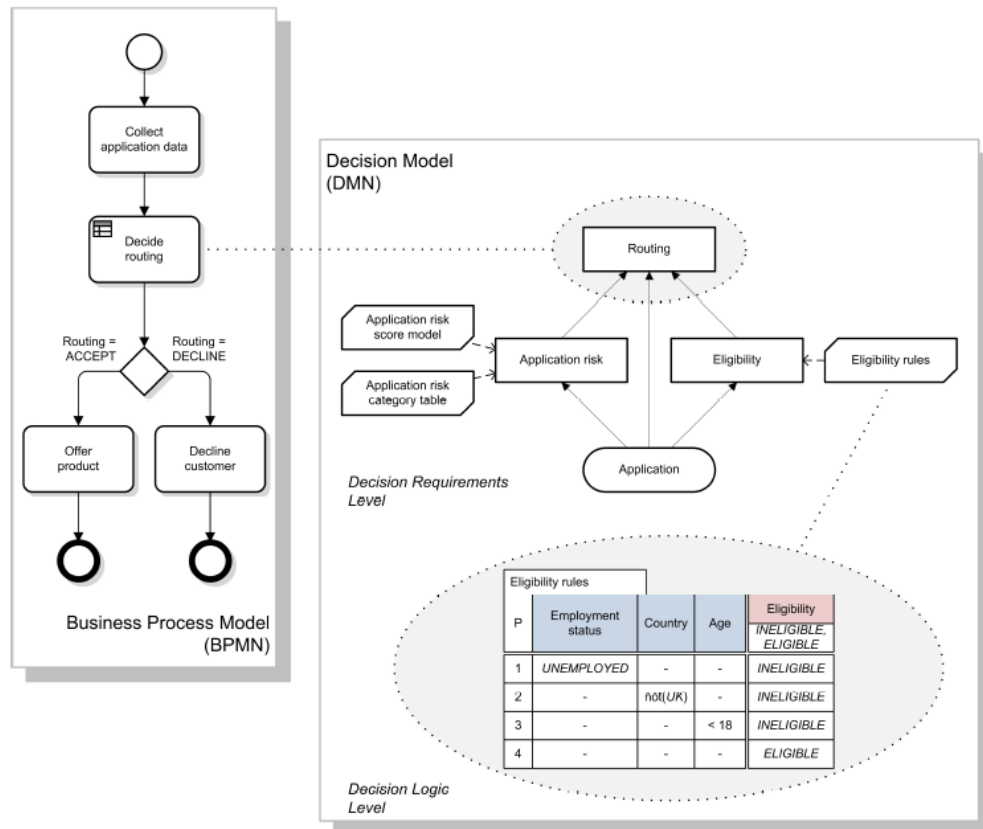


FIGURE 4.12 – La relation entre les trois perspectives du processus de prise de décision selon l'OMG [2016]

l'entreprise, aux *responsables métier* qui géreront et superviseront ces décisions [OMG, 2016].

Le diagramme des exigences de décision (partie supérieure du modèle DMN présenté dans la Figure 4.12) contient quatre éléments (Figure 4.13) :

- **Décision (Decision)** : c'est une entité représentée par un rectangle. Chaque décision a un ensemble de données d'entrée et délivre l'ensemble de sortie qui satisfait les règles de décision,
- **Données d'entrée (Input Data)** : représentées par des ovales, ce sont les données qui doivent être transmises en entrée d'une décision dans le diagramme,
- **Modèle de connaissance métier (Business Knowledge)** : représenté par un rectangle à coins coupés, c'est l'ensemble des règles métier, tables de décisions, arbres de décisions, ou parfois même des modèles analytiques prédictifs décrivant précisément comment prendre une décision,
- **Source de connaissances (Knowledge Source)** : ayant la forme d'un document sur la Figure 4.13, elle fait référence aux documents ou personnes sources à partir desquels la logique de décision a été dérivée.

Les quatre éléments détaillés ci-haut sont inter-connectés dans le diagramme via des flèches. Les flèches pleines indiquent les informations exigées pour une décision, qui peuvent être des données d'entrée ou des décisions nécessaires pour la décision en cours (étant donné que les décisions complexes sont décomposées en sous décisions). Les modèles de connaissances métier, décrivant comment prendre une décision, peuvent être liés à cette dernière avec des flèches en pointillées, et les sources de connaissances (personne ou document) sont liées aux décisions avec des flèches arrondies en pointillées.

Selon la DMS [2016b], le processus de modélisation de décision, ayant pour but de générer le modèle de décision (Figure 4.12), est constitué par quatre activités qui sont exécutées de manière itérative (Figure 4.14.a) jusqu'à ce que la décision, qui doit être prise, soit complètement spécifiée et comprise par toute l'équipe des décideurs :

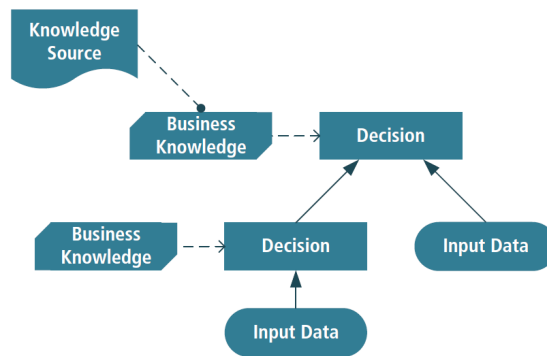
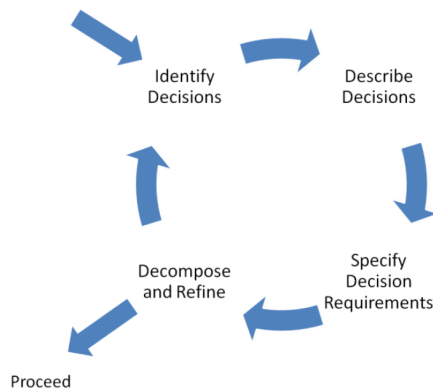
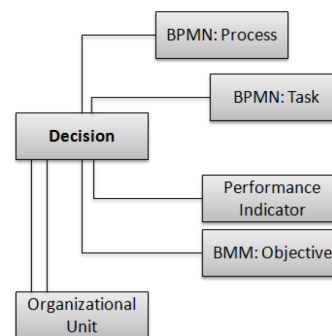


FIGURE 4.13 – Diagramme des exigences de décision [IIBA, 2015; OMG, 2016]

- **Identifier les décisions** qui doivent être prises, cette activité se situe au niveau de la perspective *modèle de processus* (Figure 4.12).
- **Décrire les décisions** et étudier comment leur amélioration impacte les objectifs et les métriques métiers liées aux indicateurs de performances. En effet, toute décision est liée -entre autres- à un ou plusieurs objectifs et un ou plusieurs indicateurs de performance comme le montre la Figure 4.14.b,
- **Spécifier les exigences de décisions** en partant des descriptions simples aux descriptions plus détaillées pour identifier les informations et connaissances requises pour prendre la décision, cette activité se situe au niveau de la perspective *diagrammes d'exigences de décision* (Figure 4.12).
- **Décomposer et raffiner** les diagramme d'exigences de ces décisions pour identifier d'éventuelles décisions additionnelles qui nécessitent d'être spécifiées.



4.14.a. Processus de modélisation de décision [DMS, 2016b]



4.14.b. Contexte de la décision dans la norme DMN [DMS, 2016a]

FIGURE 4.14 – Processus cyclique de la modélisation de décision et le contexte de cette dernière dans la norme DMN

La norme DMN a non seulement permis de créer un pont normalisé pour lier la conception des décisions et leurs implémentations, mais a aussi permis de fournir des modèles de décisions interchangeables entre les entreprises grâce à la représentation XML (eXtensible Markup Language)⁷⁴ qui leur est associée. Cette norme est conçue pour être utilisée parallèlement avec la norme BPMN [OMG, 2013], issue également de l'OMG, et est ouverte pour d'éventuelles extensions, notamment :

- L'ajout de nouveaux indicateurs aux éléments graphiques (Figure 4.13) pour souligner un attribut spécifique de l'élément DMN, ou pour représenter un nouveau sous-type du concept correspondant,
- L'ajout d'une nouvelle forme représentant un nouvel élément à condition qu'elle ne soit pas en conflit avec une forme spécifiant un élément existant,

74. C'est un langage définissant le format d'échange de données textuelles, il est compris à la fois par l'homme et la machine : <https://www.w3.org/XML/>

- La coloration des éléments graphiques pour faciliter la visualisation ou étendre l'information représentée par l'élément,
- La modification du style de ligne d'un élément graphique à condition qu'elle n'entre pas en conflit avec le style de ligne des autres éléments.

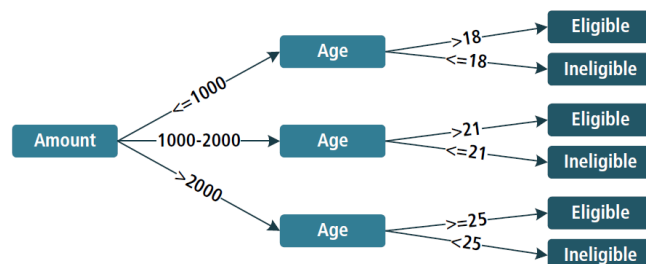
4.3.4 L'IIBA

L'International Institute of Business Analysis⁷⁵ (IIBA) est une association professionnelle ayant pour but de supporter les analystes métier en leur proposant un standard appelé "Guide to the Business Analysis Body of Knowledge (BABOK)" [IIBA, 2015]. Ce standard se voit comme une collection des expertises et connaissances métier décrivant les bonnes pratiques dans l'analyse économique (Business Analysis en Anglais) qui couvre les domaines de l'analyse des projets, gestion des processus métier, intelligence économique, développement agile, etc. L'IIBA adopte la modélisation des décisions comme une technique qui permet de montrer comment ces dernières sont prises dans une organisation. Les modèles de décision peuvent être utilisés pour les décisions simples et complexes :

Modèles de décisions simples : Ils utilisent les tables de décision ou les arbres de décisions, qui permettent de souligner la combinaison des règles qui amènent à la décision. Une table de décision, comme le montre la Figure 4.15.a, est une représentation tabulaire compacte de l'ensemble des règles dont chaque ligne (parfois colonne) représente une règle et chaque colonne (parfois ligne) représente une condition - parmi plusieurs autres - de cette règle. Lorsque toutes les conditions de la règle sont vérifiées pour un ensemble de données d'entrée, l'action qui correspond à cette règle est choisie. Prenons comme exemple la table présentée dans la Figure 4.15.a, la demande de prêt est validée si la règle suivante est vérifiée : " $(montant \leq 1000 \wedge age > 18) \vee (montant \in [1000, 2000] \wedge age > 21) \vee (montant > 2000 \wedge age \geq 25)$ ", sachant que les symboles " \wedge " et " \vee " représentent respectivement les opérateurs "ET" et "OU" logiques. Un arbre de décision, comme le montre la Figure 4.15.b, est aussi utilisé pour représenter l'ensemble des règles de décision. Chaque chemin de l'arbre allant du nœud racine (premier rectangle) au nœud feuille (dernier rectangle) représente une seule règle. Chaque niveau (vertical) dans l'arbre représente un élément spécifique de données (montant, age, etc.) et chaque branche dans l'arbre (flèche) représente une condition devant être vérifiée pour passer au niveau (rectangle) suivant. Comme dans les tables de décision, l'arbre de décision sélectionne le ou les nœuds feuille dont toutes les branches satisfont les conditions.

Eligibility Rules		
Loan Amount	Age	Eligibility
≤1000	>18	Eligible
	≤18	Ineligible
1000–2000	>21	Eligible
	≤21	Ineligible
>2000	≥25	Eligible
	<25	Ineligible

4.15.a. Exemple d'une table de décision



4.15.b. Exemple d'un arbre de décision

FIGURE 4.15 – Décisions simples : table et arbre de décision montrant les règles d'éligibilité à un prêt [IIBA, 2015]

Modèles de décisions complexes : Ces modèles commencent d'abord par décomposer les décisions en sous-décisions élémentaires pouvant être séparément décrites, puis décrivent la combinaison de ces sous

75. <http://www.iiba.org/>

décisions élémentaires, qui permet d'aboutir à la décision complexe de départ, en utilisant le diagramme des exigences de décision de l'OMG (Figure 4.13).

Le modèle de décision est lié au modèle de processus, aux mesures de performance et au modèle organisationnel. Il a plusieurs avantages dont le plus important est de permettre d'avoir une vision analytique sur le processus de prise de décision.

4.3.5 L'ISO

L'International Organization for Standardization⁷⁶, plutôt connue sous son acronyme ISO, est une organisation ayant pour but de mettre en place des standards communs pour faciliter la communication au niveau international. Dans le domaine industriel, l'ISO souligne l'importance du système de gestion de qualité qui permet de [ISO, 2015a] :

- Évaluer le contexte global d'une entreprise pour définir les clients et à quoi ils s'attendent. Ceci permettra de définir ses objectifs et ses opportunités,
- Satisfaire les clients en cherchant à répondre à leurs exigences. Ceci permettra à l'entreprise de conserver ses clients et d'attirer de nouveaux clients et ainsi augmenter son business et s'ouvrir à de nouveaux marchés,
- Être efficiente. En effet, plus les processus sont alignés et compris par tous les membres de l'entreprise, plus la productivité et l'efficacité augmentent et plus les coûts diminuent,
- Évaluer et traiter les risques associés aux activités de l'entreprise.

Les exigences de conception d'un système de gestion de qualité sont décrites par le standard ISO 9001 [ISO, 2015b,a] qui liste sept principes de gestion de qualité, parmi lesquels se trouve le **processus de prise de décision basée sur la preuve**. Ce processus, qui se compose de deux activités qui sont l'**analyse** et l'**évaluation** des données et informations, permet de produire les décisions qui ont plus de probabilité de produire les résultats désirés [ISO, 2015b]. En effet, la prise de décision est vue par l'ISO [2015b] comme un processus complexe (puisque'il implique toujours une certaine incertitude liée aux décisions), ayant souvent plusieurs sources de données dont les interprétations peuvent être différentes et subjectives. Il est, donc, important de comparer les résultats des exécutions réelles des processus à ceux souhaités ayant été préalablement définis comme objectifs de qualité pour ces processus et de comprendre les relations de cause à effet ainsi que les conséquences inattendues potentielles.

Le processus de prise de décision basée sur la preuve, comme son nom l'indique, est basé sur les preuves (informations qui montrent ou prouvent que quelque chose existe ou est vraie), souvent déduites à partir des mesures et de supervision. Ces deux concepts, utilisés ensemble dans la norme ISO 9001, ont des significations différentes. La supervision est, comme nous l'avons déjà défini, l'activité qui sert à observer de près pour recorder ou détecter, elle est utilisée pour s'assurer que le processus fonctionne normalement. Tandis que la mesure est l'activité dans laquelle est générée la valeur d'une caractéristique quelconque d'un système (comme le temps d'exécution, ressources engagées, etc.) et de combien elle diffère de la valeur requise pour le fonctionnement normal. La supervision et la mesure sont deux processus complémentaires, générant tous les deux des preuves précieuses pour le processus de prise de décision.

Le processus de prise de décision basée sur la preuve, les faits et l'analyse de données permet de générer une décision objective avec un certain degré de confiance et permet aux entreprises de [ISO, 2015b] :

- Déterminer, mesurer et superviser les indicateurs de performances,
- Mettre les données nécessaires à disposition des membres concernés,
- Assurer que les données sont justes, fiables et sécurisées,
- Analyser et évaluer les données en utilisant les méthodes d'analyse appropriées,
- Prendre les décisions en se basant sur la preuve, l'expérience et l'intuition.

76. <http://www.iso.org/>

4.4 Synthèse et discussion de l'état de l'art industriel

Cette deuxième partie de l'état de l'art général a pour objectif de découvrir les pratiques et les processus de prise de décision industriels vu qu'aucun processus du milieu scientifique n'a répondu à nos objectifs (comme discuté dans la Section 3.2.7). Bien que aucun processus -parmi ceux présentés dans cette section- ne répond parfaitement à notre hypothèse de travail, nous reprenons les notions intéressantes qu'ils présentent, pour en tenir compte lors de la conception de notre propre processus de prise de décision. En effet, nous avons commencé par explorer le processus décisionnel de la NASA qui donne beaucoup d'importance à l'identification du **risque** de chaque alternative pour permettre la sélection de l'alternative qui présente un minimum de risque. Dans un contexte sûr, la même organisation souligne l'importance de la définition des critères de **performance** à base desquels les alternatives vont être évaluées avant d'être proposées. Les processus de prise de décision pendant le vol ajoutent une autre notion intéressante, qui est le **contrôle** de la décision exécutée. De plus, les processus décisionnels au sein de l'U.S.NRC jettent la lumière sur l'importance de la **délibération** avant la prise finale de la décision. Cette notion est d'une importance extrême s'il s'agit d'un nouveau décideur (non expérimenté), ou d'un décideur qui veut explorer ce que ses collègues ont pris comme décision pour des problèmes similaires. Nous retenons du processus décisionnel de Schneider Electric, l'importance de **capturer** les informations pouvant être importantes pour faciliter la prise de décision. De même, l'INCOSE propose dans son processus décisionnel de **tracer** la décision prise en mentionnant toutes les ressources utilisées, ceci est important parce qu'il permet de **réutiliser** la décision lors d'un nouveau problème similaire ou la modifier pour correspondre à un nouveau problème de décision. Quelques modèles de décision, permettant de capturer ces dernières, sont proposés par l'OMG et l'IIBA. Une autre notion importante a été introduite dans le processus décisionnel de la BKCASE, il s'agit de l'**incertitude**. Il est en effet nécessaire d'évaluer l'impact des incertitudes de chaque décision avant de la prendre. En outre, pour proposer des décisions fiables, L'ISO propose d'adopter un processus de prise de décision basé sur la **preuve**, cette dernière fait référence aux expériences passées, entre autres.

4.5 Prononciation du verdict : adopter une solution existante ou concevoir une autre ?

Des discussions 3.2.7, 3.3.4, 3.4 et 4.4, nous concluons qu'aucun processus de prise de décision n'est adapté à notre contexte de recherche et qu'aucun système supportant la prise de décision ne répond à la totalité des quatre fonctions et aux caractéristiques recherchées. Par conséquent, il nous appartient de concevoir un nouveau système d'aide à la décision qui, en suivant un certain processus de prise de décision que l'on concevra, permet de supporter les utilisateurs dans leurs processus décisionnels, tout en répondant aux caractéristiques établies dans la Section 2.3.4.

Pour concevoir notre système d'aide à la décision, nous considérons ces deux éléments (comme mentionné dans la Figure 3.1) :

- **Technique d'aide à la décision** : nous adoptons la classe des **techniques basées sur l'intelligence artificielle** comme discuté dans la 3.2.7 et comme mentionné dans notre hypothèse de travail. L'utilisation de ces techniques est justifiée. En effet, les retours industriels montrent que les concepteurs ou superviseurs préfèrent les procédures qui ont été testées ou les approches basées sur les expériences passées [Renzi et al., 2017]. En outre, Ahmed and Hansen [2002] montrent à travers leur étude empirique que les concepteurs suivent une stratégie de décision qui dépend de leurs perceptions du statut actuel du processus de conception au lieu d'appliquer une méthode de décision. En outre, la capacité des décideurs à résoudre des problèmes en prenant des décisions efficaces a toujours été considérée comme le fruit des années d'expertise. Cependant, une observation minutieuse de leur processus de décision révèle que l'expérience n'empêche pas certains décideurs de manquer de rigueur lorsqu'ils réfléchissent et prennent des décisions. De ces retours industriels nous voyons que l'utilisation des techniques de l'intelligence artificielle permet de résoudre la problématique en

analysant et évaluant les expériences passées, avant de les proposer à l'utilisateur.

- **Processus d'aide à la décision** : il nous semble très important de garder les notions de **risque**, d'**incertitude** et de **performance** comme indicateurs d'évaluation de l'alternative lors de la conception de notre processus de prise de décision. De plus, nous souhaitons **capturer** les ressources utilisées ainsi que les décisions passées que nous considérons comme **preuves** lors de la prise d'une nouvelle décision. En outre, nous voulons que notre processus de prise de décision permette la délibération avant la prise de décision finale, **en offrant au décideur la possibilité d'explorer les décisions prises par ses collègues**.

Deuxième partie

PROPOSITION : SYSTÈME D'AIDE À LA DÉCISION INTELLIGENT ET DYNAMIQUE (IDSS)

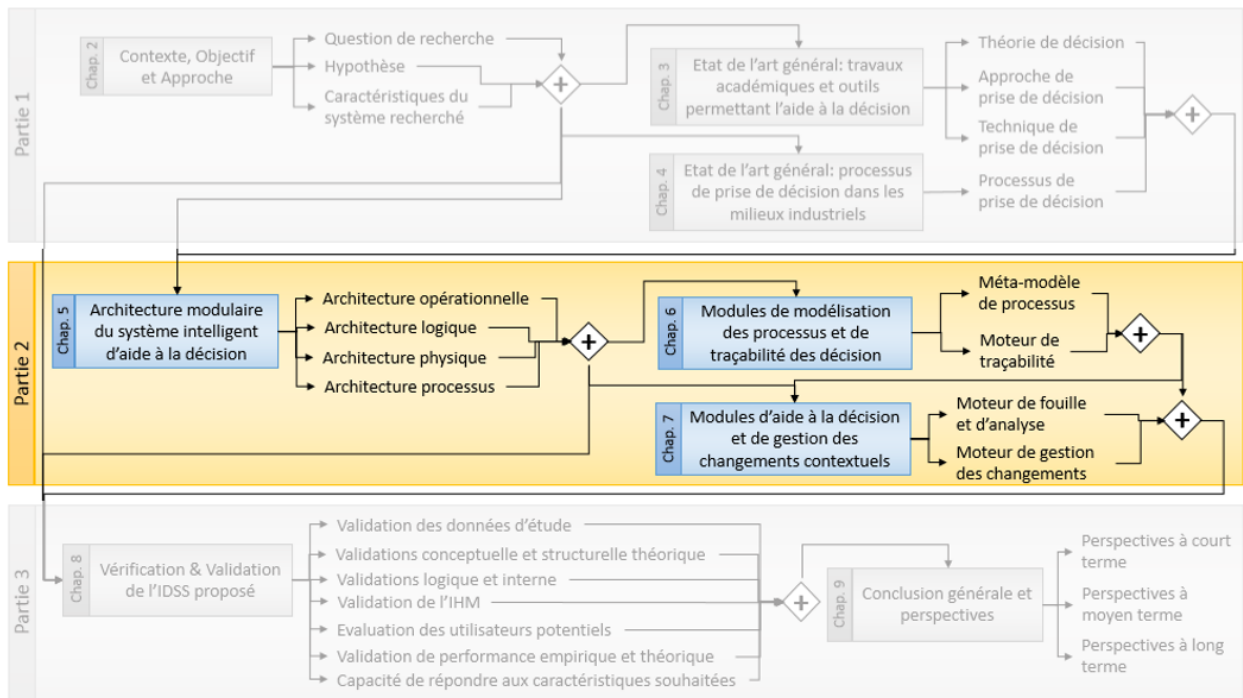


FIGURE 4.16 – Structure de la deuxième partie du manuscrit (décrite en utilisant le formalisme BPMN [OMG, 2013]).

Après avoir fait un tour d'horizon sur les travaux de recherche académiques, les outils commercialisés, et les processus industriels concernant la prise de décision ; et après s'être assuré qu'aucun des travaux étudiés ne répond intégralement à nos critères de recherche, nous introduisant dans cette partie, notre système intelligent d'aide à la décision (IDSS). Le chapitre 5 a pour but de définir l'architecture du système IDSS selon différents points de vue. Le chapitre 6 a pour but de décrire les premier et deuxième modules de notre système, qui ont pour but de modéliser et tracer les processus, respectivement. Le Chapitre 7, présente le module d'aide à la décision qui propose à l'utilisateur le bon pattern au bon moment. Ce module est dynamiquement mis à jour par le quatrième module, relatif à la gestion des changements contextuels.

Architecture modulaire du système intelligent d'aide à la décision

"User-centered design means understanding what your users need, how they think, and how they behave - and incorporating that understanding into every aspect of your process."

Jesse James Garrett

5.1 Introduction	88
5.1.1 Qu'est ce que l'architecture des systèmes? Et comment nous la définissons dans notre contexte?	88
5.1.2 État de l'art sur les méthodes d'architecture des systèmes	88
5.1.3 Synthèse et discussion	92
5.2 Architecture du système intelligent d'aide à la décision	92
5.2.1 Point de vue opérationnel (boîte noire)	93
5.2.2 Point de vue logique (boîte blanche)	93
5.2.3 Point de vue physique	95
5.2.4 Points de vue processus et scénarios	96
5.3 Synthèse et conclusion	98

La conception des systèmes modulaires, efficaces et efficients, est le résultat d'un mélange bien dosé de connaissances, de l'expérience et de la créativité artistique, toutes supportées par les sciences exactes qui garantissent le bon fonctionnement du système en fournissant des outils de synthèse et de vérification. Cette approche modulaire qui a trouvé ses premières applications dans les secteurs de l'informatique, l'automobile et l'aéronautique, a pour objectif de réduire la complexité, le temps et les coûts de fabrication des systèmes tout en offrant la possibilité de réutiliser les modules lors de la conception de nouveaux systèmes ou de les combiner pour donner lieu à un ensemble infini de configurations possibles.

Les chapitres précédents ont permis d'éclaircir les objectifs et de prononcer le besoin qui est de concevoir un système d'aide à la décision modulaire et intelligent (qui utilise les techniques de l'IA). Dans le présent chapitre, nous essayons de construire l'architecture de ce système en adoptant l'architecture des systèmes (Systems Architecting - SA), étant un sous processus de l'ingénierie des systèmes, comme une façon logique de penser et de concevoir. Nous commençons d'abord par découvrir les pratiques d'architecture des systèmes existantes dans la littérature, puis nous décrivons l'architecture de notre système intelligent d'aide à la décision selon les points de vue choisis avant de clôturer le chapitre par une synthèse résumant et discutant ce qui a été fait.

5.1 Introduction

5.1.1 Qu'est ce que l'architecture des systèmes? Et comment nous la définissons dans notre contexte?

La conception des systèmes industriels, marqués par une complexité de plus en plus croissante (due à l'interconnexion de matériels, de logiciels et des objets IoT⁷⁷ ainsi qu'à la nouvelle industrie 4.0⁷⁸, entre autres), requiert une approche système globale visant à comprendre les principes de l'architecture du système en question selon plusieurs vues et mettre en œuvre les principales démarches à suivre. Ceci est étudié par ce qu'on appelle l'*architecture des systèmes*, sous processus de l'ingénierie des systèmes (Figure 5.1) défini par l'INCOSE [2015] comme étant : "*the fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution*".

La construction de l'architecture des systèmes se compose de plusieurs activités comme le montre la Figure 5.2 :

1. **Préparer la définition de l'architecture** : en identifiant et analysant les informations permettant d'aider à construire une compréhension de l'environnement nécessitant la solution afin d'établir un meilleur aperçu des préoccupations des parties prenantes et des perspectives qui guideront le développement des vues d'architecture.
2. **Développer les points de vue de l'architecture** : selon les besoins identifiés chez les parties prenantes, identifier les points de vue⁷⁹ qui sont associés à l'architecture ainsi que les frameworks supportant le développement des modèles et des vues.
3. **Développer les modèles et les vues des architectures candidates** : en sélectionnant (ou développant) les techniques et outils de modélisation et en définissant ou cadrant le contexte du système d'intérêt (System of interest - SOI) en termes de frontière, interfaces et comportement souhaité, entre autres.
4. **Relier l'architecture au design** : en déterminant les éléments du système qui reflètent les entités architecturales pour s'assurer que l'architecture soit bien réalisable. En effet, l'architecture des systèmes est conçue pour être agnostique par rapport au design, et peut donc faire appel à des éléments pouvant être théoriques et non réalisables sur le plan pratique.
5. **Évaluer les architectures candidates** : en utilisant les critères d'évaluation, l'étude de risque et d'autres analyses systèmes pour sélectionner la meilleure architecture.
6. **Gérer l'architecture sélectionnée** : en capturant les justifications de toutes les décisions prises et en gérant la maintenance de la présente architecture ainsi que l'évolution de son contexte pouvant impacter ses éléments (changements techniques, légaux, économiques, organisationnels, opérationnels, etc.).

! Dans notre contexte, nous définissons l'architecture des systèmes exactement comme introduit par l'INCOSE [2015]. De plus nous ne considérons que les quatre premières activités de la Figure 5.2, lors de la définition de l'architecture de notre système. En effet, nous définissons une seule architecture car l'étude de performance des architectures candidates ne fait pas l'objet de ce travail de recherche.

5.1.2 État de l'art sur les méthodes d'architecture des systèmes

Après avoir défini le concept de l'architecture des systèmes, très utilisé lors de la conception des systèmes complexes, nous souhaitons maintenant parcourir quelques méthodes d'architecture proposées dans la

77. L'internet des objets connectés (Internet of Things - IoT) fait référence aux dispositifs qui collectent et partagent les données via internet sans interaction de type homme-homme ou homme-machine [Ashton et al., 2009]

78. C'est une transformation cyber-physique de la fabrication, visant à promouvoir la fabrication connectée et la convergence numérique entre l'industrie, les entreprises et d'autres processus [Lasi et al., 2014]

79. Attention à la différence sémantique entre un point de vue et une vue. Le point de vue est un partitionnement du système (parties physique et logique par exemple), tandis que la vue est la représentation (projection) du système selon le point de vue considéré [ISO, 2011]

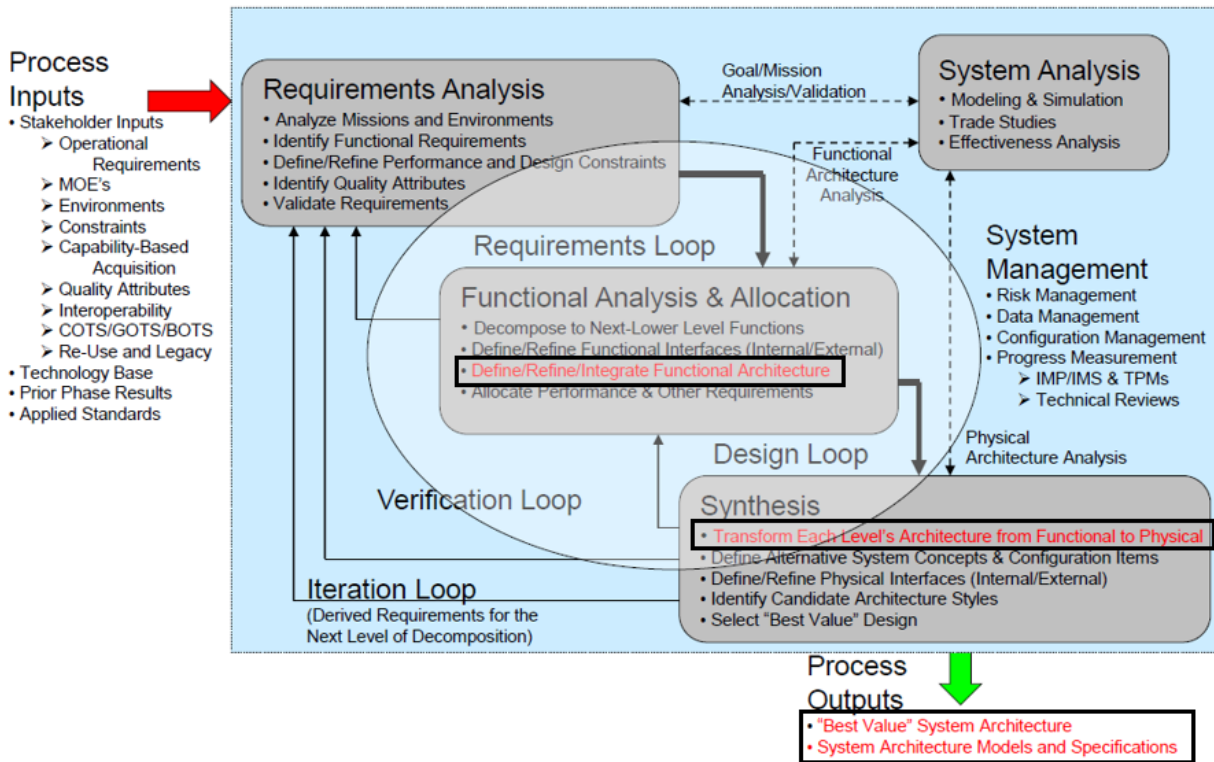


FIGURE 5.1 – Positionnement des activités de l’architecture des systèmes au sein du processus d’ingénierie des systèmes [Boeing, 2005]. Ces activités sont en rouge et sont encadrées par un rectangle avec un contour épais.

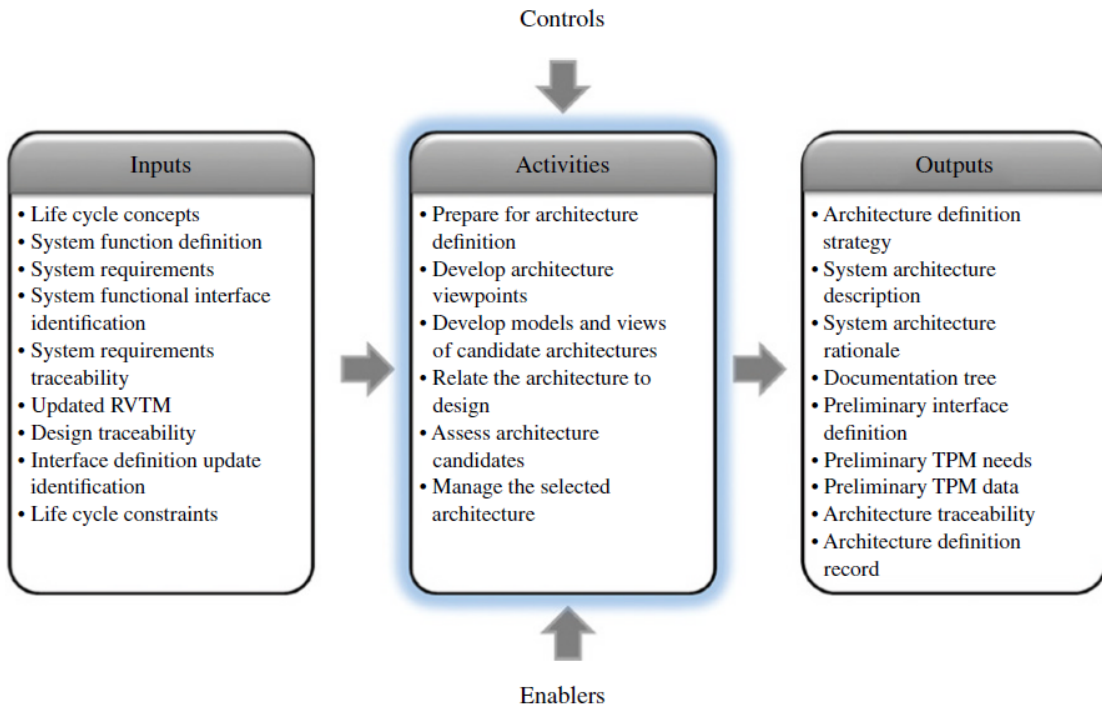


FIGURE 5.2 – Processus d’architecture des systèmes selon l’INCOSE [2015]

littérature. Rehtin and Maier [2010] ont présenté les quatre méthodologies ; qui selon eux sont les plus importantes dans le processus d’architecture des systèmes (Tableau 5.1). La méthodologie normative prescrit l’architecture telle qu’elle ”devrait être”, en d’autres mots, telle que décrit dans les manuels, les codes civils,

par les spécialistes, etc. cette méthodologie promet un résultat réussi par définition mais est limitée en termes d'insuffisance de réactivité face aux changements majeurs dans les besoins, les préférences, etc. Ceci donne lieu à la méthodologie rationnelle qui se base sur la science et les principes mathématiques pour trouver la solution au problème énoncé. Les deux méthodologies scientifiques (normative et rationnelle) sont toutes les deux analytiques, déductives, expérimentales, facilement certifiées et largement enseignées dans les milieux académiques et industriels. Elles sont certes nécessaires lors de l'élaboration de l'architecture d'un système, mais manquent de l'art ou la pratique, nécessaires, eux aussi, pour compléter la science lors de la conception des systèmes complexes. En effet, la pratique de l'architecture -tout comme la pratique de la médecine- est souvent non analytique, inductive, qui se voit plutôt un processus d'idées, de vision, d'intuitions et parfois de goûts.

Orientation	Nom de la méthodologie	Exemple
Orientée science	Normative (basée sur la solution)	Building codes et les standards de communication
	Rationnelle (basée sur la méthode)	Analyse et ingénierie des systèmes
Orientée art ou pratique	Participative (Basée sur les parties prenantes)	Ingénierie concourante et brainstorming
	Heuristique (basée sur les leçons apprises)	Simplify, simplify, simplify et SCOPE

TABLE 5.1 – Les quatre principales méthodologies de l'architecture des systèmes (adapté de [Rechtin and Maier \[2010\]](#))

La méthodologie participative reconnaît les complexités créées par l'existence de plusieurs parties prenantes, son objectif est alors de trouver un compromis qui satisfait tous les participants. À titre d'exemple courant, les concepteurs et les fabricants doivent s'accorder sur une multitude de détails; si un produit final doit être fabriqué facilement, rapidement ou de manière efficiente et rentable. Cette méthodologie se voit très importante; il est vrai que dans les cas simples mais courants, seuls le client, l'architecte et l'entrepreneur doivent être d'accord, mais dans la mesure où les systèmes deviennent plus complexes, de nouveaux participants doivent également être d'accord. La méthodologie heuristique est basée sur le "bon sens", c'est-à-dire sur ce qui est sensible dans un contexte donné. Ce dernier est tiré de l'expérience collective et est énoncé de la manière la plus simple et la plus concise possible pour décrire ce qu'on appelle "les heuristiques", définies dans le Section 3.2.3. Les heuristiques revêtent une importance particulière pour l'architecture car elles fournissent des guides permettant de pallier les problèmes de système insolubles [[Rechtin and Maier, 2010](#)].

En plus des quatre méthodologies présentées précédemment, il existe deux méthodes primaires pour concevoir l'architecture d'un système selon [Boeing \[2005\]](#). La première méthode est la SADT (Structured Analysis and Design Technique) [[Marca and McGowan, 1987](#)] qui offre une représentation graphique des exigences du système sous la forme d'un flux logique d'activités. La deuxième méthode couple l'analyse orientée objet (Object-Oriented Analysis - OOA) [[Hill, 1996](#)] et le langage de modélisation unifié (Unified Modeling Language - UML) [[Rumbaugh et al., 2017](#)], pour offrir la possibilité de décrire le système par des diagrammes de structure, de comportement, d'interaction, etc. [DoD \[2015\]](#) a présenté un framework appelé DoDAF (Department of Defense Architectural Framework) qui fournit une infrastructure de visualisation pour les préoccupations spécifiques des parties prenantes à travers des points de vue permettant de visualiser et d'assimiler la complexité d'une architecture à l'aide de moyens conceptuels tabulaires, structurels, comportementaux, ontologiques, temporels, graphiques, probabilistes, etc. Le framework DoDAF, qui est à l'origine de plusieurs modèles et frameworks d'architecture, propose plusieurs points de vue, comme le montre la Figure 5.3, pour permettre de décrire l'architecture du système d'une façon rigoureuse.

[Raymond \[1995\]](#) a proposé le modèle RM-ODP (Reference Model of Open Distributed Processing) qui se voit comme un modèle de référence pour la spécification de l'architecture d'un système distribué selon plusieurs points de vue :

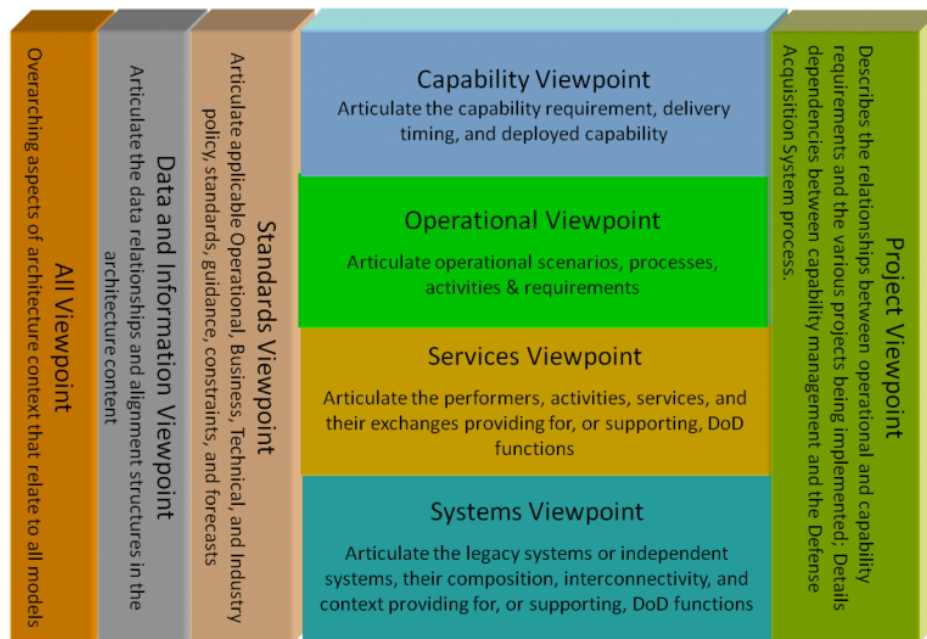


FIGURE 5.3 – Points de vue du frameworks DoDAF [DoD, 2015]

- **Point de vue entreprise** : permet de décrire l'objectif du système et comment répondre aux exigences métier,
- **Point de vue information** : permet de décrire l'information gérée par le système en termes de sémantique, structure, contenu, etc.
- **Point de vue informatique** : permet de décrire, via une décomposition fonctionnelle, les fonctionnalités du système et les objets qui le décomposent ainsi que leur interaction,
- **Point de vue ingénierie** : permet d'identifier les techniques et mécanismes supportant les interactions entre les objets du système,
- **Point de vue technologie** : permet de décrire les différentes technologies (mécaniques, électroniques, hydrauliques, logicielles, services, procédures, etc.) utilisées par le système pour répondre à son objectif.

L'AFIS [2012] propose de représenter l'architecture d'un système selon trois points de vue. Dans le point de vue opérationnel, le système est considéré comme une boîte noire interagissant avec le milieu extérieur en exécutant les fonctions (appelées aussi services) qu'il exige. Le deuxième point de vue est fonctionnel (appelé aussi logique); ici le système est d'abord décomposé d'une façon structurée via une analyse fonctionnelle interne pour dégager les fonctions qu'il doit exécuter, puis ces fonctions sont recomposées pour obtenir l'architecture finale du système. Le troisième point de vue est organique (appelé aussi physique); il permet de représenter l'arborescence des composants du système (appelé aussi organes) qui permet de réaliser les fonctions tel que déjà défini dans le point de vue fonctionnel. Kruchten [1995] propose un modèle d'architecture (Figure 5.4) décrivant les systèmes qui manipulent un nombre important de logiciels en se basant sur l'utilisation de points de vue multiples et concourants :

- **Point de vue développement (ou implémentation)** : il définit le système du point de vue du programmeur et concerne la gestion des logiciels.
- **Point de vue Logique** : il concerne les fonctionnalités que le système doit offrir aux utilisateurs finaux.
- **Point de vue Physique (ou de déploiement)** : il représente le système du point de vue de l'ingénieur système. Il concerne la topologie des composants logiciels sur la couche physique ainsi que les connexions physiques entre ces composants.
- **Point de vue Processus** : il traite l'aspect dynamique du système en expliquant les différents

processus du système et comment ils communiquent. Ce point de vue se concentre sur le comportement du système lors de son exécution.

- **Scénarios (ou cas d'utilisation)** : Il ne s'agit pas ici d'un point de vue proprement dit, mais plutôt d'une illustration décrivant les interactions entre les objets et entre les processus à travers un cas d'utilisation. Les scénarios sont utilisés pour identifier les éléments architecturaux et pour vérifier et valider la conception de l'architecture et tester son prototype.

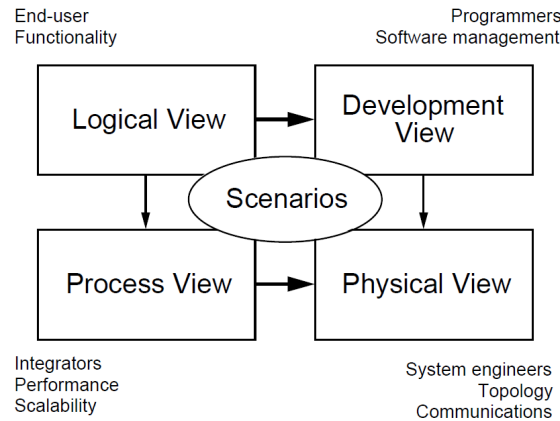


FIGURE 5.4 – Points de vue du modèle "4+1" de Kruchten [1995]

Ces différents points de vue permettent de gérer séparément les exigences fonctionnelles et non fonctionnelles et d'aborder séparément les exigences des différentes parties prenantes de l'architecture : l'utilisateur final, les développeurs, les ingénieurs système, etc. Plusieurs autres méthodes d'architecture sont proposées dans la littérature, nous citons à titre d'exemple le framework anglais MODAF (British Ministry of Defence Architecture Framework) [Bailey, 2008] et le framework TOGAF (The Open Group Architecture Framework) [Harrison, 2018].

5.1.3 Synthèse et discussion

Dans cette première section, nous avons introduit l'architecture des systèmes, étant un sous processus de l'ingénierie système. Cette discipline permet de gérer la complexité lors de la conception des systèmes en identifiant, tôt dans le cycle de vie du système, ses éléments, ses fonctions, les communications qu'aura le système avec l'environnement extérieur tout en permettant une conception modulaire réutilisable. L'architecture des systèmes permet ainsi d'identifier tout problème du à une description vague du design en ayant une vue holistique du système d'intérêt et de son environnement extérieur.

L'objectif du présent chapitre n'étant pas de présenter une étude rigoureuse des différentes méthodes d'architecture des systèmes (le lecteur peut cependant se référer au travail de Rouhani et al. [2013], entre autres, où sont discutées et comparées plusieurs différentes méthodologies d'architecture); nous souhaitons décrire notre architecture en adoptant les points de vue logique, physique, processus et scénarios du modèle "4+1" de Kruchten [1995] puisqu'il permet de décrire l'architecture d'un système intensif en logiciels (faisant appel à plusieurs logiciels, parfois en même temps), ce que nous supposons bien être le cas du système intelligent d'aide à la décision qu'on proposera. Notons que nous n'avons pas sélectionné le point de vue *développement* parce que nous voulons présenter les modules et la description des logiciels utilisés dans le point de vue physique afin d'explicitier les liaisons entre les organes physiques et les organes logiciels. En plus des points de vue du modèle "4+1" sélectionnés, nous souhaitons décrire notre système selon le point de vue opérationnel proposé par l'AFIS [2012] pour ainsi définir l'architecture la plus rigoureuse possible.

5.2 Architecture du système intelligent d'aide à la décision

Nous avons présenté dans la section précédente les différents points de vue proposés par l'AFIS [2012] et par le modèle "4+1" de Kruchten [1995]. Dans cette section, nous suivons le processus de la

Figure 5.5 pour définir les vues de notre système d'aide à la décision en le projetant sur les points de vue que nous avons sélectionnés : opérationnel, logique, physique, processus et scénarios.

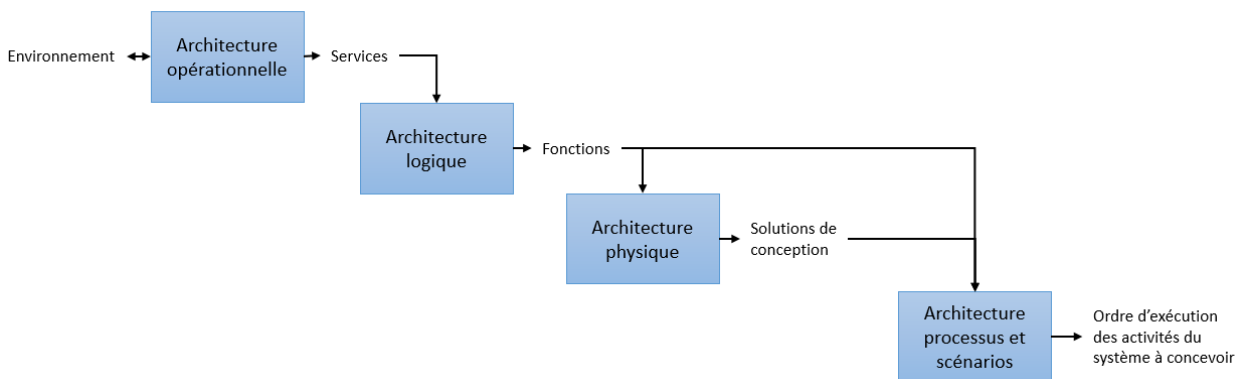


FIGURE 5.5 – Abstraction du processus suivi lors de la définition de l'architecture du système intelligent d'aide à la décision

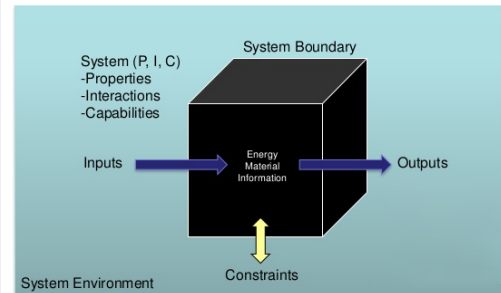
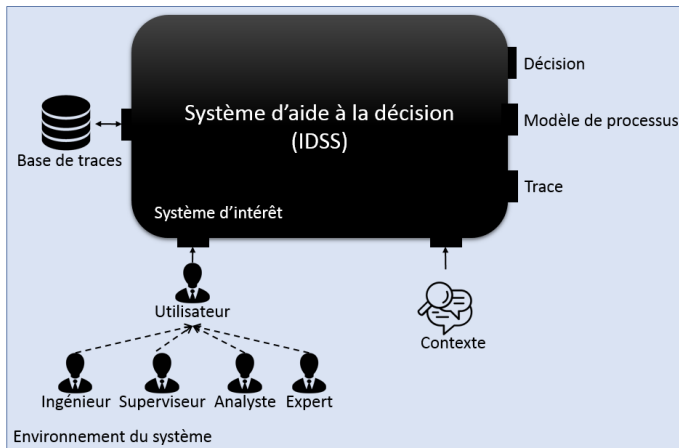
5.2.1 Point de vue opérationnel (boîte noire)

Ce point de vue permet de représenter le système comme une boîte noire qui interagit avec l'environnement externe en exécutant des fonctions (ou en rendant des services comme préfère l'appeler l'AFIS [2012]). Il existe trois types de diagrammes permettant de décrire la vue opérationnelle d'un système selon l'AFIS [2012]. Le premier est le diagramme de contexte qui décrit le système dans son environnement en identifiant les flux de données entrants et sortants. Le deuxième est le diagramme des cas d'utilisation qui décrit les différentes façons comment l'environnement extérieur (notamment les utilisateurs) interagit avec le système. Le troisième est le diagramme de séquence qui permet de spécifier les scénarios d'exécution d'une manière graphique en montrant l'ordre d'exécution des différents processus et/ou activités concurrents ainsi que les messages échangés entre eux, dans l'ordre dans lequel ils se produisent.

Nous souhaitons représenter notre système en utilisant le diagramme de contexte (ou boîte noire comme le montre la Figure 5.6.b) vu que les autres diagrammes seront utilisés pour représenter le système selon les autres points de vue considérés. La Figure 5.6.a montre que le système d'aide à la décision doit produire au moins une des trois sorties (la décision sollicitée, le modèle de processus et la trace d'exécution) à partir de la base des traces (anciennes exécutions des processus) et en satisfaisant deux contraintes majeures : le contexte de la décision et l'utilisateur. En effet, le système doit tenir compte du contexte pour proposer la bonne décision et être, ainsi, efficace (il ne doit pas proposer à un client d'acheter une voiture "Lamborghini Huracan" si son budget n'excède pas 100.000 €, ou encore ne doit pas proposer à une famille de faire un pique-nique s'il pleut). De plus, le système à concevoir doit proposer la bonne décision à la bonne personne, ceci revient à dire qu'il doit tenir compte de l'identité de l'utilisateur qui le déclenche (une machine à café intelligente, par exemple, ne doit pas proposer du café au client "Lionel" plutôt addict aux boissons au chocolat). L'utilisateur peut avoir plusieurs orientations ; il peut être un ingénieur ayant pour objectif de concevoir une voiture par exemple, un superviseur qui contrôle un processus industriel, un analyste ayant pour rôle d'améliorer l'ergonomie ou la performance du système ou encore un expert qui a pour objectif d'associer des règles métier aux modèles de processus afin de les améliorer, etc.

5.2.2 Point de vue logique (boîte blanche)

L'architecture logique répond aux préoccupations de l'utilisateur final en garantissant que ses exigences sont capturées par le système. Elle prend principalement en charge les exigences fonctionnelles, c'est-à-dire ce que le système doit fournir en termes de services à ses utilisateurs. Selon ce point de vue, le système est décomposé en un ensemble d'abstractions clés sous la forme de classes ou d'ensemble de classes réunies dans des packages. Cette décomposition sert non seulement à des fins d'analyse fonctionnelle,



5.6.a. Représentation de notre système selon le point de vue boîte noire

5.6.b. Point de vue boîte noire selon Atkins Ltd, Entreprise multinationale spécialisée dans la conception architecturale

FIGURE 5.6 – Vue holistique (externe) du système d'aide à la décision à concevoir

mais également à identifier des mécanismes et des éléments de conception communs aux diverses parties (modules) du système. Pour décrire l'architecture logique, nous souhaitons utiliser le diagramme de classe UML [OMG, 2011] qui décrit l'ensemble des classes (ou packages) et leurs relations logiques : association, dépendance, composition, héritage, etc. L'architecture logique de notre système d'aide à la décision est présentée dans la Figure 5.7.

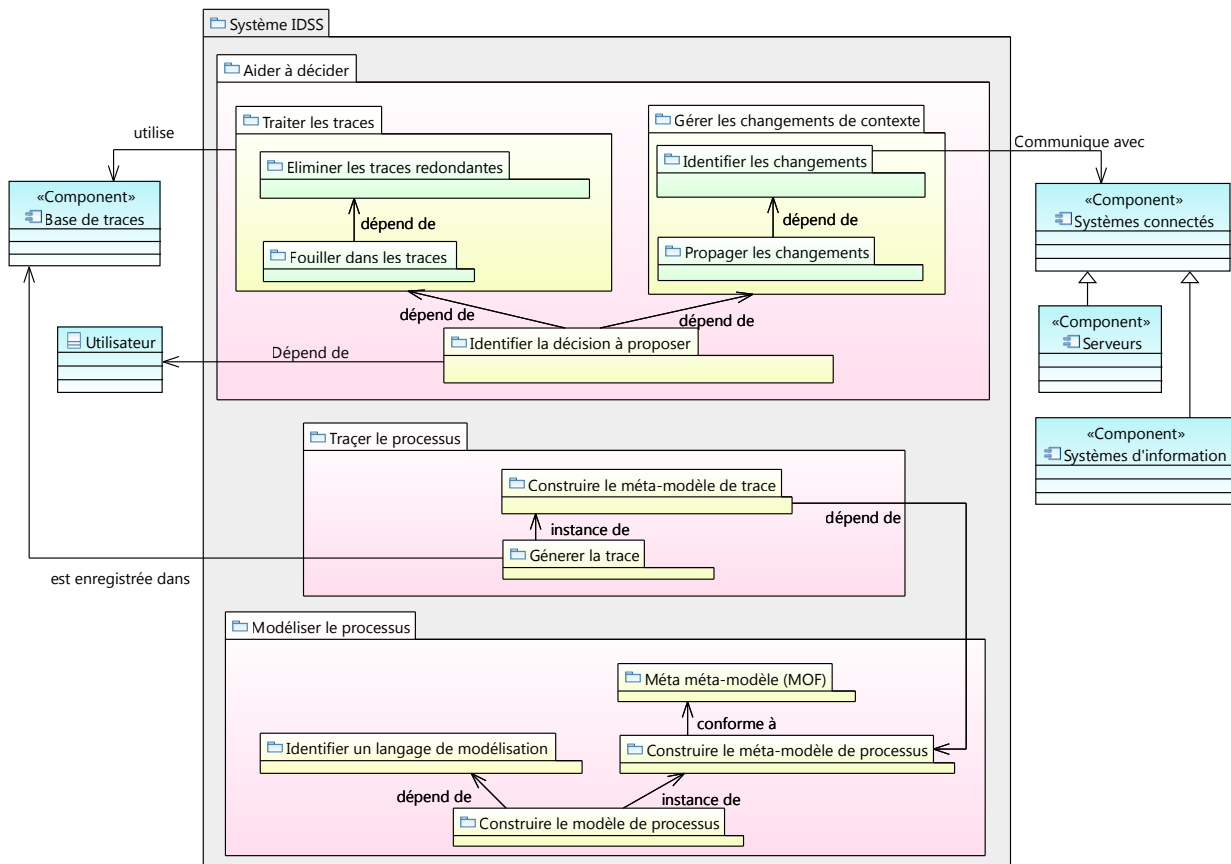


FIGURE 5.7 – Architecture logique de notre système intelligent d'aide à la décision, présentée sous la forme d'un diagramme de classe UML [OMG, 2011].

Trois grands packages sont présents dans l'architecture, dont chacun correspond à une sortie (i.e. service) parmi ceux identifiés par le point de vue boîte noire décrit dans la Section 5.2.1 :

Modéliser le processus : Ce package se voit comme une fonction, relative au service "modèle de processus" identifié dans la Figure 5.6.a, qui contient deux sous-fonctions. La première est de construire un méta-modèle (modèle d'un modèle) de processus conforme au MOF⁸⁰ (Meta-Object Facility) ayant pour but de spécifier les concepts pouvant être présents dans le modèle ainsi que leurs relations logiques. la deuxième sous-fonction est de choisir un des langages de modélisation existants pour représenter (modéliser) graphiquement les concepts capturés par le méta-méta modèle issu de la première sous-fonction.

Tracer le processus : Cette fonction, relative au service "Trace" identifié dans la Figure 5.6.a, contient deux sous-fonction, la première est de construire un méta-modèle de trace qui importe un ensemble de concepts définis par le méta-modèle de processus que nous avons identifié dans la fonction précédente. L'intérêt de construire ce méta-modèle de trace réside dans le fait qu'il n'ait parfois pas nécessaire de tracer toutes les données qui transitent dans une entreprise, décrites toutes par le méta-modèle de processus. La deuxième sous-fonction est de générer la trace à partir du méta-modèle de trace de la première sous-fonction et de l'enregistrer dans la base de traces qui sera exploitée par la fonction suivante.

Aider à décider : Cette fonction relative au service "décision" identifié dans la Figure 5.6.a, représente le noyau de notre système et se voit plus complexe que les deux précédentes. Elle contient trois sous-fonctions :

- **Traiter les traces :** les traces récupérées de la base des traces doivent d'abord être nettoyées en éliminant les traces redondantes dues à la même exécution du même processus plusieurs fois par la même personne. En effet, un ingénieur concepteur peut préférer d'exécuter dix fois un processus x en suivant toujours l'enchaînement des activités A_1 , A_2 et A_3 , ceci donnera lieu à dix traces similaires en termes d'enchaînement d'activités (la date d'exécution du processus doit évidemment être différente vu que l'être humain n'a pas encore la capacité de manipuler le temps). Après avoir éliminé les traces redondantes, il est temps de les analyser pour explorer et extraire les informations permettant de décrire les règles qui contrôlent le choix de chaque trace.
- **Gérer les changements :** une décision est prise dans un contexte donné, pouvant être sujet à plusieurs changements récurrents. Cette fonction a donc pour objectif d'identifier ces changements en communiquant avec l'environnement extérieur du système puis de les propager dans le système.
- **Identifier la décision à proposer :** ayant le résultat de l'apprentissage automatique, le profil de l'utilisateur et les changements occurrents, le système identifie la décision à proposer et la suggère à l'utilisateur.

5.2.3 Point de vue physique

L'architecture physique permet de décrire les solutions de conception permettant d'implémenter les fonctions du systèmes décrites dans la Figure 5.7. Elle est principalement destinée aux concepteurs et aux administrateurs qui doivent comprendre la cartographie du système en termes d'emplacements physiques des logiciels qui le constituent, les connexions physiques entre les nœuds⁸¹, le déploiement et l'installation, ainsi que l'évolutivité du système. L'architecture physique de notre système est présentée dans la Figure 5.8, elle consiste en trois couches physiques, dont chacune représente une solution de conception implémentant une des fonctions du système, identifiées dans la Figure 5.7. Dans la couche physique est généré le modèle de processus par rapport au méta-modèle conçu au niveau de la même couche. L'organe logiciel permettant d'implémenter cette solution est l'environnement de développement intégré "Eclipse"⁸². Dans la couche de traçabilité, le méta-modèle de trace est conçu par rapport au méta-modèle de processus, ayant déjà été

80. C'est un langage de méta-modélisation basé sur la notion de classe, association et héritage : <https://www.omg.org/spec/MOF/About-MOF/>

81. Ce sont les périphériques ou les points de données constituant un point d'intersection/connexion au sein d'un réseau donné

82. <https://www.eclipse.org/>

conçu dans la couche précédente, et la trace est générée et enregistrée dans la base des traces. L'organe logiciel implémentant cette solution est l'environnement "Eclipse"⁸². La couche d'aide à la décision reprend les traces de la couche précédente et les transforme en exécutant un programme Java⁸³. Les traces sont dorénavant traitées et préparées pour la fouille qui génère en sortie les patterns les plus significatifs ainsi que les règles de décision qui contrôlent leur choix. Dans la même couche sont filtrés les patterns, via des analyses "if-then", par rapport au contexte et au profil de l'utilisateur pour générer la ou les décisions à suggérer à l'utilisateur. Les organes logiciels permettant d'implémenter les sous-fonctions de cette couches sont "Eclipse"⁸² et les logiciels de fouille de processus "ProM"⁸⁴ et de données "Weka"⁸⁵, dont la justification de leur choix sera détaillée dans les chapitres suivants.

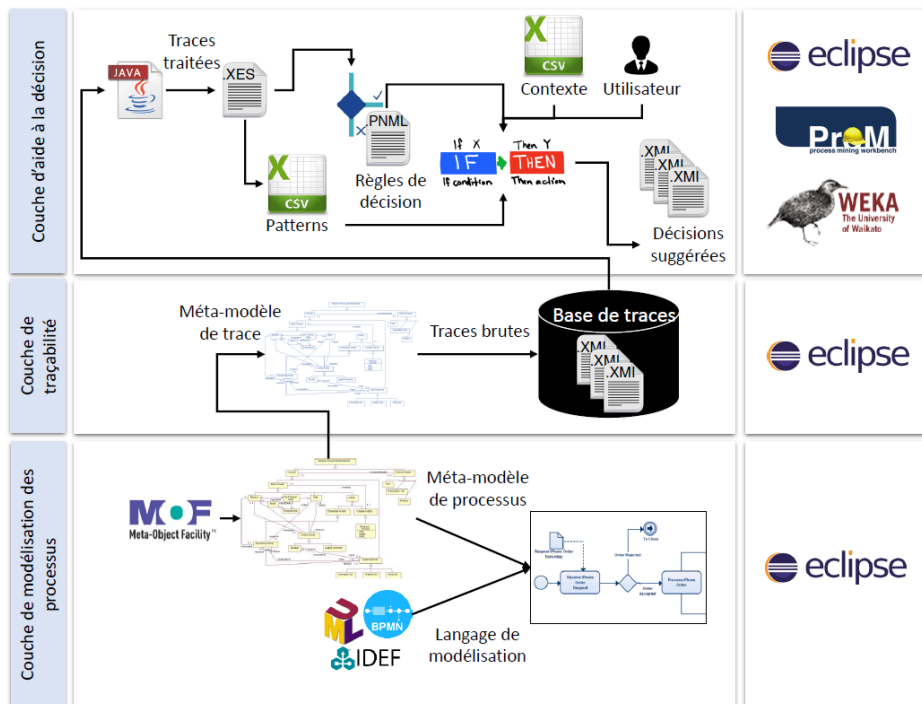


FIGURE 5.8 – Architecture physique et logicielle de notre système intelligent d'aide à la décision

5.2.4 Points de vue processus et scénarios

Les quatre points de vue du modèle "4+1" [Kruchten, 1995] (Figure 5.4), que nous avons choisi, fonctionnent ensemble de façon transparente en utilisant un petit ensemble de scénarios importants dont chacun est décrit par un script. Les scénarios sont en quelque sorte une abstraction des exigences les plus importantes qui permet à toutes les parties prenantes de comprendre comment le système est destiné à être utilisé. Le lecteur notera que ce point de vue est, plus ou moins, redondant avec les autres (d'où le "4+1" comme énonce Kruchten [1995]), la différence implicite réside dans le fait qu'il a deux objectifs principaux :

- Aider à découvrir les éléments architecturaux lors de la conception de l'architecture,
- Aider à valider et illustrer l'architecture après que sa conception soit terminée, à la fois sur papier et comme point de départ pour les tests d'un prototype architectural.

La conception des scénarios peut être exprimée à l'aide des diagrammes de scénarios d'objets et des diagrammes d'interaction d'objets [Booch, 2004] ou à l'aide des diagrammes de cas d'utilisation UML [OMG, 2011]. Nous souhaitons, par la suite, décrire notre scénario typique à l'aide de ce dernier parce qu'il est simple, plus expressif, standardisé et largement documenté. De plus, pour décrire les activités du processus de notre système (identifiées par le scénario typique) selon leur ordre d'exécution réel, et

83. C'est un langage de programmation informatique orienté-objet basé sur les classes : <https://www.java.com/en/>

84. <http://www.promtools.org/doku.php>

85. <https://www.cs.waikato.ac.nz/ml/weka/>

éventuellement identifier d'autres activités en augmentant le niveau de détail, nous souhaitons utiliser le diagramme de séquence UML [OMG, 2011]. Pour construire ces diagrammes, nous utilisons la version 1.0.5 du logiciel Papyrus Software Designer⁸⁶, environnement de modélisation open source développé par le LISE⁸⁷ du CEA-List⁸⁸ et hébergé par Eclipse⁸².

La Figure 5.9 présente le scénario général du système d'aide à la décision que nous souhaitons concevoir. Ce scénario est conforme au méta-modèle du diagramme de cas d'utilisation UML présenté dans la Figure 5.11.a. En effet, le rectangle nommé IDSS (System) marque les frontières du système d'aide à la décision de façon à ce que les éléments y appartenant feront partie du système et les éléments qui n'y appartiennent pas feront partie de l'environnement externe au système d'intérêt. Les bonshommes, associés au système à droite et à gauche, désignent les acteurs (matériels comme les utilisateurs ou immatériels comme les serveurs où sont stockées les données) qui interagissent avec le système. Les ellipses se trouvant à l'intérieur du système désignent les cas d'utilisation ("UseCase" dans le méta-modèle de la Figure 5.11.a). Un cas d'utilisation permet de capturer les services que le système offre à son environnement externe, ces services font référence aux trois fonctions principales définies dans la Figure 5.7. Deux cas d'utilisation peuvent être liés par différents types de liens (Figure 5.11.a) dont les plus utilisés sont le "include" (qui veut dire que le cas d'utilisation source est une tâche du cas d'utilisation cible⁸⁹) et le "extend" (qui veut dire que le cas d'utilisation source est une tâche optionnelle du cas d'utilisation cible).

La Figure 5.10 présente l'enchaînement réel des activités du système à concevoir, elle est conforme au méta-modèle du diagramme de séquence UML présenté dans la Figure 5.11.b. Les lignes pointillées (lifeline sur la Figure 5.11.b) sortants des boîtes rectangulaires sont appelées lignes de vie de l'objet (dont le nom est écrit sur la boîte), elles représentent la durée de vie de cet objet pendant le scénario en cours de modélisation et permet l'envoi/réception des messages entre les objets. Ces messages, pouvant être synchrones ou asynchrones, sont décrits par des flèches orientées en dessus desquelles est écrit le contenu du message. Le cadre d'interaction rectangulaire contenant le mot-clés "alt", permet de décrire des scénarios exclusivement alternatifs mettant en œuvre la suite d'opérations "si - alors, sinon". Il est possible d'exprimer les boucles (exécution plusieurs fois jusqu'à la vérification de la condition) dans un scénario en utilisant le cadre d'interaction rectangulaire contenant le mot-clés "loop".

Le script correspondant au scénario typique est le suivant :

1. L'utilisateur fait appel au système d'aide à la décision (noté IDSS (Intelligent Decision Support System) sur la Figure 5.9) en exécutant le fichier exécutable du système, qui crée et affiche son interface homme-machine (IHM) comme le montre la Figure 5.10. L'utilisateur peut demander au système de lancer l'interface relative à la modélisation de processus ou de l'aider à prendre la bonne décision dans une situation de décision lors de l'exécution d'un processus x .
2. Si le système est appelé pour la modélisation de processus, il commence par afficher le méta-modèle de processus, à la base duquel le processus présent sera modélisé, puis selon le langage de modélisation choisi par l'utilisateur, le système fait appel au logiciel approprié et l'affiche à l'utilisateur qui commence la modélisation graphique de son processus activité par activité jusqu'à ce que la dernière activité soit atteinte. Quand la boucle est arrêtée, le logiciel de modélisation affiche le modèle final du processus sous une forme graphique permettant, ainsi, à l'utilisateur de bien le visualiser et le comprendre. Après la fin de l'analyse, l'utilisateur ferme l'interface de l'IDSS (Figure 5.10).
3. Si le système est appelé pour aider l'utilisateur à décider, il commence d'abord par notifier le module

86. <https://www.eclipse.org/papyrus/>

87. Laboratoire d'Ingénierie dirigée par les modèles pour les Systèmes Embarqués

88. Commissariat à l'Énergie Atomique et aux Énergies alternatives-Laboratory for Integration of Systems and Technology : <http://www-list.cea.fr/en/>

89. Un élément est dit source s'il émet le lien, et est dit cible s'il reçoit le lien. En d'autres mots, la flèche du lien se dirige toujours vers l'élément cible.

- de traçabilité pour capturer la trace en cours, puis récupère les traces des anciennes exécutions du processus (Figure 5.10),
4. L'IDSS traite (nettoie et analyse) les anciennes traces (Figure 5.9) en éliminant d'abord les traces redondantes puis en trouvant les patterns et les règles de décision (Figure 5.10),
 5. Tout le long du processus décisionnel (d'où la boucle -loop- dans la Figure 5.10), l'IDSS récupère le contexte de la décision des serveurs et systèmes d'informations externes (ligne de vie contexte dans la Figure 5.10), puis identifie et propage les changements de contexte pouvant impacter la décision à proposer (Figure 5.9),
 6. En tenant compte du contexte et des traces d'exécution, l'IDSS identifie la solution (décision) la plus adéquate pour la situation de décision en question, en filtrant les patterns par rapport au contexte et au profil de l'utilisateur, et la suggère à l'utilisateur (Figure 5.10). Ce dernier a la possibilité d'accepter ou de refuser la décision suggérée. Si cette dernière est acceptée, le système IDSS continue de suggérer la décision suivante (si le processus décisionnel est toujours en cours), sinon (si la décision est refusée par l'utilisateur), l'IDSS ferme son IHM, mais continue d'écouter en arrière-plan ce que l'utilisateur fait ; à travers son module de traçabilité (Figure 5.10).
 7. Après la fin de l'exécution du processus en question, l'IDSS demande à son module de traçabilité de finir la capture de la trace, qui contient toutes les décisions prises par l'utilisateur, et enregistre la dans la base de traces pour l'enrichir (Figure 5.10).
 8. Après avoir eu toutes les décisions qu'il souhaitait avoir, l'utilisateur ferme l'IHM du système IDSS (Figure 5.10).

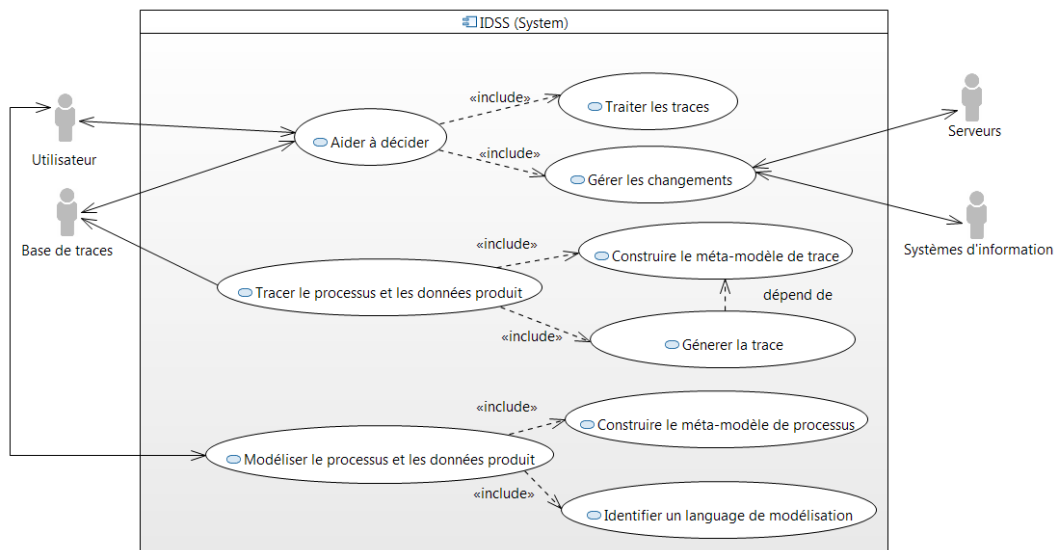


FIGURE 5.9 – Diagramme de cas d'utilisation UML décrivant le scénario typique du système à concevoir

5.3 Synthèse et conclusion

Dans le présent chapitre, l'architecture du système intelligent d'aide à la décision a été décrite selon plusieurs points de vue et en suivant un processus architectural défini, comme le montre la Figure 5.5. En effet, l'architecture *opérationnelle* présente le système comme une "boîte noire" en décrivant les flux de données entrants et sortants faisant références, respectivement, aux demandes des utilisateurs et aux services devant être rendus par le système. L'architecture *logique*, quant à elle, décrit le système comme une boîte blanche permettant d'identifier les fonctions qui permettent de donner lieu aux services décrits par l'architecture opérationnelle. Ces fonctions sont implémentées par des solutions de conception décrites dans l'architecture *physique* du système à l'aide des organes logiciels, décrits dans la même architecture. Ayant

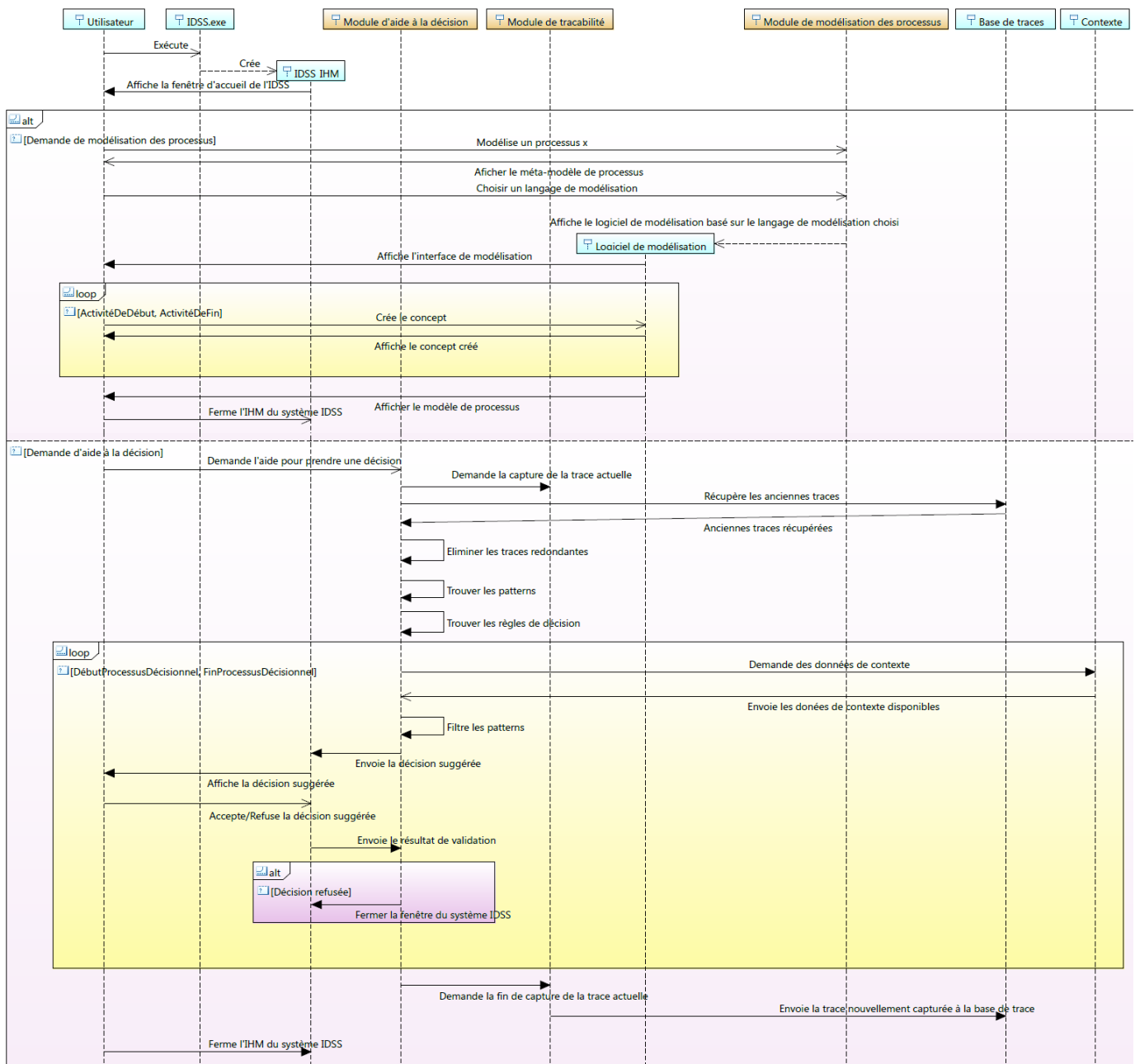
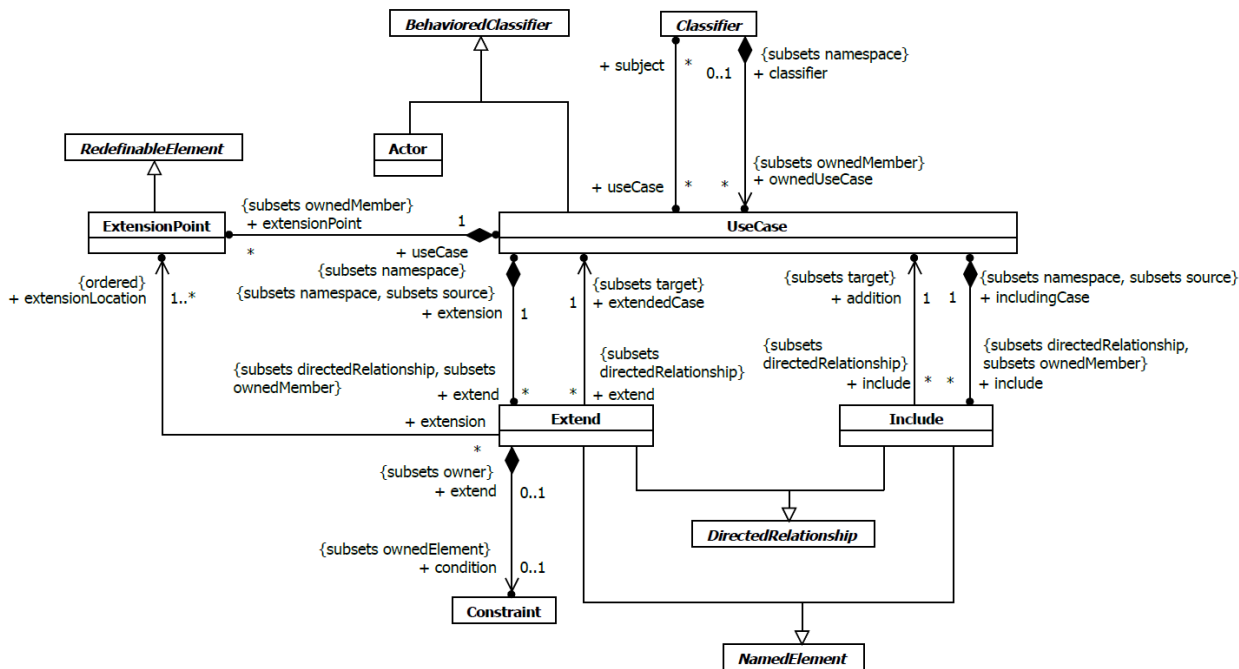


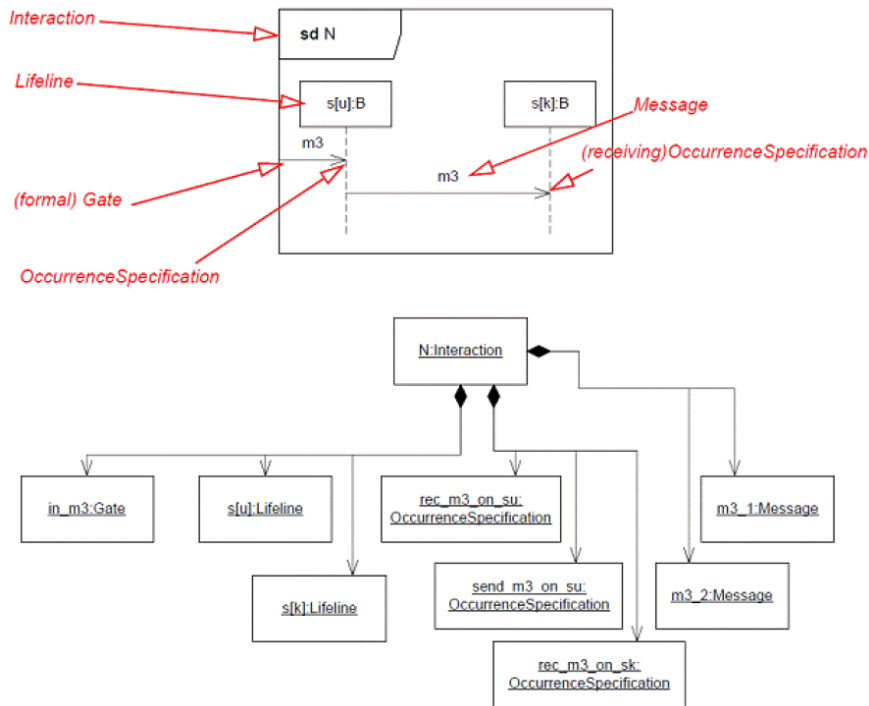
FIGURE 5.10 – Diagramme de séquence décrivant, dans l'ordre réel d'exécution, les activités du système d'aide à la décision à concevoir

les fonctions et leurs solutions de conception, il est nécessaire de décrire la façon et l'ordre d'exécution permettant de répondre à l'objectif principal, exactement comme il a été énoncé. À cette fin, a été rigoureusement détaillée l'architecture *processus et scénarios* du système pour définir qui fait quoi et quand. Ayant défini l'architecture modulaire de notre système et la façon dont ces modules communiquent, il reste maintenant à décrire le fonctionnement interne de chacun de ces modules. Ceci fera l'objet des deux chapitres suivants :

- Modules de modélisation des processus et de traçabilité,
- Module d'aide à la décision.



5.11.a. Partie du méta-modèle du diagramme de cas d'utilisation UML [OMG, 2011]



5.11.b. Partie du méta-modèle du diagramme de séquence UML [OMG, 2011]

FIGURE 5.11 – Méta-modèles des diagrammes de cas d'utilisation et de séquence UML montrant leurs concepts et les relations entre eux

6

CHAPITRE

Modules de modélisation des processus et de traçabilité des décisions

"We are a mission-driven company. In order to do this, we have to build a great team. And in order to do that, you need people to know they can make a bunch of money. So we need a business model to make a lot of money."

Mark Zuckerberg

6.1	Module de modélisation des processus et des données produit	103
6.1.1	Introduction : pourquoi avons-nous besoin de modéliser ?	103
6.1.2	État de l'art sur les techniques de modélisation des processus	105
6.1.2.1	Comment est définie la modélisation des processus dans notre contexte de recherche ?	105
6.1.2.2	Travaux de recherche ayant contribué à la méta-modélisation des processus	107
6.1.3	État de l'art sur les techniques de modélisation des données produit	108
6.1.3.1	Comment est définie la modélisation des données produit dans notre contexte de recherche ?	108
6.1.3.2	Quels sont les travaux de recherche qui ont contribué à la méta-modélisation des produits ?	108
6.1.4	Synthèse et discussion de l'état de l'art sur la modélisation des processus et des données produit	110
6.1.4.1	Concernant le choix du langage de modélisation	110
6.1.4.2	Concernant le choix du méta-modèle de processus	110
6.1.4.3	Concernant le choix du méta-modèle des données produit	112
6.1.5	Proposition : méta-modèle PP de processus-produit	114
6.1.5.1	Identification des différents cas d'utilisation du méta-modèle PP	115
6.1.5.2	Sémantique du méta-modèle PP	117
6.2	Module de traçabilité des décisions	120
6.2.1	Introduction	120
6.2.2	État de l'art sur les techniques de traçabilité	121
6.2.2.1	Comment est définie la traçabilité des décisions dans notre contexte de recherche ?	121
6.2.2.2	Travaux de recherche ayant contribué à la méta-modélisation des traces	123
6.2.3	Synthèse et discussion de l'état de l'art sur les techniques de traçabilité	125
6.3	Conclusion	126

Les entreprises opèrent rarement au maximum d'efficacité et d'efficience dans le cadre socio-économique actuel. Pour s'assurer qu'elles fonctionnent aussi bien comme elles devraient potentiellement l'être, les responsables d'entreprise doivent constamment suivre les performances de leurs entreprises (en termes de chiffre d'affaire, nouveaux produits, maîtrise du temps et des ressources, etc.) et réévaluer, améliorer et parfois même retravailler complètement leurs processus. La modélisation et la traçabilité des processus sont deux disciplines agissant en synergie pour offrir à l'entreprise ce "kick-start" vers la maîtrise de ses activités pour être au-dessus du lot et s'envoler vers le succès industriel.

Ce chapitre, dont la structure est présentée dans la Figure 6.1, a pour objectif de décrire deux des modules identifiés dans le chapitre précédent. Il s'agit des modules de modélisation des processus et de traçabilité dont chacun fera l'objet d'une étude séparée (la communication entre ces deux modules a été décrite dans le chapitre précédent Chapitre 4). Cette étude pragmatique commence d'abord par explorer les travaux de recherche pourront répondre à la fonction du module pour ensuite sélectionner le travail le plus pertinent ou, le cas échéant, proposer la solution la plus adaptée au contexte de recherche. Chaque étude est clôturée par une synthèse qui sera reprise à la fin pour être discutée avant de conclure le chapitre.

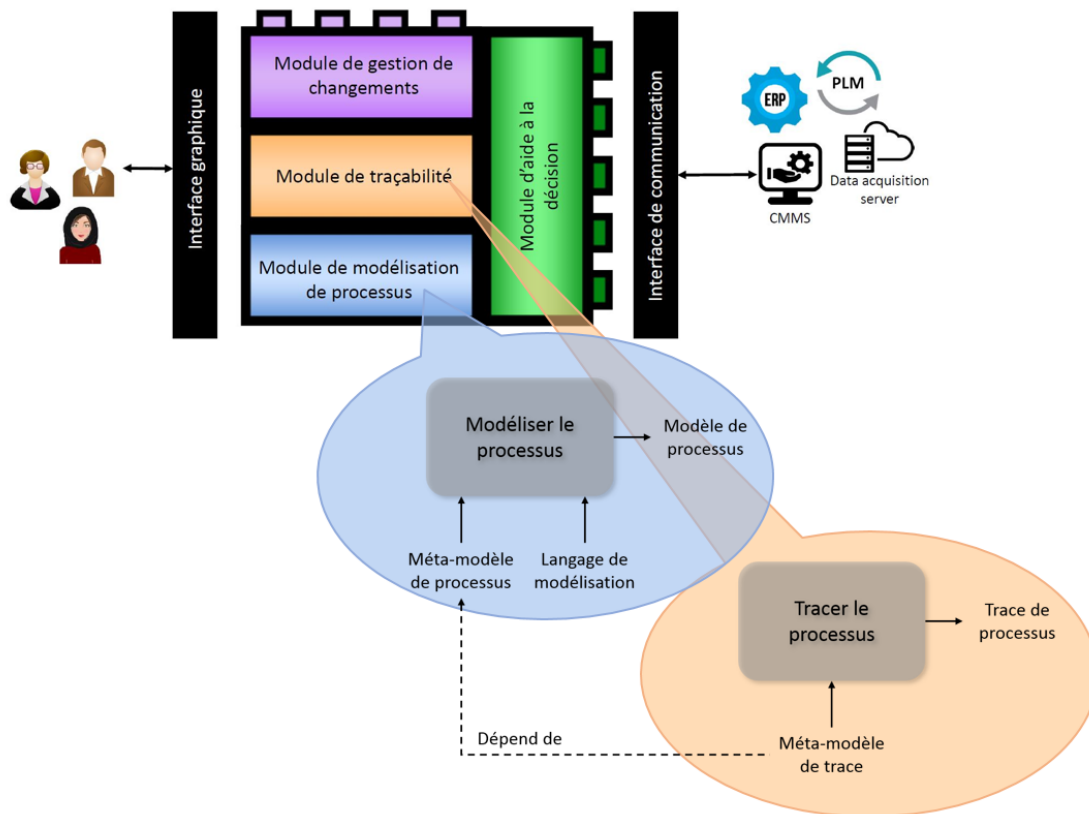


FIGURE 6.1 – Structure du Chapitre 6

6.1 Module de modélisation des processus et des données produit

6.1.1 Introduction : pourquoi avons-nous besoin de modéliser ?

Werner Heisenberg⁹⁰, physicien allemand ayant apporté des contributions fondamentales à la mécanique quantique et à la physique des particules, avait bien raison en disant que *”Not only is the Universe stranger than we think, it is stranger than we can think.”*. Il est en effet difficile pour l’être humain de comprendre la réalité caractérisée par une complexité dépassant ses capacités cérébrales. De ce fait, il est primordial de simplifier cette réalité en la décrivant par des modèles, simples mais avec un niveau suffisant de détails, comme l’a énoncé Albert Einstein⁹¹, mathématicien et physicien allemand pionnier de la théorie de la relativité : *”Everything should be made as simple as possible, but not simpler”*.

Étant très utilisés dans les sciences fondamentales depuis des siècles, les modèles sont de plus en plus utilisés dans les différentes disciplines de l’ingénierie également. Un modèle est défini par Morris [1969] et Lave and March [1975] comme étant une structure idéationnelle provisoire utilisée comme un appareil d’essai pour tester le comportement du système modélisé (à concevoir). La définition la plus simple et communément utilisée pour un modèle est qu’il est une représentation simplifiée de la réalité excluant les détails inutiles, visant ainsi à rendre les systèmes complexes plus compréhensibles [Fischer, 1991]. C’est en effet une abstraction permettant de faciliter la conception et/ou gestion de ces systèmes ou processus complexes, en les décomposant en une hiérarchie d’abstractions [Pels, 2006]; cette notion d’abstraction est définie par Smith and Smith [1977] comme étant un processus qui permet de cacher intentionnellement les détails d’un modèle dans le but de se concentrer sur les éléments les plus pertinents par rapport à des fins spécifiques.

Nécessité croissante de modéliser les processus métier : Personne ne peut douter que le développement des entreprises actuelles soit, en partie, déterminé par leur interaction avec (et leur impact sur) un nombre important de parties prenantes dans un environnement commercial en évolution rapide, poussé par de nouvelles technologies, des concurrents plus agiles, entre autres. Le besoin des investisseurs de mieux comprendre le modèle économique d’une entreprise n’a, ainsi, jamais été aussi pressant. Ce modèle, appelé *”business model”* par les experts du métier, permet de capturer les performances de l’entreprise et son comportement en termes de comment elles créent de la valeur et comment elles résilient aux changements, entre autres, lui permettant ainsi de comprendre et d’améliorer leurs processus. En effet, la modélisation des processus - définie par Bubenko and Kirikova [1995] comme étant un ensemble de techniques utilisées pour représenter et structurer les connaissances métier au sein d’une organisation - a trois objectifs comme les décrivent Havey [2005]; Fasbinder [2007] :

- **Documentation** : elle est assurée par un analyste d’affaires qui maîtrise comment les processus fonctionnent, son objectif est de :
 - Capturer les processus pour que le personnel de l’entreprise, ainsi que les personnes qui lui sont externes, comprennent comment ces processus fonctionnent et la manière dont les activités se déroulent du début à la fin,
 - Identifier les personnes impliquées et dans quelle étape du processus elles le sont,
- **Amélioration** : Après avoir eu les modèles des processus, l’analyste technique (maîtrisant à la fois les besoins de l’entreprise et les systèmes I/T) les examine à partir de différents points de vue à savoir, entre autres, un point de vue orienté donnée, un point de vue orienté organisation et un point de vue orienté contrôle [Becker et al., 2000] puis, éventuellement, les re-conçoit pour réduire les pertes d’efficacité et des coûts et pour répondre plus rapidement aux demandes des clients. De cette analyse, il serait aussi possible de concevoir de nouveaux processus en réutilisant un ou plusieurs processus déjà existants.

90. https://www.nobelprize.org/nobel_prizes/physics/laureates/1932/heisenberg-bio.html

91. https://www.nobelprize.org/nobel_prizes/physics/laureates/1921/einstein-bio.html

- **Automatisation** : Après avoir capturé (et éventuellement amélioré) le processus, l'agent I/T de l'entreprise ajoute une nouvelle couche d'amélioration consistant à automatiser le processus au maximum en réduisant ou éliminant le travail manuel, pour lui permettre ainsi d'être exécuté plus rapidement et à moindre coût.

Vers la modélisation des données produit : Les entreprises d'aujourd'hui cherchent désormais à ouvrir leurs portails à l'international en devenant des entreprises globales⁹² où pratiquement tout le travail est distribué et coordonné dans le monde entier [Isaksson et al., 2000]. De plus, ces entreprises cherchent la satisfaction de leurs clients en proposant une grande quantité de produits dont chacun peut avoir une configuration spécifique répondant aux besoins individuels des clients. Le développement de produits dans ce contexte global - favorisant à la fois la collaboration et une production de masse - nécessite, en plus d'un modèle compréhensible de processus, que le produit soit rigoureusement décrit dans son contexte et que les données produit (ou encore informations produit, comme préfère l'appeler Scholz-Reiter and Krause [2001]), qui le définissent, soient bien capturées dans un modèle de produit et communiquées entre les personnes, les organisations et les outils logiciels durant tout le cycle de vie produit. À cette fin, et pour gérer les produits configurables où un petit changement impacte tout le produit et supporter les processus opérationnels au sein d'une organisation (Figure 6.2), les techniques de modélisation des données produit se voient très importantes et avantageuses.

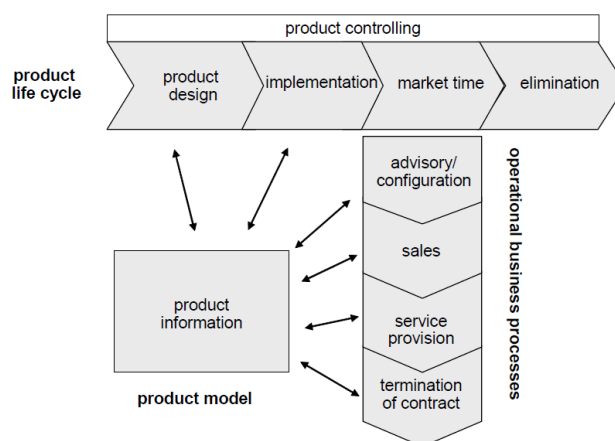


FIGURE 6.2 – Utilisation des modèles de produit pour supporter les processus opérationnels durant le cycle de vie produit [Scholz-Reiter and Krause, 2001]

Les données produit sont définies comme étant une collection de tous les objets de la base de données décrivant un produit industriel (fichiers CAO, modèles 3D/2D, documents texte, etc.) et produits par des applications différentes et hétérogènes tout le long du cycle de vie produit [Bellatreche et al., 2006]. La modélisation dans ce contexte permet d'avoir une vision claire sur le produit en définissant la forme de ses données et les relations entre elles. Elle est définie par Shaw et al. [1989] comme étant une classe de modèles de données sémantiques prenant en compte les besoins des données d'ingénierie et permettant de concevoir la structure de donnée du produit pour une base de donnée spécifique, souvent liée à l'organisation ayant besoin de l'information [Bekke, 1992].

Ayant introduit et défini les notions de modélisation des processus et des données produit, nous souhaitons parcourir, dans les deux sous-sections suivantes, les différents travaux de recherches sur ce sujet pour sélectionner la ou les solutions les plus adaptées à notre contexte de recherche.

92. Il est important de noter que la notion d'entreprise globale est différente de celle d'entreprise multinationale. Les deux types d'entreprise sont présents et investissent dans de nombreux pays, la différence majeure est que le premier type est plus axé sur l'adaptation de ses produits/service à chaque marché local, tandis que le deuxième commercialise ses produits en utilisant la même image/marque sur tous les marchés. Le lecteur peut se référer au travail de Azevedo and Bertrand [2000] pour plus de détails.

6.1.2 État de l'art sur les techniques de modélisation des processus

6.1.2.1 Comment est définie la modélisation des processus dans notre contexte de recherche ?

Dans notre contexte, explicitement décrit dans la Section 2.2.3, l'objectif de la modélisation des processus est de capitaliser (capturer ou encore enregistrer) les informations permettant de décrire, d'une façon exhaustive, le processus industriel en question pour pouvoir retrouver les informations correspondantes à n'importe quelle étape du processus. Ceci revient à décrire le sextuplet QQQCOP : **Qui ? a fait Quoi ?**, **Quand ? Comment ?**, **Où ?** et **Pourquoi ?**, comme décrit par le framework de Zachman [1987] (présenté dans la Figure 6.3) originellement destiné à l'architecture des entreprises puisqu'il permet de définir et visualiser une entreprise d'une façon formelle et structurée selon plusieurs perspectives. Il est, en effet, considéré par certains chefs d'entreprise comme un ensemble de règles idéal pour la gestion des entreprises complexes et en évolution.

Pour modéliser nos processus, en adoptant la logique de Zachman [1987], nous devons identifier deux concepts comme le montre la Figure 6.4 :

- Un **méta-modèle de processus** qui a pour but d'identifier les concepts que nous voulons capitaliser. C'est en d'autres mots, la projection des 6Ws de Zachman [1987] sur notre contexte de travail,
- Un **langage de modélisation** qui permet de modéliser (représenter) graphiquement les concepts du méta-modèle de processus.

	What (Data)	How (Function)	Where (Locations)	Who (People)	When (Time)	Why (Motivation)
Scope {contextual} Planner	List of things important to the business	List of processes that the business performs	List of locations in which the business operates	List of organizations important to the business	List of events/cycles important to the business	List of business goals/strategies
Enterprise Model {conceptual} Business Owner	e.g. Semantic Model	e.g. Business Process Model	e.g. Business Logistics System	e.g. Workflow Model	e.g. Master Schedule	e.g. Business Plan
System Model {logical} Designer	e.g. Logical Data Model	e.g. Application Architecture	e.g. Distributed System Architecture	e.g. Human Interface Architecture	e.g. Process Structure	e.g. Business Rule Model
Technology Model {physical} Implementer	e.g. Physical Data Model	e.g. System Design	e.g. Technology Architecture	e.g. Presentation Architecture	e.g. Control Structure	e.g. Rule Design
Detailed Representation {out-of-context} Subcontractor	e.g. Data Definition	e.g. Program	e.g. Network Architecture	e.g. Security Architecture	e.g. Timing Definition	e.g. Rule Definition
Functioning System	e.g. Data	e.g. Function	e.g. Network	e.g. Organization	e.g. Schedule	e.g. Strategy

FIGURE 6.3 – Les six concepts (désormais notés 6Ws) de Zachman [1987], décrits par Scott Ambler, coach et consultant sénior : <http://enterpriseunifiedprocess.com/>

Méta-modèle de processus : Le modèle de processus, comme nous l'avons décrit auparavant, doit capturer les 6Ws de Zachman [1987] (le sextuplet QQQCOP projeté sur notre contexte). Ces concepts ainsi que les relations entre eux doivent être décrits à un niveau d'abstraction plus élevé appelé méta-modèle. L'OMG a défini le standard MOF (Meta-Object Facility) conçu sous la forme d'une architecture à quatre niveaux d'abstraction selon lesquels le système d'intérêt peut être modélisé, comme le montre la Figure 6.5 :

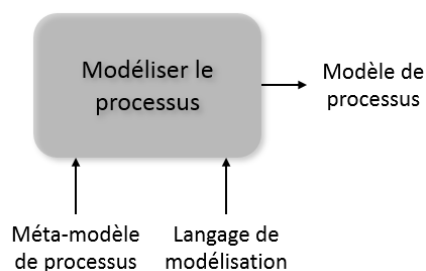


FIGURE 6.4 – Représentation IDEF0 de l'activité de modélisation des processus

- **M0 (couche réalité)** : C'est le système réel (physique) que nous souhaitons concevoir (La maison réelle, en bas dans la Figure 6.5),
- **M1 (couche modèle)** : C'est un modèle, parmi d'autres, décrivant le système réel pour permettre sa conception (le modèle dessiné sur papier dans la Figure 6.5 décrit la forme de la maison à construire),
- **M2 (couche méta-modèle)** : C'est le méta-modèle définissant les concepts qui sont présents dans le modèle et les relations entre eux (le méta-modèle de la maison définit trois formes géométriques (Figure 6.5), qui sont utilisées par le modèle : un triangle pouvant être associé à un rectangle, qui est peut être associé à lui même (pour définir la porte)),
- **M3 (couche méta-méta-modèle)** : Il s'agit ici du MOF qui est le méta-méta-modèle définissant les concepts présents dans le méta-modèle. Il est basé sur la notion de classe, association et héritage (pour définir les concepts du méta-modèle de la maison (dessiner les formes géométriques), il faut un papier, un crayon et une gomme comme le montre la Figure 6.5). Le méta-méta-modèle de l'environnement Eclipse, que nous avons choisi d'utiliser, s'appelle **ECORE**.

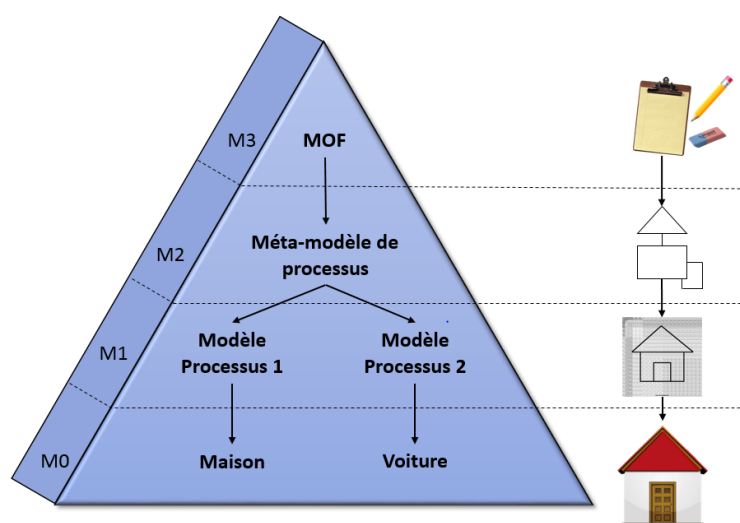


FIGURE 6.5 – Représentation des quatre couches du standard MOF

Langage de modélisation : Un langage de modélisation est utilisé pour spécifier, visualiser, construire et documenter un système logiciel [Rumbaugh et al., 1999]. Cette définition est étendue par Fowler [1997] qui considère que les langages de modélisation sont utilisés pour décrire les concepts dans le domaine du problème étudié. Cette dernière définition nous semble plus adaptée à notre contexte, puisque nous cherchons un langage de modélisation qui répond parfaitement à l'objectif de la modélisation des processus ; qui est la capture et la description des 6Ws.

Un langage de modélisation, tout comme un langage de programmation informatique, doit être pratique, utilisable et d'une valeur durable [Paige et al., 2000]. Nous cherchons, par conséquent, un langage qui comprend :

- **Un vocabulaire** : c'est l'ensemble des concepts permettant de décrire notre problème,
- **Une notation** : c'est la représentation graphique de ces concepts. Nous cherchons une notation intuitive pour faciliter la compréhension des modèles,
- **Un méta-modèle** : c'est la définition de l'organisation des concepts du langage et les relations entre eux. Nous cherchons un langage décrit par un méta-modèle rigoureusement défini.

Il existe une multitude de langages de modélisation permettant de modéliser les processus métier, nous citons, entre autres, IDEF0, IDEF3, BPMN, UML.

6.1.2.2 Travaux de recherche ayant contribué à la méta-modélisation des processus

En tapant "*business process modelling*" sur un moteur de recherche d'article scientifiques, comme Google Scholar⁹³, vous aurez environ 2,070,000 résultats en moins de 0.2 secondes. Il existe, en effet, plusieurs articles proposant des approches de modélisation des processus dont une bonne partie se base sur des méta-modèles de processus. Cette diversité qui est certes le résultat de la différence des contextes de recherche, entre autres, peut être maîtrisée en proposant des frameworks de benchmarking permettant de classer et comparer les méta-modèles existants dans la littérature puis identifier les différents méta-modèles abstraits pouvant être personnalisés après selon le contexte de chaque équipe de recherche. Le but de cette partie est loin d'être une comparaison rigoureuse de tous les méta-modèles existants, nous ne sélectionnons en effet que ceux que nous supposons être les plus compatibles avec notre contexte de recherche. Les concepts, du méta-modèle, qui nous intéressent le plus sont encadrés en rouge et sont discutés dans la partie synthèse (Section 6.1.4).

Nowak et al. [2004] ont proposé le méta-modèle PPO (Processus, Produit, Organisation), partiellement basé sur la méthodologie GRAI (Graphs with Results and Actions Inter-related) [Suparno and Carrie, 1997]. Ce méta-modèle, présenté dans l'annexe 2, décrit la relation entre les données produit, les processus dans lequel ces données transitent et les organisations dans lesquelles les processus s'exécutent. Les travaux de thèse de Ouertani [2007] se sont aussi intéressés à la modélisation des processus et la traçabilité en vue de la gestion des conflits (annexe 3) en proposant l'approche DEPNET (product Data dEPendencies NETwork identification and qualification) ayant pour but de supporter la gestion de conflits en se basant sur la gestion des dépendances entre les données du processus de conception de produit. Couturier et al. [2014] proposent un modèle de donnée conceptuel ayant pour but d'évaluer et tracer les décisions prises dans un environnement mécatronique et leurs conséquences (annexe 4). Moones et al. [2015] se sont intéressés à l'identification des concepts clés permettant de capturer la logique de conception (Design Rational) permettant de gérer les changements (annexe 5). Labrousse and Bernard [2008] a introduit un modèle générique qui permet de modéliser les objets de l'entreprise, notamment les objets PPRE (Produit, Processus, Ressource et Effets externes ou contraintes), selon les points de vue Fonctionnel, Comportemental et Structurel (annexe 6). Heidari et al. [2011] proposent un méta-modèle (annexe 7) permettant d'identifier les concepts relatifs aux processus métier et les dimensions de qualité correspondants, pour permettre une évaluation de la qualité des processus métier. Thom et al. [2005] proposent un méta-modèle transactionnel (annexe 8) permettant de modéliser les concepts des processus métier en identifiant ses différentes tâches, leur ordre d'exécution, les ressources utilisées, etc. Martins and Zacarias [2015] proposent le méta-modèle BPPAM (Business Process and Practice Alignment Methodology) de trois couches : service, structure et action, comme présenté dans l'annexe 9. Ce méta-modèle a la particularité de définir les concepts du processus métier en se basant sur la pratique quotidienne dont les informations sont décrites par les concepts de la couche action (Figure 9.b de l'annexe 9).

Les méta-modèles de quelques langages de modélisation permettent, également, de capturer les informations liées aux processus métier. Le méta-modèle de BPMN [OMG, 2013] permet de capturer une partie des 6Ws (annexe 10). De même, le langage de modélisation UML [OMG, 2011], à travers son diagramme

93. <https://scholar.google.fr/>

d'activité associé à d'autres diagrammes de structure et de séquence, permet de modéliser une partie des concepts relatifs aux processus métier. Un processus est représenté par un système composite (Diagramme de structure composite) et contient des sous-processus. Selon leur nature, les activités sont représentées par des actions et les données d'entrée et de sortie sont respectivement représentées par des ports d'entrée et de sortie. Les contraintes sont explicitement représentées mais les ressources ne sont pas encore clairement définies. Le formalisme de modélisation IDEF0, dont la représentation graphique est décrite dans l'annexe 11, permet de capturer une bonne partie des informations relatives aux 6Ws.

6.1.3 État de l'art sur les techniques de modélisation des données produit

6.1.3.1 Comment est définie la modélisation des données produit dans notre contexte de recherche ?

Le processus de modélisation des données, au sens large du terme (les données produit y comprises), peut impliquer la définition de trois modèles de données définis à différents niveaux d'abstraction selon Ribeiro et al. [2015] :

- **Modèle de donnée conceptuel** : il est conçu pour définir les entités (ou les concepts qui représentent les données) et leurs relations, à un niveau d'abstraction très élevé et indépendant à la fois de la technologie choisie pour représenter la donnée et de la façon comment elle sera implémentée. Ce modèle ne décrit pas les détails sur les entités (tels que leurs attributs, types, etc.) et est généralement utilisé pour explorer les concepts abstraits du produit avec les parties prenantes.
- **Modèle de donnée logique** : il s'agit ici d'un raffinement du modèle conceptuel précédent en fournissant plus de détails sur les entités et leurs relations, mais se voit dépendant du choix technologique de la représentation de la donnée (base de donnée relationnelles, fichiers XML, etc.) mais est toujours indépendant de la façon comment cette donnée est implémentée.
- **Modèle de donnée physique** : ce modèle représente visuellement la structure des données telle qu'elle est implémentée. Par conséquent, les entités sont représentées en tant que classes, les attributs sont représentés en tant que propriétés de la classe et ont un type de données donné. Ce modèle dépend de l'implémentation de la technologie choisie.

Nous définissons la modélisation des données, dans notre contexte, comme étant un moyen pour suivre et documenter l'évolution des données produit durant tout le processus de conception ou de supervision d'un produit/système. Ceci revient à dire que pour chaque activité du processus, l'espace des données⁹⁴ résultant est capturé comme le montre la Figure 6.6. À cette fin, nous avons besoin de définir deux concepts :

- **Un méta-modèle de produit** qui identifie toutes les données produit que nous souhaitons modéliser ainsi que les relations entre elles (Figure 6.7). Notons qu'au niveau de ce méta-modèle, doivent aussi être mentionnées les informations liées au processus permettant de donner lieu à ce produit (Figure 6.6).
- **Un langage de modélisation** permettant de représenter graphiquement les concepts du méta-modèle de produit.

Nous avons déjà discuté, dans la Section 6.1.2.1, le notion de langage de modélisation. Il sera donc intéressant d'explorer, par la suite, les travaux de recherche ayant contribué à la méta-modélisation des produits.

6.1.3.2 Quels sont les travaux de recherche qui ont contribué à la méta-modélisation des produits ?

Tout comme les méta-modèles de processus, un grand nombre d'articles scientifiques sont proposés pour définir les méta-modèles de produit dont ceux que nous supposons être les plus adaptés à notre contexte sont cités.

Noël and Roucoules [2008], à travers leur modèle PPO (annexe 2), décrivent à la fois le produit et les données produit comme le contenu de la donnée et sa version entre autres. Le modèle de processus de

94. C'est un ensemble fini de données produit

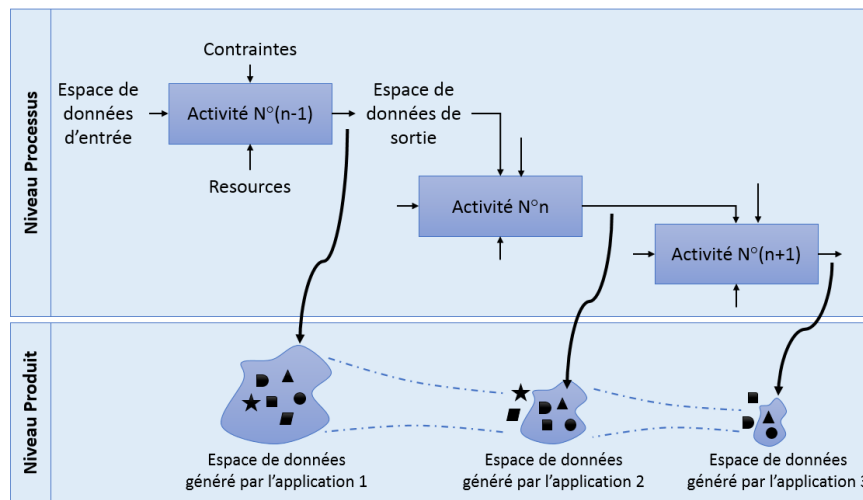


FIGURE 6.6 – Évolution des données produit/système par rapport aux activités du processus de conception/supervision

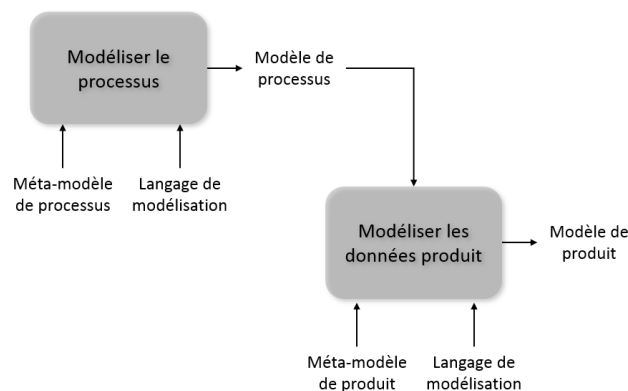


FIGURE 6.7 – Représentation IDEF0 de la modélisation des données produit dans notre contexte

Ouertani [2007] (annexe 3) permet de capturer quelques informations sur les données produit notamment la probabilité de leur modification pour gérer les conflits entre les acteurs de conception lors de l'intégration de leurs propositions. Moones et al. [2015] permet également, à travers leur méta-modèle (annexe 5), de capturer les données produit, notamment leur type et leur niveau de granularité, entre autres. Labrousse and Bernard [2008] ont introduit un modèle générique (annexe 6) pour modéliser les objets d'une entreprise, notamment les données produit, leurs états d'entrée et de sortie et leur structure entre autres. Li et al. [2011] ont proposé un modèle de donnée (annexe 12) basé sur le domaine (dont six sont considérés : domaine du concept design, domaine de la conception structurale, domaine de la conception détaillée, domaine de la fabrication, domaine des ventes et des services). Le mapping entre les modèles de données produit des différents domaines, qui en plus de la garantie de la traçabilité et la consistance des données produit, permet de souligner l'aspect dynamique de l'évolution de ces dernières durant les phases du cycle de vie de produit. Sudarsan et al. [2005] ont décrit un framework de modélisation des données produit (annexe 13) constitué de quatre composants dont chacun se repose sur un méta-modèle décrivant ses éléments et les relations entre eux. Le système se voit comme un serveur de données produit pour les systèmes PLM (Product Lifecycle Management), en ayant la capacité d'accéder, stocker et réutiliser les données produit pendant tout le cycle de vie produit. Lee and Suh [2009] ont développé un modèle de produit pour le système UPLS (Ubiquitous⁹⁵ Product Lifecycle Support) basé sur le paradigme de UbiDM (Design and Manufacturing

95. Ce concept est défini par le Cambridge Dictionary comme : "seeming to be everywhere". En effet, ce sont des systèmes omniprésents intégrés dans l'environnement et dans les objets de tous les jours, au lieu de n'être utilisés que sur des ordinateurs, permettant ainsi une interaction plus naturelle et causale [Carvalho et al., 2003]. Le produit "Apple watch" est un exemple d'une montre basée sur la technologie omniprésente.

via Ubiquitous technology). Le modèle de produit développé (annexe 14) capture à la fois les données produit et l'environnement dans lequel le produit est utilisé permettant, ainsi, de récupérer les informations du produit pendant toutes les phases de son cycle de vie (BOL (Begin-of-Life), MOL (Middle-of-Life) et EOL (End-of-Life)). Wimmer et al. [2003] ont proposé un méta-modèle pour modéliser les produits/services financiers (comme ceux proposés par une banque ou une compagnie d'assurance, par exemple) permettant de faciliter leur implémentation suite aux demandes de personnalisation en masse (annexe 15).

6.1.4 Synthèse et discussion de l'état de l'art sur la modélisation des processus et des données produit

Dans les sections 6.1.2 et 6.1.3, nous avons fait un tour d'horizon sur quelques travaux qui ont contribué à la modélisation des processus et des données produit ayant été définies dans notre contexte comme présenté dans les sections 6.1.2.1 et 6.1.3.1, respectivement. Dans la présente sous-section, nous comparons les travaux étudiés par rapport aux critères que nous avons identifiés dans notre contexte de recherche.

6.1.4.1 Concernant le choix du langage de modélisation

Bien qu'il existe plusieurs langages de modélisation, l'Ingénierie Dirigée par les Modèles (Model-driven engineering - MDE) permet via des transformations, ATL⁹⁶ (ATLAS Transformation Language) entre autres, de passer d'un modèle conçu en utilisant un langage de modélisation IDEF0, par exemple, à un autre conçu en utilisant UML (Figure 6.8). Il nous paraît donc judicieux de ne pas imposer un langage de modélisation à l'utilisateur et de lui laisser le choix de choisir le langage qu'il préférera. Ainsi, il ne nous reste que d'identifier le méta-modèle de processus à adopter pour définir le module de modélisation de processus.

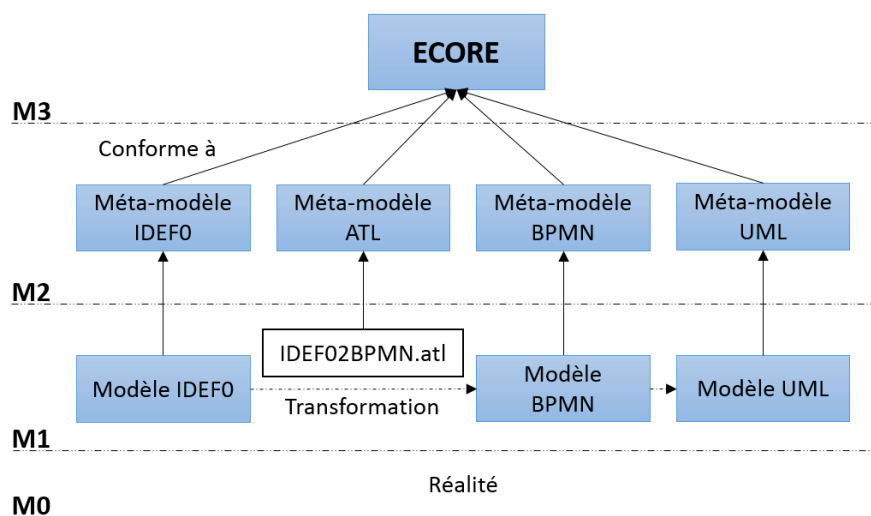


FIGURE 6.8 – Passage automatique d'un langage de modélisation à un autre via des transformations ATL.

6.1.4.2 Concernant le choix du méta-modèle de processus

Pour choisir le méta-modèle le mieux adapté à notre contexte, parmi ceux étudiés dans la Section 6.1.2.2, nous souhaitons comparer ces derniers selon trois points de vue, définis comme suit [Es-Soufi et al., 2017a] :

Point de vue exhaustivité de modélisation : Cet aspect, étant le plus important, concerne l'exactitude et la capacité du méta-modèle à représenter et exprimer les informations (6Ws) qu'on cherche à capitaliser. Les 6Ws sont définis dans notre contexte comme suit :

96. <http://www.eclipse.org/at1/>

- **Qui (Who)** : indique la capacité de modéliser l'acteur qui exécute l'activité en détaillant les informations le concernant ; à savoir, son nom, rôle, compétences, etc. Un acteur est considéré comme étant une ressource humaine.
- **Quoi (What)** : c'est la capacité de décrire les données produit (données d'entrée et de sortie) nécessaires pour exécuter l'activité. Dans le contexte de la conception de produit, ce concept fait référence à l'espace de solutions admissibles en lien avec le produit en cours de conception. Tandis que dans le contexte de la supervision, le Quoi fait référence aux états du système supervisé avant et après la prise de décision.
- **Quand (When)** : ce concept décrit les dates planifiées et réelles du début d'exécution de l'activité ainsi que les dates planifiées et réelles de sa fin.
- **Comment (hoW)** : c'est l'ensemble des ressources (matérielles, logicielles, humaines, etc.) utilisées lors de l'exécution de l'activité.
- **Où (Where)** : ce concept fait référence à l'activité en question pour la différencier des autres activités du processus.
- **Pourquoi (Why)** : c'est la justification derrière tous les choix (ou les décisions) faits durant l'exécution de l'activité. Ces choix peuvent concerner les ressources (choix d'utiliser un logiciel "L₁" parmi tous les autres) ou les solutions admissibles (parmi toutes ces solutions, il est possible de choisir comme solution finale, la solution "S₅" parce qu'elle est moins chère ou parce que le décideur la préfère, tout simplement).

Point de vue représentatif : cet aspect concerne la façon dont le méta-modèle est représenté et sa capacité d'être simple et précis. Dans notre contexte, nous identifions les critères suivants :

- **Simplicité** : elle décrit le niveau de complexité du méta-modèle. Un méta-modèle simple est plus pratique car il est facile à comprendre et facile à modifier au cas où des changements sont identifiés dans l'organisation. La simplicité peut être caractérisée par le nombre des concepts du méta-modèle ainsi que la qualité de leur signification graphique [Daoudi and Nurcan \[2007\]](#).
- **Richesse** de représentation : elle représente la capacité du méta-modèle à représenter les connaissances au sein de l'organisation. Ce critère fait référence au nombre de concepts et leur pouvoir d'expression [Daoudi and Nurcan \[2007\]](#). Un langage est considéré littéralement riche s'il est capable d'être étendu.
- **Norme** : elle introduit la syntaxe et la sémantique qui caractérisent respectivement la règle d'écriture et la signification mathématique des concepts du méta-modèle. Un méta-modèle normé est facilement compréhensible et vérifié.
- **Formalisme/Notation** : il décrit comment sont représentés les concepts du méta-modèle (sous une forme graphique, textuel, équations mathématiques, etc.). Nous souhaitons que le méta-modèle de processus soit exprimé graphiquement pour expliciter les relations entre les différents concepts.
- **Support logiciel** : Il décrit l'existence, ou pas, d'un logiciel supportant le méta-modèle et permettant son implémentation.

Point de vue méthodologique : Cet aspect concerne la démarche systématique du méta-modèle. Dans notre contexte, nous identifions les trois critères suivants :

- **Granularité** : il représente le niveau d'abstraction (appelée aussi décomposition) du processus. En effet, les méta-modèles sont utilisés pour décrire le système en termes d'abstractions et les relations entre elles [Paige et al. \[2000\]](#). Nous cherchons un méta-modèle permettant une description architecturale (granularité) totale en décomposant le processus en sous-processus et activités.
- **Consistance** : ce principe exige que les concepts du méta-modèle aient bien un sens [\[Meyer, 1992\]](#). Par conséquent, les concepts redondants ou irrationnels doivent être éliminés. Dans notre contexte, nous définissons ce critère comme étant la capacité du méta-modèle à décrire notre problème de recherche, en n'introduisant que les concepts intéressants, mais sans l'empêcher d'être extensible (et donc riche). Le but étant d'avoir un méta-modèle simple mais bien adapté à notre contexte.

- **Implémentation** : ce critère fait référence à l'existence d'une implémentation concrète du méta-modèle permettant son test et son exploitation (instanciation, entre autres).

Tous les méta-modèles, présentés dans la Section 6.1.2.2, ont une norme, une notation et permettent d'exprimer graphiquement leurs concepts. Le méta-modèle PPO de Nowak et al. [2004] a une assez bonne capacité de modélisation puisqu'il modélise complètement les concepts Qui, Comment et où. De plus, il est assez simple et riche, consistant et permet une granularité totale. Le méta-modèle de Ouertani [2007] a une bonne capacité de modélisation puisqu'il modélise complètement les concepts Qui, Quand, Comment, Où et Pourquoi. Il est simple, riche, consistant, implémenté mais ne permet pas une granularité totale. Le méta-modèle de Couturier et al. [2014] a une capacité de modélisation limitée puisqu'il ne modélise que le concept QUI. Il est assez simple et riche, inconsistant, implémenté et ne permet pas une granularité totale. Le méta-modèle de Moones et al. [2015] a une très bonne capacité de modélisation car il modélise complètement les concepts Qui, Quoi, Quand, Pourquoi et partie des concept Comment et où ; il est simple, riche, consistant, implémenté mais ne permet pas une granularité totale. Le méta-modèle Labrousse and Bernard [2008] a une capacité de modélisation limitée puisqu'il nous permet de modéliser uniquement les concepts Quoi et Où. Il n'est pas simple mais assez riche, consistant mais ne permet une granularité totale. Le méta-modèle de Heidari et al. [2011] a une capacité de modélisation limitée car il ne modélise que les concepts Où, Quoi et Quand, il est assez simple, assez riche et assez consistant mais ne permet pas une granularité totale. Le méta-modèle de Thom et al. [2005] a également une capacité de modélisation limitée puisqu'il ne permet de modéliser que les concepts Qui, Comment, et Où. Il est simple, assez riche, assez consistant et permet une granularité totale. Le méta-modèle de Martins and Zacarias [2015] a une capacité de modélisation moyenne, il permet de modéliser les concepts Qui, Quoi, Comment et Où tout en étant assez simple, assez riche, assez consistant sans permettre une granularité totale.

Les méta-modèles BPMN et UML ont une assez bonne capacité de modélisation puisqu'ils modélisent les concepts Qui, Quoi, Comment et Où. Ils sont riches et assez constants. Cependant, ils ne sont pas simples. Le méta-modèle IDEF0 a une bonne capacité de modélisation car il modélise complètement les concepts Who, What, How, Where et modélise partiellement le concept Why. Cependant, il ne permet pas une granularité totale. La Figure 6.9 présente graphiquement le résultat de la comparaison de ces méta-modèles par rapport à nos critères de recherche.

Les méta-modèles étudiés ne répondent pas à la totalité de nos besoins puisqu'ils ont été proposés dans des contextes différents (Figure 6.9). Il est donc nécessaire de les étendre quelques uns d'eux pour construire un méta-modèle parfaitement adapté à nos besoins. Nous choisissons d'étendre les méta-modèles IDEF0 et BPMN en spécifiant leurs concepts (par exemple, les ressources IDEF0 sont étendues en spécifiant les ressources humaines, matérielles, logicielles et documentaires et l'entrée BPMN est étendue pour spécifier les entrées, les contraintes et les ressources). Nous étendons également les méta modèles de Nowak et al. [2004], Ouertani [2007] et Moones et al. [2015] car ils modélisent une grande partie des concepts 6Ws tout en étant simples, riches, cohérents avec la capacité d'exprimer la granularité totale d'un processus.

6.1.4.3 Concernant le choix du méta-modèle des données produit

Afin de comparer les méta-modèles de données produit étudiés, nous définissons les deux critères suivants :

- Capacité de **gérer la version** de la donnée produit : une activité peut s'exécuter autant de fois que possible, nous souhaitons donc avoir un méta-modèle de produit permettant de noter (capturer) la version des données produit après chaque ré-exécution de l'activité afin de permettre le suivi de l'évolution des données. Prenons comme exemple une activité produisant un document donné ; lors de la ré-exécution de cet activité, le nouveau document doit avoir une nouvelle version et ne doit pas écraser l'ancien.
- Capacité de **lier la donnée produit au processus** produisant le produit : chaque donnée est, en effet, générée par une activité du processus en utilisant une ressources donnée (logicielle entre autres), nous souhaitons donc que le méta-modèle mette en évidence la relation entre la donnée produit et l'activité correspondante. Ayant l'activité en question, il est facile de retrouver les ressources utilisées car ces deux concepts (ressource et activité) sont liées dans le méta-modèle de processus.

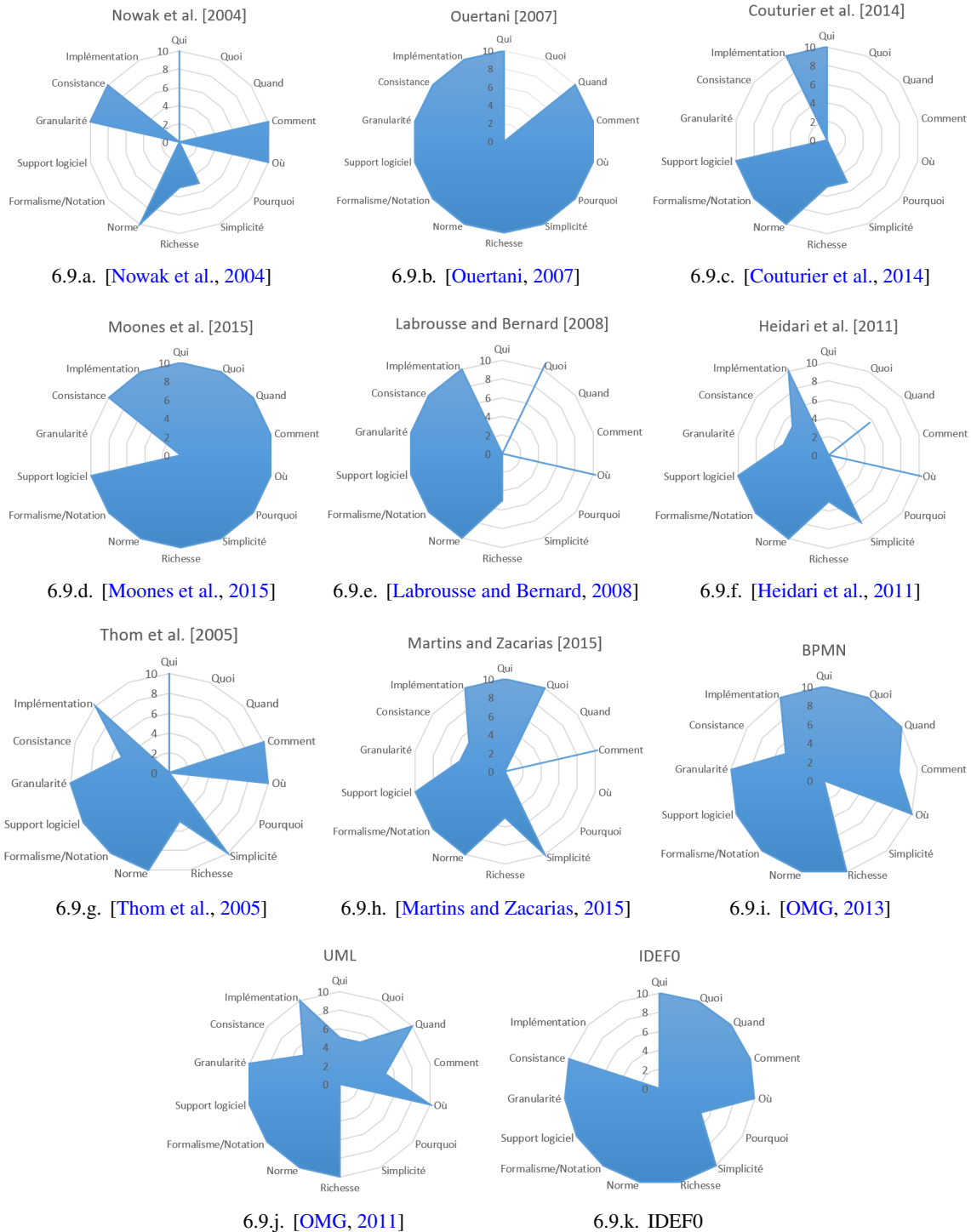


FIGURE 6.9 – Comparaison des différents méta-modèles étudiés par rapport à nos critères de recherche. Le méta-modèle que nous cherchons doit occuper toute la surface du radar.

Les méta-modèles de Noël and Roucoules [2008], Ouertani [2007], Moones et al. [2015] et Lee and Suh [2009] répondent parfaitement à nos deux critères. Ils permettent, en effet, de capturer la version de la donnée produit à travers les concepts "version number, propriété de la classe Product Data (versioned)", "Version, propriété de la classe Donnée Produit", "Version, propriété de la de la classe Producty_Data" et "Product_version", respectivement. De plus, ils ont la capacité de lier la donnée produit à l'activité en question à travers les concepts "Project", "Action_de_Conception", "Design_Activity" et "Activity_as_realized_arm.Actual_activity", respectivement. En revanche, les méta-modèles de Labrousse

and Bernard [2008] et Wimmer et al. [2003] ne répondent pas à nos deux critères de choix ; et ceux de Li et al. [2011] et Sudarsan et al. [2005] répondent à un de nos deux critères. La représentation graphique de la comparaison des méta-modèles de produit est présentée sur la Figure 6.10.

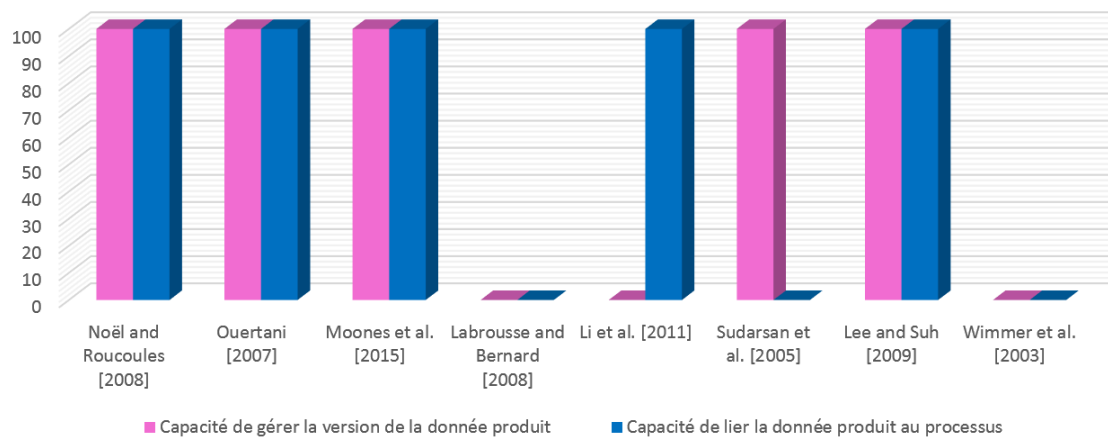


FIGURE 6.10 – Résultat de la comparaison des méta-modèles de données produit par rapport à nos critères de recherche

Il est, par conséquent, clair que nous devrions choisir un des méta-modèles de produit proposés par Noël and Roucoules [2008], Ouertani [2007], Moones et al. [2015] et Lee and Suh [2009]. Cependant, nous distinguons deux types de données produit dans le monde industriel : les données produit fermées dont la structure n'est pas dévoilée (vues comme une boîte noire, Figure 6.11.b) et les données produit ouvertes dont la structure est dévoilée (vues comme une boîte blanche, Figure 6.11.a). En effet, si la structure des données produit est connue, il est possible de décrire, dans un modèle, la relation entre les différents espaces de données résultants (espaces issus des activités 1, 2 et 3 dans la Figure 6.11.a), et il sera, par la suite, possible de propager les contraintes⁹⁷ (ou n'importe quel changement) d'une façon automatique. Ceci est loin d'être le cas parce que dans la plupart des applications métier utilisées dans l'industrie, les données produit sont ambiguëment structurées, ou ont une structure exprimée à un très haut niveau d'abstraction (comme dans le cas des logiciels payants par exemple) et sont générées, le long du processus, par différentes applications, souvent hétérogènes et parfois non interopérables. Par conséquent, il est difficile d'identifier les relations logiques (ou les modèles de données structurés) qui lient l'espace des données résultant d'une première activité à celui résultant d'une deuxième ; ces espaces sont donc considérés comme étant des images figées et indépendantes (Figure 6.11.a) ; capturant l'état de la sortie de chaque activité.

Pour pallier ce problème et garder l'historique des données produit résultantes, nous pensons à capturer l'état de sortie de chaque activité, et d'y accéder via une des URI (chemin vers la donnée indiquant où elle se trouve sur le disque dur) par exemple. Cet URI sera ensuite intégrée dans le méta-modèle de processus pour que les données produit soient accessibles et exploitées par toutes les activités du processus. De plus après chaque nouvelle exécution de l'activité, et si son statut indique qu'elle a été validée (pour faire référence à la fin de l'exécution de l'activité en cours), le nom du fichier des données produit s'incrémente et sera ensuite sauvegardé dans un répertoire de données produit accessible à tout moment.

6.1.5 Proposition : méta-modèle PP de processus-produit

L'objectif de cette section est de construire un méta-modèle de processus-produit adapté à 100% aux besoins que nous impose notre contexte de recherche. Pour cela, nous suivons la démarche suivante, basée sur le langage UML [OMG, 2011] :

- **Définir les scénarios possibles** du méta-modèle du point de vue de l'utilisateur (diagramme de cas d'utilisation UML),

⁹⁷. En utilisant des techniques de satisfaction des contraintes comme la technique CSP (Constraint Satisfaction Problem) [Kumar, 1992], entre autres.

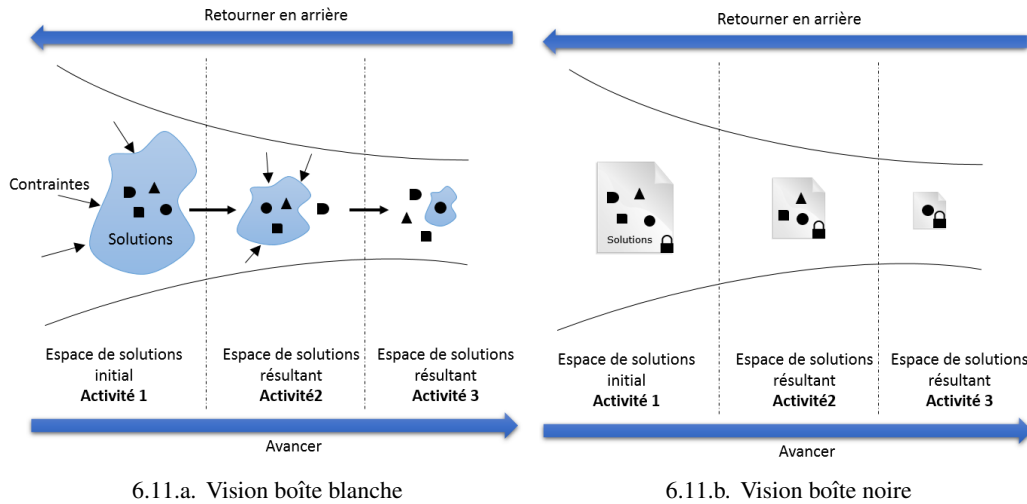


FIGURE 6.11 – Les deux types de données produits identifiés dans les milieux industriels et scientifiques

- **Construire le méta-modèle** en définissant sa structure statique en termes de classes et de relations entre classes (diagramme de classe UML).

6.1.5.1 Identification des différents cas d’utilisation du méta-modèle PP

Dans le cadre de la *conception* et la *supervision* des systèmes, nous identifions les différents cas d’usage qu’un utilisateur pourrait rencontrer lors de la création de son processus (Figure 6.12) :

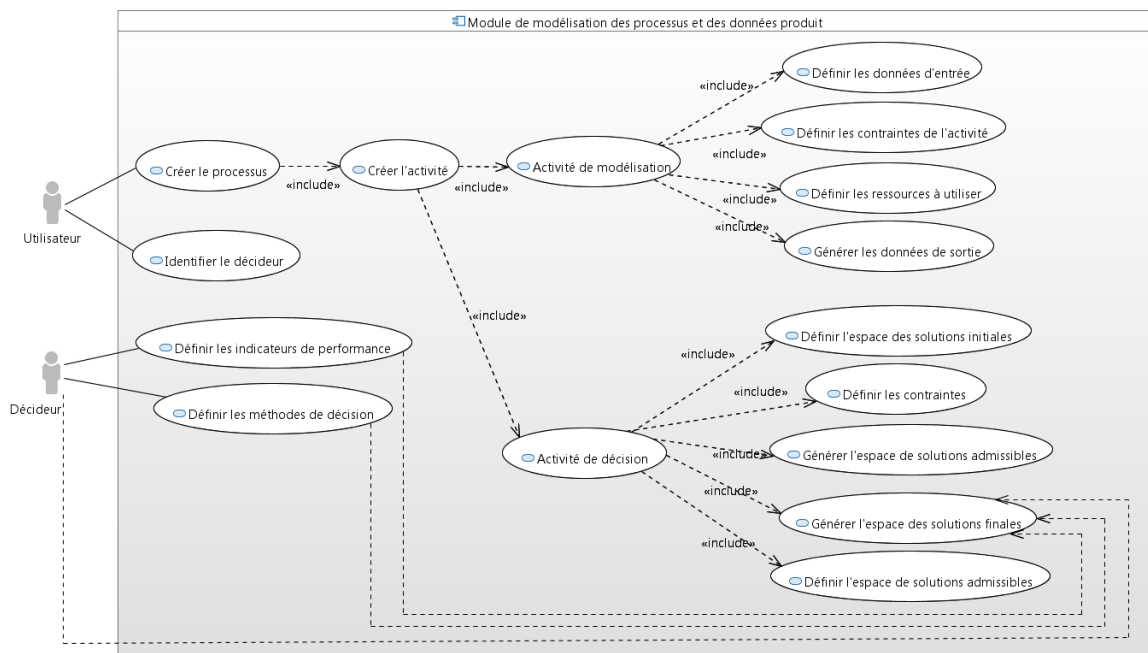


FIGURE 6.12 – Les différents cas d’usage lors de la modélisation des processus

- Il commence d’abord par la création du processus pour ensuite procéder à la création des activités, une par une,
- Selon son besoin et conformément à notre contexte de recherche, l’utilisateur peut créer soit une activité de *modélisation* (lors de la conception des systèmes) soit une activité de *décision* (lors de la conception et/ou supervision des systèmes). En effet, l’activité de modélisation (Figure 6.13) a

pour but de transformer les données d'entrée, en se servant des ressources nécessaires (logicielles, matérielles, etc.) en données de sortie qui satisfont les contraintes imposées. Une activité de décision, dont la logique est totalement différente de celle de l'activité de modélisation, peut se projeter sur deux niveaux différents N_1 et N_2 , parfois complémentaires. En effet, l'utilisateur peut exécuter une activité de décision pour choisir l'ensemble des solutions qui répondent à ses contraintes comme le montre la Figure 6.14.a. Le résultat de l'activité, vue à ce niveau N_1 , est l'ensemble de toutes les solutions admissibles qui satisfont les contraintes imposées sans identifier les solutions optimales. Cette optimisation est, en effet, abordée par le deuxième niveau N_2 de l'activité de décision, qui en tenant compte des critères de performance et/ou des choix subjectifs des utilisateurs, génère en sortie l'ensemble des solutions finales considérées comme optimales par rapport au contexte de l'utilisateur Figure 6.14.a.

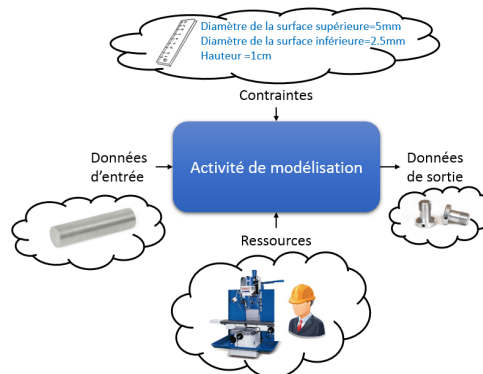
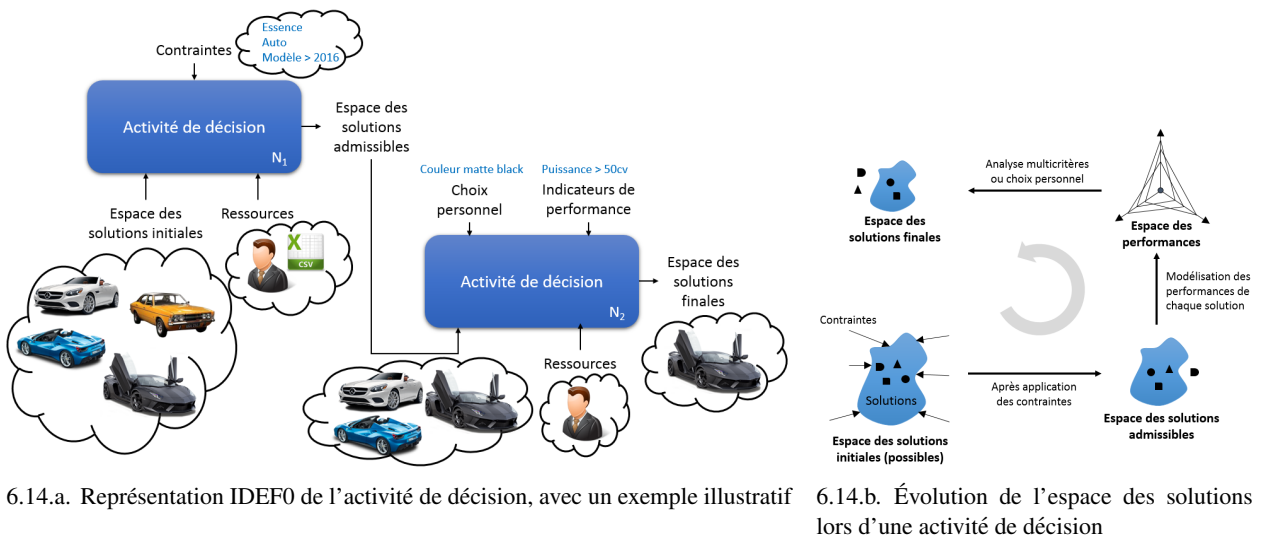


FIGURE 6.13 – Représentation IDEF0 de l'activité de modélisation, avec un exemple illustratif



6.14.a. Représentation IDEF0 de l'activité de décision, avec un exemple illustratif

6.14.b. Évolution de l'espace des solutions lors d'une activité de décision

FIGURE 6.14 – Représentation IDEF0 de l'activité de décision et évolution de l'espace des solutions correspondant

- Si l'utilisateur choisit de créer une activité de modélisation, il doit ensuite identifier (comme le montre la Figure 6.13) les données d'entrées, les contraintes à satisfaire, les différentes ressources à utiliser pour supporter l'exécution de l'activité, et doit ensuite générer les données de sortie. Cette génération peut être automatique (filtrage des données d'entrées par rapport aux contraintes en utilisant les macros VBA sur Excel) ou manuelle (car parfois la contrainte n'est pas bien définie sous la forme d'équations et/ou repose sur l'expertise humaine et n'est donc pas complètement automatisable).
- Dans le cas où l'utilisateur choisit d'exécuter une activité de décision et s'il s'agit d'une décision du niveau N_1 comme décrit sur la Figure 6.14.a, il doit identifier l'espace des solutions initiales

(rassemblant toutes les solutions possibles) et les contraintes pour générer l'espace des solutions admissibles (c'est-à-dire qui satisfont les contraintes). Lorsqu'il s'agit d'une décision du niveau N_2 (Figure 6.14.a), l'utilisateur doit identifier deux concepts. Le premier est l'espace des solutions admissibles qui sera considéré comme une ressource parce qu'il ne s'agit pas ici d'une transformation réelle, de l'espace d'entrée en espace de sortie, comme la montre l'activité de la Figure 6.13 dans laquelle nous comprenons que la barre en Aluminium est physiquement transformée en deux clous. Le deuxième concept est l'espace des solutions finales généré par l'activité de décision du niveau N_2 . Cet espace dépend de l'utilisateur et des indicateurs de performance choisis (car la performance de chacune des solutions est étudiée par rapport aux indicateurs de performance définis) et est généré par l'activité en utilisant comme ressource une des techniques de décisions existantes (analyse multi-critères, entre autres). La Figure 6.14.b montre l'évolution de l'espace des solutions durant une activité de décision et met en évidence la complémentarité des niveaux de décision N_1 et N_2 .

6.1.5.2 Sémantique du méta-modèle PP

En se basant sur les différents scénarios possibles, identifiés dans la Figure 6.12, et les définitions des activités de modélisation et de décision (Figure 6.13 et Figure 6.14), nous proposons le méta-modèle PP de la Figure 6.15. Ce méta-modèle exprime les différents concepts identifiés sous la forme de classes UML et leurs caractéristiques sous la forme d'attributs de classe UML. Le premier concept à créer, selon la Figure 6.12, est le **processus** qui est considéré comme la *racine* (point de départ) de notre méta-modèle dont les différentes classes sont décrites comme suit :

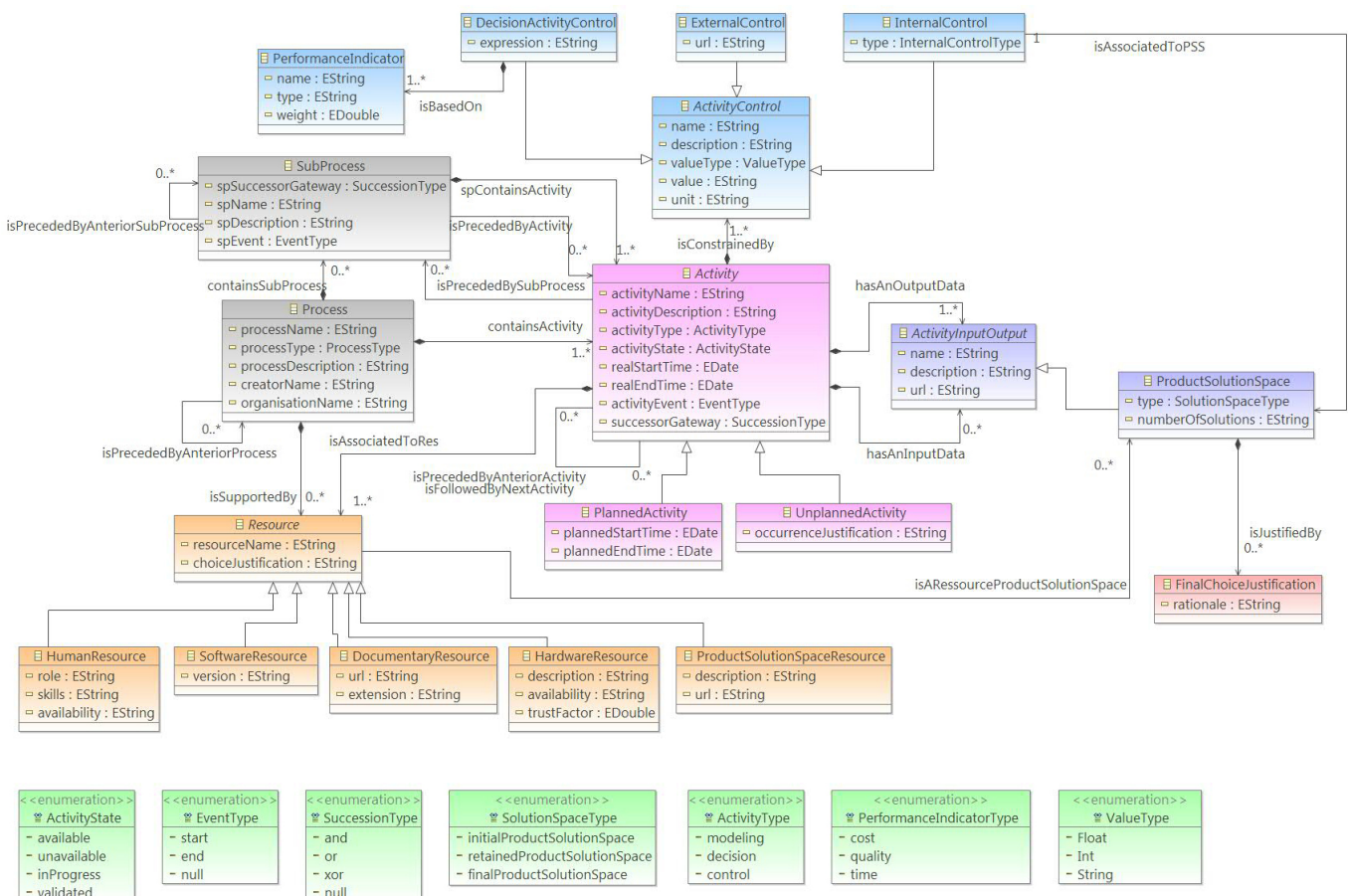


FIGURE 6.15 – Proposition : méta-modèle PP de modélisation des processus et des données produit

Classe Process (Processus) : Le processus, ayant été déjà défini dans la Section 2.2.3, est modélisé par une classe UML nommée Process contenant les attributs que nous jugeons nécessaires pour caractériser le

processus en question :

- `processName` : c'est le nom du processus,
- `processDescription` : c'est la description de l'objectif de l'exécution du processus,
- `organisationName` : c'est le nom de l'organisation dans laquelle le processus est en train d'être exécuté,
- `creatorName` : c'est le nom de l'utilisateur qui crée le processus.

Dans une organisation, un processus peut être exécuté seul comme il peut être exécuté en parallèle (ou en séquence) avec plusieurs autres, il est donc intéressant de localiser le processus en question en notant celui qui le précède (relation `isPrecededByAnteriorProcesses` dans la Figure 6.15). En outre, un processus peut contenir des activités comme il peut contenir des sous processus qui, eux même, contiennent des activités séparées selon leurs tâches, la relation `containsSubProcesses` dans la Figure 6.15 permet d'exprimer l'existence des sous-processus dans un processus; elle se voit comme une association particulière de composition, indiquant qu'un sous-processus n'existe que si le processus existe déjà (un sous-processus est en effet une partie d'un processus).

Classe Activity (Activité) : Une activité, définie dans la Section 2.2.3, est modélisée par une classe UML contenant un certain nombre d'attributs :

- `activityName` : c'est le nom de l'activité en question,
- `activityDescription` : c'est le détail concernant l'objectif de l'activité, sa tâche, etc.
- `activityType` : précise le type de l'activité en question. Nous avons, en effet, défini deux types d'activités : modélisation (Figure 6.13) et décision (Figure 6.14.a),
- `activityState` : précise l'état actuel de l'activité (disponible ou pas pour l'exécution, en cours d'exécution ou validée),
- `realStartTime` : est le temps réel du début de l'exécution de l'activité,
- `realEndTime` : est le temps réel de la fin de l'exécution de l'activité,
- `activityEvent` : c'est l'événement qui précède l'activité. Nous distinguons trois types d'événements : *start*, *end* et *null*. Les premières activités du processus auront comme événement "*activityEvent=start*", les dernières activités du processus auront pour événement "*activityEvent=end*" et les activités intermédiaires auront comme événement "*activityEvent=null*",
- `successorGateway` : c'est une entité qui identifie la nature de la liaison entre l'activité actuelle et l'activité qui suivra. Nous distinguons trois types de jonction (appelée aussi gateway, porte, fonction ou connecteur) : *and*, *or* et *xor* comme le décrit le Tableau 6.1.

Nom de la jonction	Type	Description
And	Convergente	Toutes les activités en amont s'achèvent
	Divergente	Toutes les activités en aval commencent
Or	Convergente	Une ou plusieurs activités en amont s'achèvent
	Divergente	Une ou plusieurs activités en aval commencent
Xor	Convergente	Une seule activité en amont s'achève
	Divergente	Une seule activité en aval commence

TABLE 6.1 – Types des différentes jonctions successeurs

La Figure 6.16 présente un exemple de processus où on peut trouver toutes les jonctions décrites dans le Tableau 6.1 :

- La jonction And_1 (resp. And_2 et And_3) est une jonction divergente indiquant que les activités (A_1, A_2 et A_8) (resp. (A_6, A_7) et (A_9, A_{10})) peuvent s'exécuter.
- La jonction And_4 est une jonction divergente donc les activités (A_9, A_{10}) commencent leur exécution.

- La jonction Or_1 est divergente donc elle indique qu'une ou plusieurs activités du groupe (A_4, A_5) commencent, c'est-à-dire soit A_4 s'exécute toute seule, soit A_5 s'exécute toute seule, soit les deux s'exécutent en parallèle. Le raisonnement reste le même pour la jonction Or_2 , le sous-processus SP_1 et l'activité A_{11} .
- La jonction Xor_1 est convergente, donc la première activité (parmi A_7 et A_5) qui termine l'exécution valide la jonction et les activités suivantes commencent leur exécution. Le raisonnement est le même pour la jonction Xor_2 , la première activité qui finit son exécution parmi (A_{12}, A_{11}) valide la jonction et déclenche l'exécution de l'activité A_{13} .

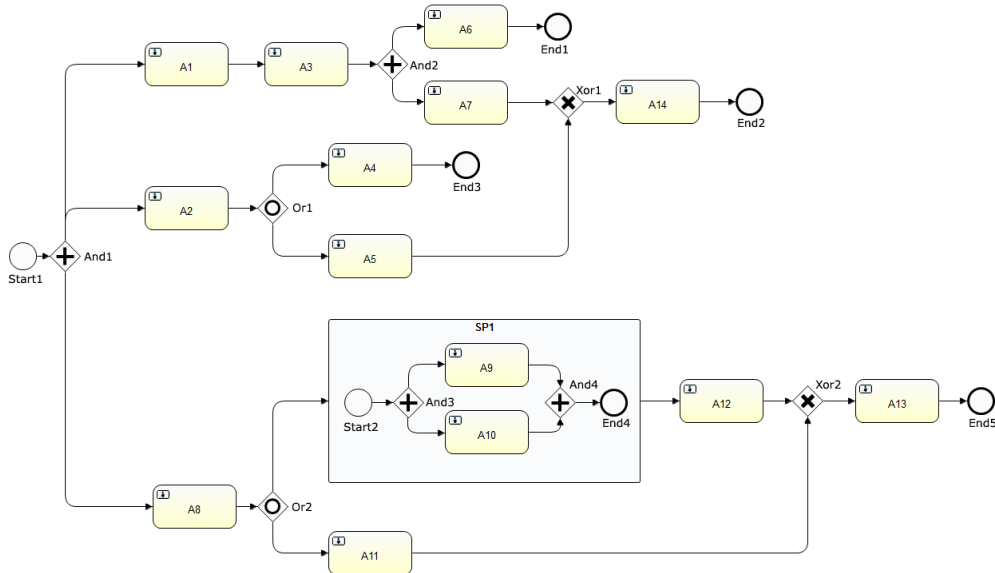


FIGURE 6.16 – Exemple de processus contenant les jonctions *And*, *Or* et *Xor*

Une activité peut être déjà planifiée par le concepteur (*PlannedActivity*) comme elle peut parvenir en temps réel au cours de l'exécution du processus (*UnplannedActivity*). Les deux activités sont représentées par deux classes UML héritant de la classe activité mère :

- Si l'activité est planifiée, nous devons mentionner le temps planifié de début de l'exécution (*plannedStartTime*) et le temps marquant sa fin (*plannedEndTime*),
- Si l'activité est non planifiée, nous devons justifier la raison de son apparition dans le processus en cours d'exécution (*occurrenceJustification*).

Une activité peut s'exécuter toute seule dans un processus (ce qui est rarement le cas), comme elle peut s'exécuter avec plusieurs autres activités. Il est intéressant de référencer l'activité en question en notant la ou les activités qui précèdent (*association isPrecededByAnteriorActivities*).

Classe *ActivityInputOutput* (données produit d'entrée et de sortie) : Une activité possède des entrées et des sorties. Ces deux dernières sont représentées par une seule classe UML abstraite *ActivityInputOutput* contenant trois attributs :

- *name* : c'est le nom de la donnée d'entrée ou de sortie,
- *description* : pour introduire ce que sont ces données, à quoi servent elles, etc.
- *url* : c'est l'emplacement du fichier contenant les données produits.

La classe *ProductSolutionSpace*, qui hérite de la classe *ActivityInputOutput*, peut être une entrée ou une sortie d'une activité de conception (modélisation et décision) et elle contient les attributs suivants :

- *type* : c'est la nature de l'espace de solutions. En effet, comme le montre la Figure 6.14.b, nous distinguons trois types d'espace de solutions, le premier étant l'espace de toutes les solutions possibles qui alimentent l'entrée d'une activité de modélisation ou la ressource d'une activité de décision.

Ce premier espace sera réduit à cause des contraintes appliquées sur l'activité, donnant lieu ainsi à l'espace des solutions admissibles. Toutes les solutions de l'espace admissibles peuvent être prises en considération parce qu'elles satisfont les contraintes, mais parfois nous cherchons à choisir la solution la plus performante, plus efficace et/ou plus efficiente. De ce fait, l'espace des solutions admissibles se transforme, après une étude de performance, à l'espace des solutions finales (Figure 6.14.b).

- `numberOfSolutions` : c'est le nombre des solutions contenues dans l'espace des solutions. Ce paramètre est introduit pour distinguer les données produits des activités de modélisation et de décision. En effet une activité de modélisation prend en entrée un espace de solution d'une seule solution (donnée produit d'entrée) tandis que l'activité de décision prend comme ressource un espace contenant au moins deux solution.

Classe Ressources (ressources) : Les ressources sont l'ensemble des mécanismes mis à disposition pour réaliser l'activité Gingele et al. [2002]. Une ressource est représentée par une classe UML précisant le nom de la ressource et la justification de choix de cette dernière. De cette classe dérivent plusieurs types de ressources :

- `Ressources humaines` : c'est l'ensemble des personnes exécutant l'activité, dont chacun a un rôle dans l'organisation, des compétences, et un état de disponibilité.
- `Ressources logicielles` : c'est l'ensemble des logiciels utilisés lors de l'exécution de l'activité ainsi que leurs versions.
- `Ressources documentaires` : c'est l'ensemble des documents servant de support pour l'exécution de l'activité, dont l'url (chemin d'accès sur le disque dur ou un lien web) est noté pour pouvoir retrouver le logiciel en question à tout moment. L'extension du document généré par la ressource logicielle est aussi notée pour des fins de précision.
- `Ressources matérielles` : c'est l'ensemble des machines/appareils utilisés lors de l'exécution de l'activité. Chaque machine est décrite et sa disponibilité est notée.
- `Espace de solutions produit` : dans le cas d'une activité de décision, une ressource peut aussi être un espace de solutions initial ou admissible, comme expliqué dans Figure 6.14

Classe ActivityControl (contraintes) : Les contrôles sont l'ensemble des contraintes imposées sur l'activité. Ils sont représentés par une classe UML précisant leurs noms et descriptions, et sont de trois types :

- `Contrôles internes` : ce sont les contrôles imposés par les autres activités (postérieures ou antérieures) appartenant au même processus (l'association `isAssociatedToPSS` relie le contrôle à la donnée de sortie appelée `ProductSolutionSpace` d'une activité quelconque).
- `Contrôles externes` : ils peuvent être exprimés par l'ensemble des spécifications (exigences) imposées par le cahier des charges.
- `Contrôles de décision` : ils peuvent être des contrôles normaux comme ceux appliqués sur les activités de modélisation, comme ils peuvent être des contrôles de performance définis par 0 ou plusieurs performances attendues (car c'est optionnel), et reposant sur un (au moins) ou plusieurs indicateurs de performances pondérés (coût, qualité, délai, entre autres).

6.2 Module de traçabilité des décisions

6.2.1 Introduction

En septembre 2007, *Mattel Inc*⁹⁸ -multinationale américaine de fabrication de jouets- a été appelée à payer une pénalité civile de 2,3 millions de dollars suite à la violation du standard interdisant l'utilisation des

98. <https://www.mattel.com/en-us>

peintures au plomb⁹⁹ considérées comme toxiques; et a été appelée à retirer du marché des millions de ses jouets, en plus des 21 millions déjà retirés pour des défauts de conception. Le fabricant de la poupée Barbie a estimé le coût de ses pertes à 28,5 millions de dollars, qui bien qu'il soit négligeable au regard de ses profits (5,6 milliards de dollars en 2006), il pourrait connaître une hausse importante suite à la perte de confiance des consommateurs. La multinationale Samsung, elle aussi, n'a pas échappé à la conception défectueuse et la mal-supervision de ses produits. En effet, la première et deuxième version du *Note 7* ont posé un problème de batterie¹⁰⁰ menant à une explosion dont les dégâts enregistrés sont d'ordre humain. *Mattel Inc* et *Samsung* ne sont pas les seules entreprises à proposer des produits défectueux, en effet, les services après-vente des entreprises traitent quotidiennement les plaintes des clients concernant les produits achetés [Kvarnström, 2008]. Ceci montre que les conséquences (résultats) des choix de décision ne sont pas analysés par les concepteurs avant qu'ils soient pris en considération, et ont donc mené à des produits défectueux, ou déviés¹⁰¹. Cette analyse de choix décisionnels n'est en revanche pas évidente parce que ces derniers, qui sont aussi à l'origine des produits excellents, sont étroitement liés au contexte dans lequel la décision est prise. En effet, les contraintes économiques peuvent évoluer, de nouvelles technologies peuvent apparaître, de nouvelles exigences ou normes peuvent s'imposer, etc. Par conséquent, la notion de traçabilité des décisions se voit très importante pour :

- Identifier les causes de la déviation des produits,
- Identifier les autres produits susceptibles d'être affectés par le produit dévié ou par un problème identifié,
- Identifier Qui est responsable de quoi,
- Retrouver les décisions qui ont été prises, dans quel contexte et pourquoi elles l'ont été,
- Identifier les opportunités de changements positifs dans les caractéristiques des produits,
- Permettre l'amélioration continue des produits et des processus,
- Améliorer la prise de décision, etc.

6.2.2 État de l'art sur les techniques de traçabilité

6.2.2.1 Comment est définie la traçabilité des décisions dans notre contexte de recherche ?

Plusieurs définitions sont proposées pour introduire ce qu'est la traçabilité, dont le rôle a été largement reconnu dans la conférence parrainée par le comité scientifique de l'OTAN (Organisation du traité de l'Atlantique nord) [Naur and Randell, 1969]. Cette notion est définie par l'ISO [1994] comme étant "*the ability to trace the history, application or location of an entity by means of recorded identifications*". Ríos [2001] définit la traçabilité comme étant "*related to carrying out a study or to the history of something*" et souligne que bien que cette notion dépende du domaine d'étude, elle a pour but principal de décrire les caractéristiques du produit/système tracé. Matta et al. [2016] définissent la traçabilité comme étant un moyen pour suivre et/ou identifier l'impact du développement d'un objet quelconque sur son environnement. La définition la plus fondamentale de la traçabilité, selon Cleland-Huang et al. [2012], est qu'elle est le potentiel de relier les données qui sont stockées, avec la possibilité d'examiner cette liaison entre les données. Par conséquent, la capacité à réaliser cette fonction de traçabilité dépend de l'enregistrement des données d'intérêt et de la création des liens navigables entre elles. Matta et al. [2016] identifient deux types de traçabilités, la traçabilité des informations et la traçabilité des connaissances, cette dernière se voit comme l'historique de l'utilisation des données et des informations par l'utilisateur pour exécuter une activité donnée, en exprimant le quoi, comment et pourquoi du raisonnement. La traçabilité est souvent liée à la notion de suivi (tracking en anglais); cette dernière est définie comme étant le processus par lequel un produit est suivi à partir du point de départ au point actuel du processus tandis que la traçabilité est vue comme le processus inverse du suivi, en étant le parcours inverse du chemin depuis son point actuel jusqu'à

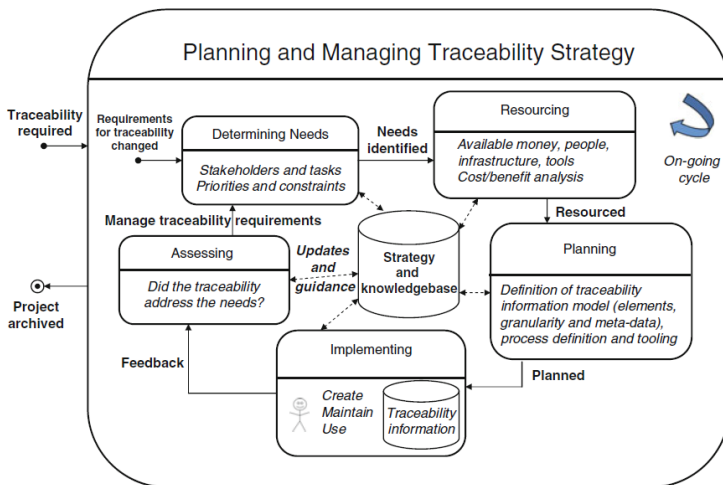
99. <https://money.cnn.com/2009/06/05/news/companies/cpsc/>

100. <https://www.forbes.com/sites/aarontilley/2017/01/22/samsung-note-7-fire-investigation-battery/#4fbc8f442017>

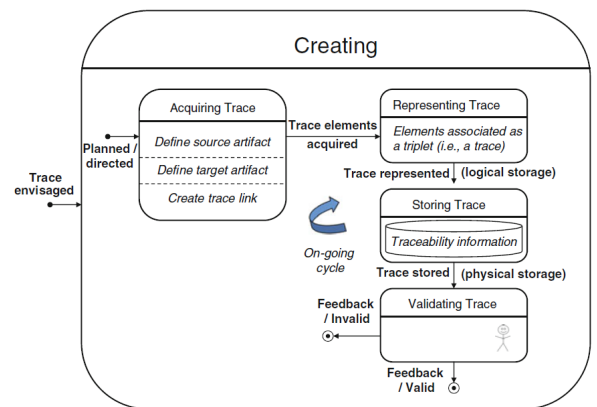
101. Nous appelons déviation l'écart entre le produit fabriqué et le produit souhaité

son point de départ [Pizzuti et al., 2014]. Quatre grandes phases sont identifiées lors de la définition du processus de traçabilité [Cleland-Huang et al., 2012] :

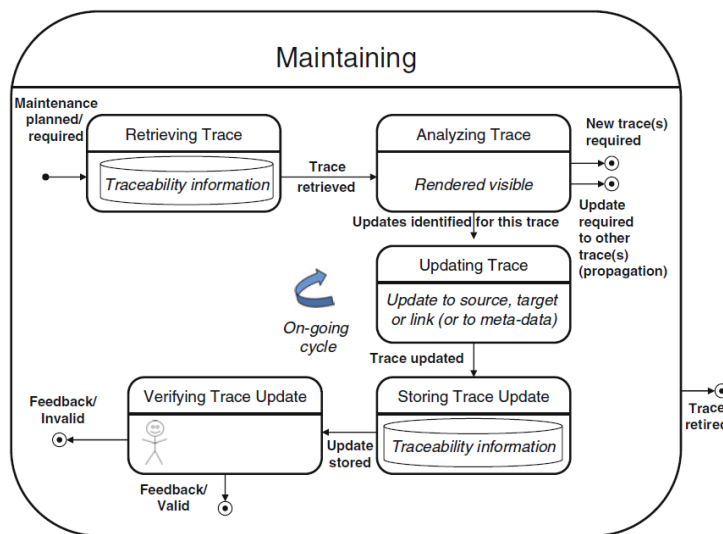
- **Planification de la stratégie de traçabilité** : il s’agit ici de déterminer les exigences des parties prenantes en matière de traçabilité, puis concevoir une solution de traçabilité appropriée. La stratégie de traçabilité comprend la planification et la gestion de la traçabilité Figure 6.17.a.
- **Création de la trace** : il s’agit ici d’associer deux objets (ou plus) par des liens de trace (Figure 6.17.b). La création de trace peut se faire manuellement, automatiquement ou semi-automatiquement, et peut nécessiter des annotations supplémentaires pour caractériser les attributs des traces.
- **Maintenance de la trace** : la traçabilité est censée évoluer en concordance avec l’évolution de son contexte. Elle doit, ainsi, être maintenue et mise à jour en ajoutant/supprimant les objets ou les liens de trace entre eux pour adapter la stratégie au nouveau contexte Figure 6.17.c.
- **Utilisation de la trace** : il s’agit ici de l’utilisation des traces collectées pour supporter les activités de l’utilisateur, comme la vérification et la validation ou encore l’analyse de l’impact et la gestion des changements Figure 6.17.d.



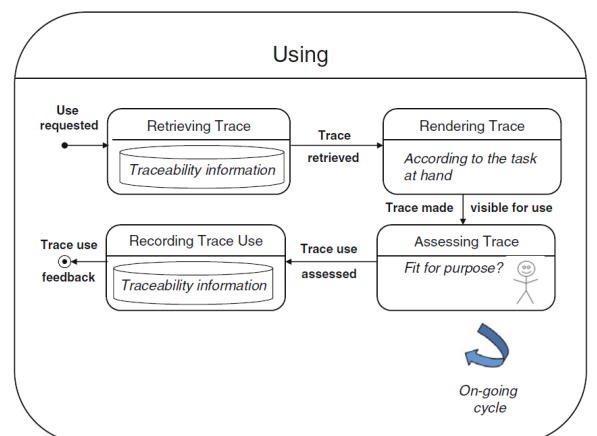
6.17.a. Phase de planification de la stratégie de traçabilité



6.17.b. Phase de création de la trace



6.17.c. Phase de maintenance de la trace



6.17.d. Phase de l’utilisation de la trace

FIGURE 6.17 – Processus de traçabilité selon Cleland-Huang et al. [2012]

Dans notre contexte de recherche, nous nous intéressons au processus de traçabilité et notamment à la phase de création de la trace (les phases de maintenance et d'utilisation seront assurées par le module d'aide à la décision qui sera présenté dans le chapitre suivant). Nous introduisons la traçabilité dans notre contexte, en reprenant les définitions proposées par [Matta et al. \[2016\]](#) et [Cleland-Huang et al. \[2012\]](#), comme étant :

Un moyen pour capturer, d'une façon exhaustive, et enregistrer, d'une façon digitale, toutes les décisions prises (en les reliant par des liens navigables selon la logique de [Zachman \[1987\]](#)) ainsi que le contexte dans lequel elles ont été prises lors de la conception et la supervision des systèmes. De plus, la traçabilité sera un moyen pour capturer et enregistrer la logique de conception (Design rationale), étant la justification de tous les choix faits lors de la conception d'un produit/système, notamment, les raisons justifiées derrière la prise de décision et les autres solutions admissibles [[Lee and Lai, 1991](#)]. Elle sera aussi un moyen pour capturer et enregistrer la logique de supervision (pouvant avoir la même définition de la logique de conception, en considérant cette fois-ci la supervision comme processus).

Le but de ce travail de recherche est loin d'être un débat sur la différence entre information, donnée et connaissance, le lecteur peut se référer au travail de [Bellinger et al. \[2004\]](#) où la différence entre les trois notions est explicitée, nous voulons en effet tracer les décisions qui se voient plutôt un raisonnement basé principalement sur les connaissances. Pour tracer ces décisions dans notre contexte, nous avons besoin de deux éléments, comme le montre la Figure 6.18 :

- Un **méta-modèle de trace** ayant pour but d'identifier les concepts (6Ws de [Zachman \[1987\]](#)) que nous souhaitons tracer exactement comme nous l'avons décrit dans la Section 6.1.4.2. Le méta-modèle de trace est un sous-ensemble du méta-modèle de processus car ce dernier identifie tous les concepts que l'utilisateur peut rencontrer lors de la modélisation de son processus, alors que le méta-modèle de trace n'identifie que les éléments que l'utilisateur souhaite analyser. Cet élément fait référence à la phase de planification de la stratégie de traçabilité de la Figure 6.17.a.
- Un **système ou moteur de traçabilité** permettant de capturer et enregistrer les concepts identifiés par le méta-modèle de trace au cours d'une exécution réelle du processus de conception ou de supervision.

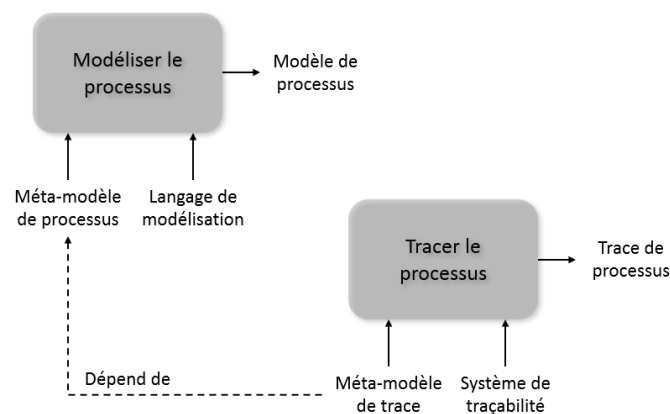


FIGURE 6.18 – Représentation IDEF0 de la traçabilité des décisions dans notre contexte.

La section suivante a pour objectif de présenter les travaux de recherche ayant contribué à la méta-modélisation des traces. Notons que le système de traçabilité peut, en fait, être n'importe quel système d'information permettant de réaliser la fonction de traçabilité dont le méta-modèle, identifiant les concepts à tracer, lui sera communiqué par le système IDSS via son interface de programmation applicative (API - Application programming interface). Notre contribution concernant ce deuxième élément sera présentée plus en détails dans le chapitre suivant.

6.2.2.2 Travaux de recherche ayant contribué à la méta-modélisation des traces

Plusieurs techniques de traçabilité sont proposées dans la littérature dont chacune dépend du domaine d'étude. Dans le domaine de la traçabilité des logiciels par exemple, les techniques basées sur la recherche

d'information¹⁰² (IR - Information Retrieval) [Baeza-Yates et al., 1999; Harman, 1993] se voient très utilisées, elles permettent de créer les liens entre les différents composants du système logiciel, à savoir, le code source, les documents de conception, les document des exigences, les cas de test, reports de bugs, communications entre les parties prenantes, etc. Dans l'industrie alimentaire, plusieurs techniques de traçabilité sont utilisées pour tracer les produits alimentaire "from farm to fork", elles reposent sur l'utilisation des codes bar 1D/2D et des méthodes RFID (Radio Frequency IDentification), entre autres [Comba et al., 2013; Espiñeira and Santaclara, 2016]. Dans le domaine du e-commerce, entre autres, les techniques de profilage des utilisateurs (user profiling) se voient très utilisées pour garder une trace sur le comportement de l'utilisateur afin d'adapter l'environnement pour répondre aux exigences de l'utilisateur [Matta et al., 2016]. Dans le domaine de résolution de problème, la méthode IBIS (Issue-Based Information System) [Conklin and Begeman, 1987], qui se repose sur la notion du Design Rationale [Lee and Lai, 1991], se voit très utilisée pour structurer le problème et permettre la détection de sa cause principale. Dans le domaine de la créativité et l'innovation, la théorie CK (Concept-Knowledge) [Hatchuel and Weil, 2003] permet d'offrir une formalisation du processus de conception et une traçabilité de ses éléments. Il existe plusieurs autres techniques et méthodes de traçabilité appliquées dans différents domaines d'études [Matta et al., 2016; Cleland-Huang et al., 2012]. Dans notre contexte, et conformément à nos besoins de traçabilité discutés dans la Section 6.2.2.1, nous cherchons les techniques de traçabilités basées sur les méta-modèles de trace.

Plusieurs travaux de recherches se sont intéressés à la méta-modélisation des traces, notamment les méta-modèles de Nowak et al. [2004], Ouertani [2007] et Couturier et al. [2014] que nous avons abordés dans la Section 6.1.2.2. Karray [2012] introduit un méta-modèle de trace en vue d'assister la maintenance (annexe 16) dont les classes qui nous intéressent sont entourées en rouge. Shen et al. [2009] proposent un méta-modèle permettant de tracer l'information en allant des exigences jusqu'à l'implémentation du logiciel et en intégrant à la fois les techniques d'implémentation et la logique du métier (annexe 17). Anquetil et al. [2010] proposent le framework ATF (AMPLE Traceability Framework) reposant sur un méta-modèle générique de trace qui, en l'instanciant, permet au framework d'être configuré pour s'adapter au processus de développement en question. ATF permet de définir les artefacts, leur architecture et les types de liens qui les relient ainsi que les contraintes contrôlant les différents types de liens. Maté and Trujillo [2012] proposent un méta-modèle de trace (annexe 19) pour le développement des entrepôts de données basé sur le framework MDA (Model Driven Architecture) de l'OMG. Le méta-modèle permet d'améliorer la validation des exigences et évaluer l'impact des changements en identifiant les objets affectés. Laghouaouta et al. [2017] proposent un méta-modèle de trace pour générer, automatiquement, la trace de la composition des modèles (annexe 20) qui consiste à combiner les différents modèles des systèmes logiciels. Hug et al. [2012] proposent un méta-modèle de trace (annexe 21) se basant sur le formalisme Map. Ce dernier est un formalisme qui permet de décrire les processus d'une façon flexible en présentant ses objectifs et les différentes méthodes permettant de les réaliser [Rolland et al., 1999]. Tang et al. [2007] ont proposé un modèle basé sur la logique qui permet de capturer la logique de conception, les objets de conception et leurs relations, il a défini trois façons pour le faire : tracer en avant pour analyser les impacts, tracer en arrière pour analyser des causes et la tracer l'évolution lors de la conception du produit. Mohan and Ramesh [2007] ont proposé une approche de traçabilité basée sur les concepts QQQCOP en vue d'intégrer la connaissance pour supporter le groupe de décision et de négociation. Pavković et al. [2013] ont proposé un système de traçabilité en vue d'améliorer la communication dans un environnement collaboratif. L'application ProWhy¹⁰³, introduite dans l'ouvrage de [Matta et al., 2016], a été développée pour structurer, normaliser, centraliser et partager les expériences de résolution de problèmes. L'application permet l'exécution de plusieurs fonctions dont une concerne la description du problème pour capitaliser les connaissances en se basant sur un méta-modèle décrivant les 6Ws de Zachman [1987] (Figure 6.19). Il existe d'autres techniques permettant la traçabilité du comportement des utilisateurs; nous citons par exemple les techniques de profiling qui se basent sur les fichiers journaux (log files) [Matta et al., 2016] et qui permettent

102. C'est une science traitant la quête de l'information dans un document, ou la quête des documents, des images, etc.

103. C'est une solution open source basée sur le web : <http://www.prowhy.org/prowhy-site/>

de lier les traces à l'objectif et aux tâches des utilisateurs.

FIGURE 6.19 – Interface Prowhy dédiée à la description du problème (http://www.prowhy.org/prowhy_site/principal/onglet_fonctionalites)

6.2.3 Synthèse et discussion de l'état de l'art sur les techniques de traçabilité

L'outil ProWhy et les techniques de profiling des utilisateurs se voient très intéressants pour capitaliser les connaissances et le comportement des utilisateurs. Cependant, ils ne répondent pas parfaitement à nos besoins. En effet :

- L'outil ProWhy permet de tracer les 6Ws de Zachman [1987] mais ne permet pas de capturer les informations relatives au séquençement d'activités, en d'autres mots, l'information "A₂ vient après A₁ dans le processus", ne peut être capturée.
- Les techniques de profiling se basent sur les logs qui ne sont pas forcément conformes à notre méta-modèle de processus (Figure 6.15) sur lequel nous souhaitons nous baser.

Des autres travaux de recherches étudiés, nous distinguons deux approches de traçabilité :

- **Approche intégrale** : il s'agit ici de tracer toutes les connaissances relatives au processus, notamment les concepts 6Ws comme nous les avons décrits.
- **Approche sélective** : il s'agit ici de ne tracer qu'une partie des concepts 6Ws pour répondre à un objectif donné. Par exemple, tracer les concepts Qui et Quoi, pour savoir, tout le long du processus, qui fait quoi pour des fins d'analyse de la performance humaine, ou encore tracer le Quand, pour calculer le temps nécessaire pour concevoir une voiture par exemple.

Dans notre contexte, nous souhaitons récupérer les traces textuelles contenant toutes les informations relatives aux concepts 6Ws pour permettre au module d'aide à la décision et de gestion des changements d'être le plus efficace possible. En effet, il est plus important pour les techniques de fouille de données et de processus -utilisées par le module de l'aide à la décision- d'avoir un corpus de données important contenant des traces d'exécution complètes, réelles et non pré-traitées, à partir desquelles les informations concernant la dépendance entre les données, les patterns de données, et les règles permettant de les contrôler peuvent être déduites avec un certain niveau de certitude.

Par conséquent, nous considérons le méta-modèle de processus (Figure 6.15) comme le méta-modèle de trace à partir duquel d'autres méta-modèles de trace -dont chacun modélise les concepts à tracer, relatifs à un domaine particulier- peuvent être conçus (Figure 6.20).

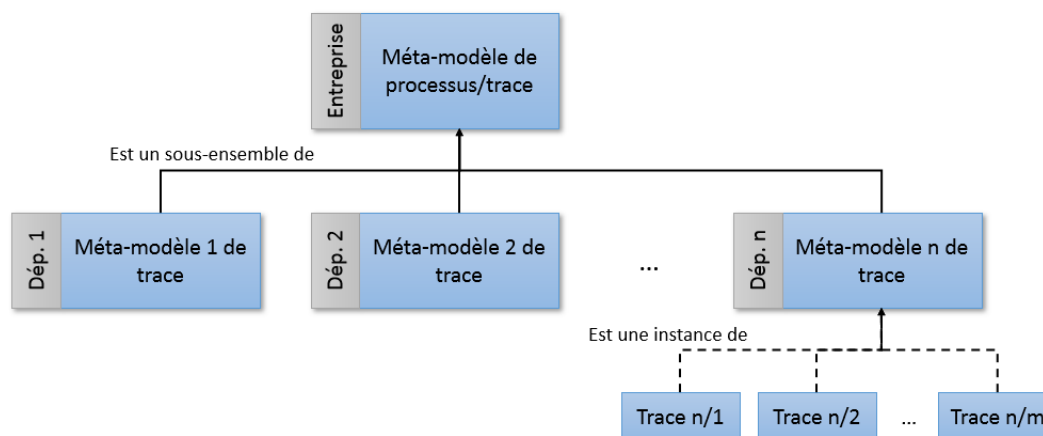


FIGURE 6.20 – Spécialisation des méta-modèles de trace selon le domaine d'étude.

6.3 Conclusion

Pour maîtriser leur complexité, les processus de conception et de supervision des systèmes doivent être modélisés. Cette modélisation qui concerne à la fois les processus et les données produit, étant les données que consomment et génèrent les activités du processus, permet entre autres de structurer les processus métier pour les rendre compréhensibles et archivables et de capturer l'évolution des données produit au cours de l'exécution de ces processus pour pouvoir adapter le produit à de nouveaux contextes ou de réutiliser ses informations pour concevoir un nouveau produit. La première partie de ce chapitre, qui concerne le module de modélisation des processus et des données produit de notre système intelligent d'aide à la décision, a pour but d'établir un méta-modèle PP de processus produit (dont la logique est décrite dans la Section 6.1.5.2) permettant d'identifier les concepts, et leurs relations logiques, nécessaires pour permettre à l'utilisateur du système IDSS de modéliser ses processus dans le contexte de conception et de supervision des systèmes. La traçabilité de ses éléments (concepts et les relations qui les lient) est d'une très grande importance puisqu'elle permettra aux entreprises d'analyser et/ou réutiliser les connaissances précédentes et leur permet ainsi de réduire le temps, l'effort et les coûts lors d'une prochaine conception de produit/système ou supervision de système [Kroll et al., 2011].

7

CHAPITRE

Module d'aide à la décision

"Now, pay attention, 'cause I'm about to tell you the secret to life. You're ready? The whole damn thing is about decisions. Little, seemingly insignificant decisions that clear the road for monster truck life-altering ones. You see, every path you take leads to another choice, and some choices can change everything. Every damn moment of the rest of your life hangs on them. And, boy, do I got a choice I got to make."

Travis Parker

7.1	Introduction : vers des systèmes intelligents d'aide à la décision	129
7.2	Comment est définie l'aide à la décision dans notre contexte de recherche ?	129
7.3	État de l'art sur quelques techniques de l'IA et de la BI supportant la prise de décision	132
7.3.1	Quelle est la différence entre l'intelligence artificielle, la fouille de données, l'apprentissage automatique, la fouille de processus et la business intelligence ?	132
7.3.2	Techniques de fouille de données (Data mining)	134
7.3.2.1	Fouille des règles d'association (Association rule mining)	134
7.3.2.2	Fouille des patterns séquentiels (Sequential pattern mining)	135
7.3.3	Techniques de l'apprentissage automatique (Machine learning)	137
7.3.3.1	Apprentissage supervisé (Supervised learning)	138
7.3.3.2	Apprentissage non-supervisé (Unsupervised learning)	138
7.3.3.3	Apprentissage semi-supervisé (semi-supervised learning)	140
7.3.3.4	Apprentissage par renforcement (Reinforcement learning)	141
7.3.4	Techniques de fouille de processus (Process mining)	143
7.3.4.1	Découverte des processus (Process discovery)	143
7.3.4.2	Contrôle de conformité (Conformance checking)	144
7.3.4.3	Amélioration des processus (Enhancement)	146
7.3.5	Synthèse et discussion de l'état de l'art sur les techniques de l'intelligence artificielle et de la business intelligence supportant la prise de décision	147
7.4	Module d'aide à la décision : Aide à la décision globale et locale	150
7.4.1	Processus de l'aide à la décision et exemple d'illustration	150
7.4.2	Préparation du journal des événements	153
7.4.3	Phase d'exploration des différents patterns existants dans un journal des événements	158
7.4.4	Phase de la découverte des règles de décision	161
7.4.4.1	Sous-phase 1 : découverte du modèle réseau de Pétri à partir du journal des événements	161
7.4.4.2	Sous-phase 2 : fouille des règles de décision (Decision Mining)	165

7.4.5	Prédiction des paramètres des activités les plus adaptés au contexte de la décision	171
7.4.6	Synthèse et conclusion du module d'aide à la décision	174
7.5	Module de gestion des changements contextuels : vers un IDSS dynamique	174
7.5.1	Qu'est ce que la gestion des changements ?	174
7.5.2	État de l'art sur les techniques de gestion des changements contextuels	175
7.5.3	Discussion des techniques de gestion des changements contextuels	175
7.5.4	Proposition : application permettant la gestion dynamique des changements contextuels	176
7.6	Synthèse et conclusion	177

Vous êtes-vous déjà demandé pourquoi, si vous vous concentrez en regardant vers le ciel, vous voyez des formes ou des figures bien définies ; comme un lion qui rugit en colère ou encore un vieux nain barbu qui joue de la trompette ? Ces formes, appelées aussi patrons (patterns en Anglais), sont en effet le résultat de l'interprétation du cerveau humain selon un modèle qui lui est propre. De plus, elles sont vues comme un signe d'une grande imagination décrivant l'intelligence de l'être humain de trouver quelque chose significative dans un environnement non significatif, dit bruité. Cette capacité d'identifier les patterns existants, ou encore ceux émergents, est devenue un des sujets phares de l'intelligence artificielle (Artificial Intelligence - AI), qui permet le développement des machines, de plus en plus intelligentes, exécutant les fonctions cognitives humaines comme l'*apprentissage* à partir des événements passés et la *prédiction* de ce qui se passera dans le futur. Ces deux disciplines clés de l'IA permettent non seulement de supporter l'être humain dans ses processus décisionnels, mais aussi de lui donner la chance de naviguer dans le monde de la créativité et de l'innovation.

Dans ce chapitre, qui présente le cœur de notre contribution, nous concevons les deux modules, d'aide à la décision et de la gestion des changements, du système IDSS. Pour réaliser cet objectif, nous commençons d'abord par décrire ce qu'est l'aide à la décision dans notre contexte, puis nous faisons un tour d'horizon sur les techniques de l'intelligence artificielle dédiées au support de la prise de décision. Ensuite, après avoir choisi la ou les techniques qui répondent à nos besoins, nous présentons nos propositions relatives aux deux modules (celui de l'aide à la décision et celui de la gestion des changements) en les illustrant avec un exemple facilitant leur compréhension. Ces deux propositions sont, ensuite, discutées avant de conclure le présent chapitre.

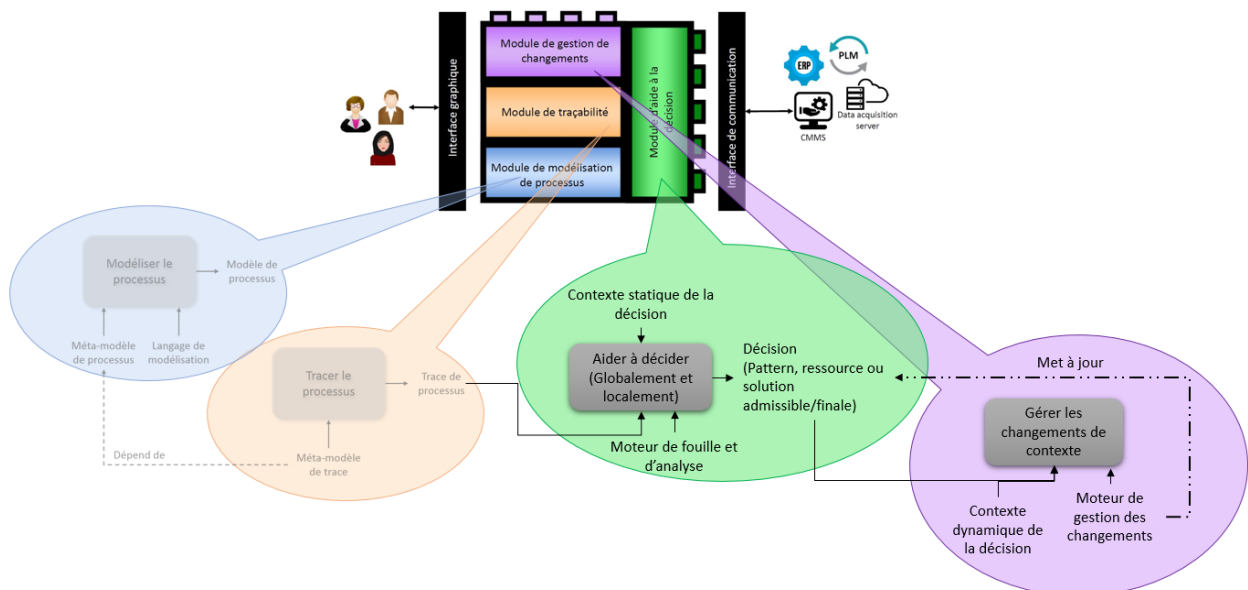


FIGURE 7.1 – Structure du Chapitre 7.

7.1 Introduction : vers des systèmes intelligents d'aide à la décision

La prise de décision à l'ère actuelle, comme cela a été longuement discuté dans les chapitres 3 et 4, est une tâche critique où une mauvaise décision peut engendrer des conséquences néfastes à court, moyen et/ou long terme. Pour pallier ce problème, et avoir une visibilité sur les conséquences des alternatives de décision entre autres, les systèmes d'aide à la décision (DSS - Decision Support Systems) ont été développés. Ces derniers se voient comme des applications informatiques permettant d'analyser les données et de les présenter de façon à ce que l'utilisateur puisse prendre ses décisions plus facilement et plus sûrement [Shim et al., 2002]. Le développement des DSSs a ouvert de nouvelles perspectives de recherche, dont les plus fameuses reposent sur l'utilisation des techniques de l'intelligence artificielle permettant de développer des systèmes démontrant une intelligence humaine, notamment la capacité d'apprendre. En effet, Phillips-Wren and Ichalkaranje [2008] voient que *"the combination of AI and DSSs provides formidable new computational assistants to humans that extend their capabilities in routine and complex stressful environments"*. Cette combinaison donne lieu aux systèmes intelligents d'aide à la décision¹⁰⁴ (IDSS - Intelligent Decision Support Systems) qui se voient plus adaptables, ayant une bonne capacité de support, et plus efficaces que les DSSs traditionnels [Phillips-Wren and Ichalkaranje, 2008]. Avant d'introduire le système que nous proposons, nous souhaitons définir, dans la section suivante, l'aide à la décision dans notre contexte.

7.2 Comment est définie l'aide à la décision dans notre contexte de recherche ?

Lors de la conception ou la supervision d'un système complexe, les ingénieurs/superviseurs se trouvent confrontés au besoin d'un support effectif lorsqu'il s'agit de prendre une décision. Cette dernière, qui a été introduite et discutée en détails dans le Chapitre 3, est devenue une des préoccupations de l'ère actuelle pour plusieurs raisons, dont quelques unes ont été présentées dans la Section 2.3.1 ; et sont brièvement rappelées ci-dessous :

- La prise des mauvaises décisions a un effet néfaste sur la performance des processus de conception et de supervision (Section 2.3.1),
- Le temps perdu lors de la prise de la prétendue bonne décision¹⁰⁵ engendre un retard significatif lors du développement d'un système ou lors de la résolution d'un problème donné (Section 2.3.1 et Section 2.3.1),
- Quelques unes des décisions prises lors de la conception ou la supervision des systèmes n'ajoutent pas de la valeur aux deux processus et peuvent, ainsi, être considérées comme une simple perte de temps (Section 2.3.1).

Une mauvaise décision peut, en effet, être vue comme l'utilisation d'une mauvaise ressource pour exécuter une activité du processus (prenons l'exemple de la Figure 7.2, l'utilisation de la ressource R_2 au lieu de la ressource R_1 peut donner lieu à un autre produit, qui malgré le fait qu'il soit légèrement modifié, ne sera pas accepté par le client qui souhaite avoir une voiture bleue, par exemple). La mauvaise décision peut aussi concerner le choix du processus en entier ; en effet, le choix d'un des processus P'_1 , P_2 , P_3 , P'_3 et P_4 est vu comme une mauvaise décision si le client exige un cabriolet bleu (Figure 7.2).

La mauvaise décision est aussi vue comme la prise d'une décision qui est bonne, certes, mais pas suffisamment performante pour atteindre les objectifs fixés. En effet, supposons maintenant que le client souhaite avoir une voiture 4x4 bleue marine (Produit 3 de la Figure 7.2), la décision d'utiliser la ressource

104. Plusieurs autres appellations existent dans la littérature, nous citons par exemple : les systèmes intelligents de décision (IDS - Intelligent Decision Systems), les systèmes d'aide à la décision basés sur les connaissances (KBDSS - Knowledge-Based Decision Support Systems), les systèmes d'aide à la décision actifs (Active DSSs) et les systèmes cognitifs conjoints (Join Cognitive Systems) [Phillips-Wren and Ichalkaranje, 2008].

105. L'adjectif "prétendue" est utilisé ici parce que les décisions venant d'être prises (après une éventuelle analyse décisionnelle) sont jugées être bonnes. Toutefois, l'évaluation objective n'aura lieu qu'après l'implémentation de ces décisions et l'étude de leur performance.

R_1 ou R_2 permet d'avoir le même produit final, cependant, R_2 se voit plus efficiente que R_1 puisqu'elle est moins chère et par conséquent le coût total du processus de conception P'_3 sera inférieur au coût total du processus P_3 . De la même façon, la décision de suivre le processus P_4 se voit plus performante parce que ce dernier, dont le coût est le même que le processus P'_3 , permet de concevoir la voiture souhaitée en temps record (deux ans au lieu de trois ans par exemple).

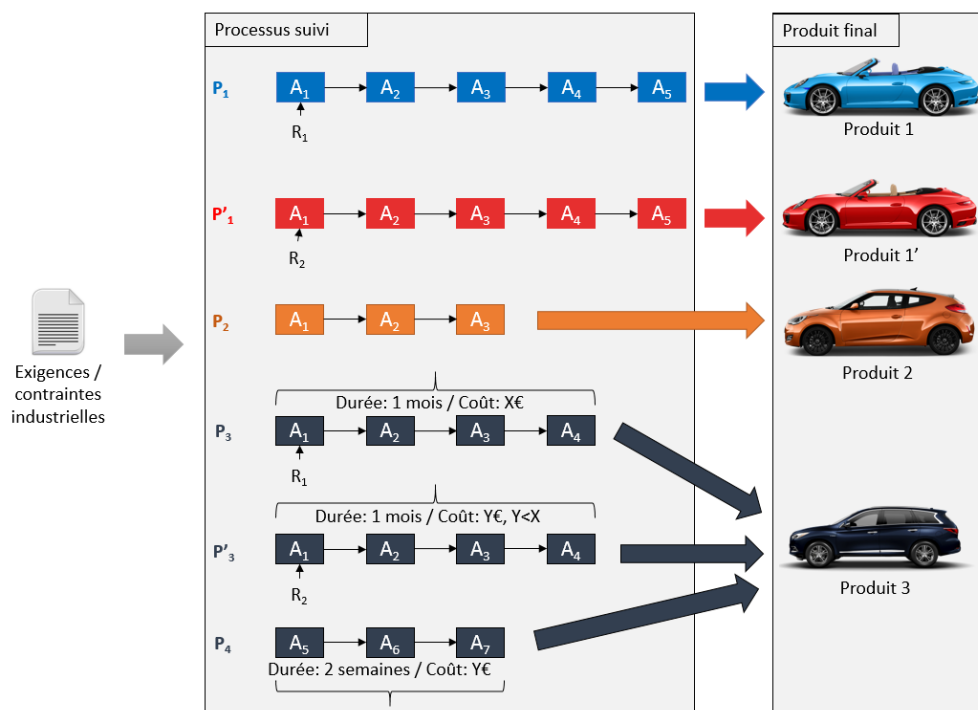


FIGURE 7.2 – Représentation IDEF0 des deux types de la prise de décision. Nous n'avons présenté que la ressource utilisée par la première activité de chaque processus pour ne pas charger la figure.

Deux niveaux de prise de décision dans notre contexte de recherche : De la discussion précédente, nous identifions deux niveaux de prise de décision comme le montre la Figure 7.3. Le niveau que nous appelons **global**, concerne le choix du pattern¹⁰⁶ de processus à exécuter (par exemple, le choix d'un des patterns P_1 à P_5 du processus de la Figure 7.3, définis dans le Tableau 7.1). Tandis que le niveau **local** (à l'activité), concerne le choix des ressources permettant d'exécuter chaque activité du pattern choisi. Par ailleurs, le niveau local concerne aussi le choix d'une ou plusieurs des solutions admissibles (resp. initiales) comme solutions finales (resp. admissibles) lorsqu'il s'agit d'une activité de décision (présentée dans la Figure 6.14 et discutée plus en détails dans la Section 6.1.5.1).

Pattern	Activités
P_1	A_1 A_3 A_5 A_6 A_4 A_9
P_2	A_1 A_3 A_4 A_5 A_6 A_9
P_3	A_1 A_3 A_5 A_4 A_6 A_9
P_4	A_2 A_7 A_8 A_9
P_5	A_2 A_8 A_7 A_9

TABLE 7.1 – Les patterns résultants des différentes exécutions possibles du processus de la Figure 7.3

Ainsi, pour aider l'utilisateur à prendre la bonne décision aux niveaux global et local identifiés, nous avons

106. Ou patron, c'est une exécution possible et complète du processus, de son début à sa fin. Ce concept sera discuté plus en détails dans les sections suivantes.

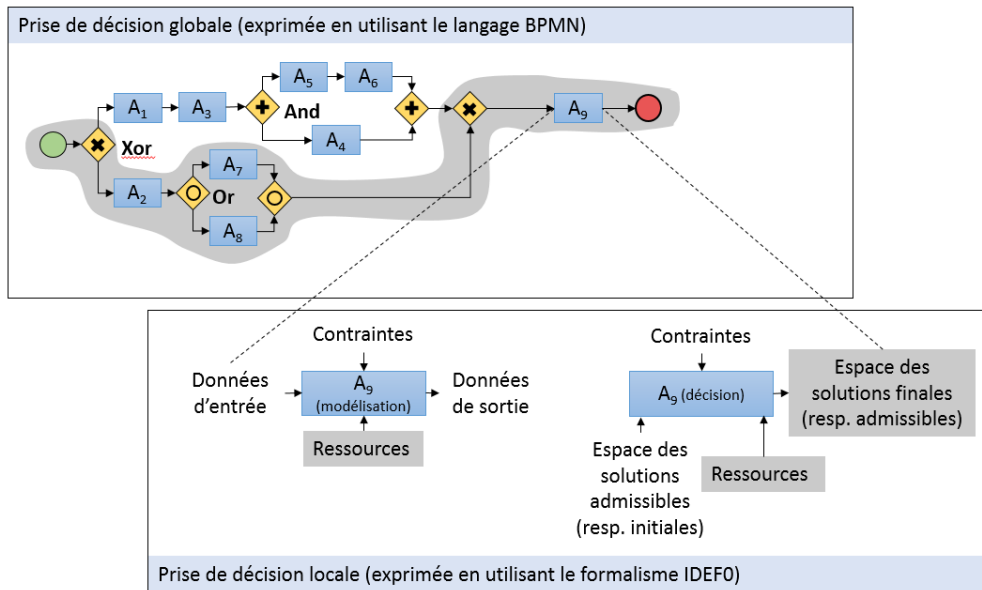


FIGURE 7.3 – Les deux niveaux de la prise de décision : local et global. Le premier est exprimé en IDEF0 pour souligner les paramètres de l'activité et le deuxième est exprimé en BPMN pour souligner la présence des jonctions.

besoin de trois éléments comme le montre la Figure 7.4 :

- Les **traces de processus** issues des exécutions passées, elles sont générées par le module de traçabilité et enregistrées dans une base de trace accessible par l'IDSS,
- Le **contexte de la décision** permettant de capturer toutes les informations décrivant la situation de la décision, comme le nom du décideur, la date de la décision, etc.
- Un **moteur de fouille et d'analyse** qui, conformément à notre hypothèse de travail présentée dans la Section 6.2, analyse les traces et propose la ou les décisions satisfaisant le contexte de décision.

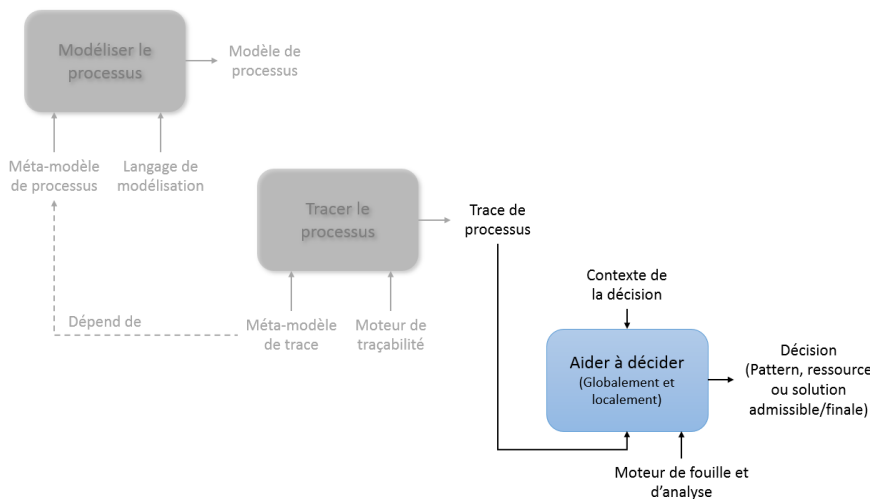


FIGURE 7.4 – Représentation IDEF0 de l'activité de prise de décision

Le premier élément du module de l'aide à la décision, étant le *contexte de la décision*, est acquis par le système IDSS via une interface informatique, permettant la communication et l'échange des données (comme décrit dans la Figure 2.10), dont le protocole n'est pas imposé par le système IDSS pour deux raisons principales. La première est que nous souhaitons concevoir un système intelligent d'aide à la décision *générique*, pouvant être utilisé dans n'importe quel environnement. La deuxième raison est qu'il existe plusieurs techniques d'échange de données dont le choix dépend des standards de communication

adoptés et parfois même de la préférence de l'utilisateur final qui peut, par exemple ¹⁰⁷, préférer d'utiliser le web service ¹⁰⁸ au lieu de l'échange des données via des sockets programmés ¹⁰⁹. Le deuxième élément du module, étant l'ensemble des traces de processus, est généré par le moteur du module de traçabilité dont l'implémentation sera discutée en détails dans le chapitre suivant. Pour choisir la ou les techniques à adopter pour concevoir le troisième élément, étant le moteur de fouille et d'analyse, nous souhaitons parcourir les travaux de recherche qui se basent sur l'Intelligence Artificielle (IA) et la Business Intelligence (BI) pour supporter la prise de décision globale et locale ; ceci fera l'objet de la sous-section suivante.

7.3 État de l'art sur quelques techniques de l'IA et de la BI supportant la prise de décision

7.3.1 Quelle est la différence entre l'intelligence artificielle, la fouille de données, l'apprentissage automatique, la fouille de processus et la business intelligence ?

Malgré le tas de définitions qu'elle a, l'intelligence artificielle est tout simplement une science visant le développement des systèmes informatiques capables d'exécuter des tâches nécessitant une intelligence humaine, telles que la vision par ordinateur et la reconnaissance de la parole entre autres. Dans la pratique, les tâches nécessitant de l'intelligence ont besoin de se doter de la capacité d'induire, de manière autonome, de nouvelles connaissances à partir des expériences ou des observations passées. Ainsi, une grande partie du domaine de l'intelligence artificielle est occupée par l'apprentissage automatique (Figure 7.5.a), qui permet de construire des systèmes ayant la capacité de tirer des leçons (ou apprendre) de leurs propres expériences pour améliorer leur performance. De plus, les techniques de l'apprentissage automatique et ceux des statistiques sont très utilisées, ou considérées comme base pour le développement de nouvelles techniques dédiées à la fouille de données ¹¹⁰, comme le montrent les figures 7.5.b et 7.5.c. Cette dernière traite un grand volume de données, appelé "Big Data", et a généralement pour but de découvrir des informations préliminaires dans un domaine où il y'a peu de connaissances préalables, et/ou de prédire avec précision les situations futures, appelées observations par la communauté de l'apprentissage automatique.

La notion de fouille de processus, étant aussi récente par rapport aux autres domaines discutés ci-dessus, a aussi gagné beaucoup d'intérêt ces dernières années pour son support précieux pour la prise de décision. La question la plus populaire qui se pose à ce stade là est : "La fouille de processus n'est-elle pas juste une nouvelle appellation, plus mystérieuse et accrochante, de la business intelligence ?" Pour répondre objectivement à cette question, nous devons d'abord définir la notion et le processus de la business intelligence. En effet, cette dernière se voit comme l'ensemble des techniques, outils et services qui analysent les données pour permettre la prise des décisions métier stratégiques et tactiques. Selon Vuori [2006], le processus de la business intelligence, présenté dans la Figure 7.6.a, consiste à :

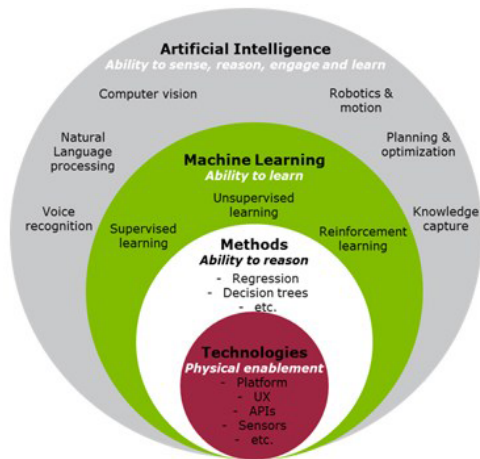
- **Identifier et collecter les informations** nécessaires pour une analyse donnée. Ces informations peuvent venir de différentes sources de données et doivent subir un éventuel pré-traitement pour être

107. Plusieurs standards et approches existent, nous avons cité deux pour justifier le choix de ne pas imposer un protocole de communication.

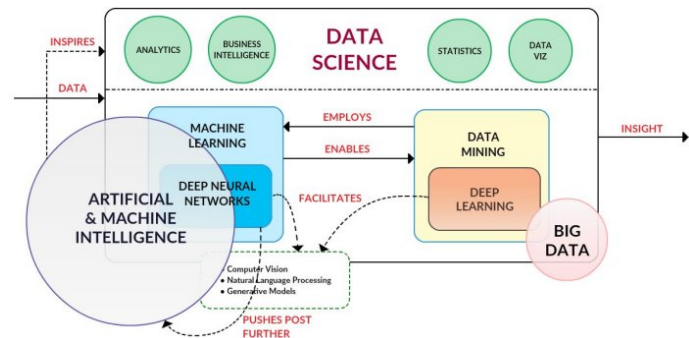
108. C'est un système logiciel permettant de supporter la communication machine-machine sur un réseau informatique. En d'autres termes, une machine peut appeler, sans intervention humaine, les services offerts par d'autres machines via internet et indépendamment des plates-formes et des langages utilisés [Ferris and Farrell, 2003].

109. Ils permettent d'établir un canal de communication bidirectionnel entre deux machines du réseau. Une des deux machines sera le serveur qui, après avoir accepté la demande de connexion, écoute la demande de l'autre machine appelée client et fournit la réponse appropriée [Stevens et al., 2004].

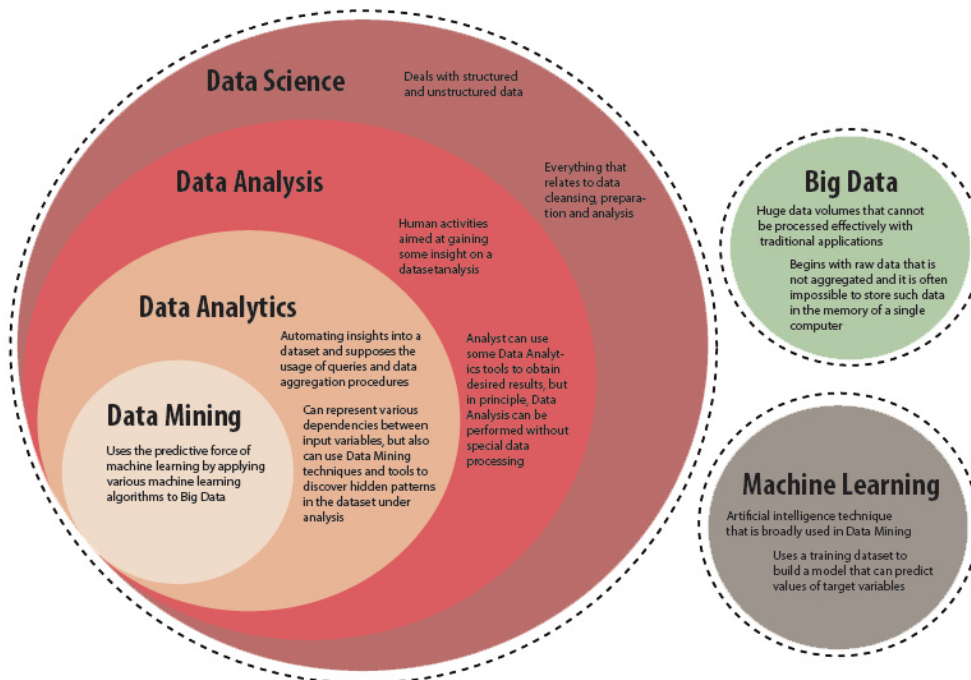
110. Deux grandes parties se distinguent quand il s'agit de comparer la fouille de données et l'apprentissage automatique. La première voit que ce dernier est une des techniques utilisées pour fouiller dans les données, tandis que la deuxième équipe assure que l'apprentissage automatique utilise les techniques de la fouille de données pour construire un modèle permettant la prédiction des situations futures. Nous partageons l'avis de la première équipe puisque la fouille de données, ayant été introduite en 1930 sous le nom de "Knowledge discovery", se voit plus ancienne que l'apprentissage automatique ayant été introduit en 1950 [Import.io, 2017]. De ce fait, nous considérons que l'apprentissage automatique est une des techniques utilisées par la fouille de données, comme décrit dans la Figure 7.5.b.



7.5.a. L'utilisation des techniques de l'apprentissage automatique pour développer les capacités d'un système artificiellement intelligent [Van Duin and Bakhshi, 2017]



7.5.b. L'utilisation des techniques de l'apprentissage automatique pour explorer les données [Mayo, 2017]



7.5.c. Différence entre les domaines de fouille de données et de l'apprentissage automatique selon <https://iotechnologies.com>

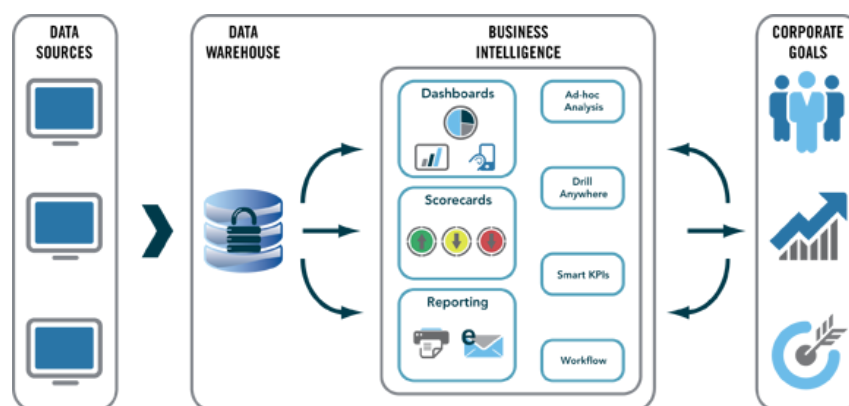
FIGURE 7.5 – Combinaison des forces de l'apprentissage automatique et de la fouille de données pour construire des systèmes artificiellement intelligents.

nettoyées et consolidées en vue de leur exploitation,

- **Analyser les informations collectées** pour découvrir les patterns communs et mesurer et/ou mettre à jour les indicateurs de performances entre autres,
- **Représenter les résultats de l'analyse** d'une façon intuitive (sous forme de tableaux ou autres graphiques) pour permettre à l'utilisateur de bien les visualiser et correctement les interpréter.

L'étape de l'analyse des informations, du processus de la Business Intelligence, fait appel à plusieurs techniques comme la fouille de données et de processus entre autres. Ceci nous pousse à poser une autre question : "Quelle est la différence entre ces deux concepts ?" Il est vrai que les deux ont la notion de "fouille" en commun, mais la finalité n'est pas vraiment la même. En effet, la fouille de données -comme

son nom l'indique- consiste à explorer les données pour découvrir les patterns les plus significatifs. La fouille de processus, en revanche, consiste à explorer les processus en entier -qui se voient comme un ensemble fini d'activités manipulant des données (données d'entrée, de sortie, contraintes et ressources si nous considérons la notation IDEF0)- en combinant les techniques de la fouille de données et de la modélisation des processus, comme le montre la Figure 7.6.b. De plus, la fouille de processus permet non seulement d'explorer les processus en générant automatiquement les modèles de processus qui sont souvent manuellement modélisés, mais aussi de vérifier leur conformité par rapport aux processus réellement exécutés, et de les améliorer si nécessaire.



7.6.a. Processus de la Business Intelligence selon takefiveconsulting.com



7.6.b. Combinaison des techniques de la fouille de données et de la modélisation des processus pour des fins de fouille de processus (fluxicon.com)

FIGURE 7.6 – Différence entre la fouille de processus et la Business Intelligence.

Il est maintenant clair que les systèmes d'aide à la décision intelligents font, principalement, appel aux techniques des domaines de l'apprentissage automatique, de la fouille de données et de la fouille de processus. Par conséquent, nous souhaitons par la suite explorer les techniques de base de ces trois domaines, notamment :

- Les règles d'association et la fouille de séquences du domaine de la fouille de données,
- L'apprentissage supervisé, non supervisé et semi-supervisé et par renforcement du domaine de l'apprentissage automatique,
- L'exploration des processus, le contrôle de conformité et l'amélioration du domaine de la fouille des processus.

Après avoir introduit chacune de ces techniques, nous discutons la possibilité de leur application à l'aide à la décision comme nous l'avons déjà définie dans la Section 7.2.

7.3.2 Techniques de fouille de données (Data mining)

7.3.2.1 Fouille des règles d'association (Association rule mining)

Le but de cette section est de définir la logique de la fouille des règles d'association afin de discuter, dans la section suivante, la possibilité de son utilisation dans notre contexte. Plusieurs travaux de recherche listent les approches et comparent les algorithmes permettant de fouiller les règles d'association, le lecteur peut se

référer aux travaux de [Zhao and Bhowmick \[2003a\]](#) et [Hipp et al. \[2000\]](#) par exemple. [Agrawal et al. \[1993\]](#) définissent le problème de la fouille des règles d'association comme suit :

Définition 7.1. Soient $\{I = i_1, i_2, \dots, i_m\}$ un ensemble de m attributs distincts, T une transaction contenant un ensemble d'éléments tel que $T \subseteq I$ et D une base de données contenant des records de transaction différents. Une règle d'association est une implication de la forme $X \Rightarrow Y$ où $X, Y \subset I$ sont des ensembles d'éléments appelés respectivement l'antécédent et le conséquent, et $X \cup Y = \emptyset$.

Les règles d'association ont deux paramètres intéressants [[Zhao and Bhowmick, 2003b](#)] :

- L'indice de **support**, qui représente la fréquence d'apparition de la règle d'association dans la base de données :

$$s(XY) = \frac{\sigma(X \cup Y)}{\text{nombre total de transactions dans } D} \quad (7.1)$$

où σ calcule la fréquence d'apparition d'un ensemble d'éléments,

- L'indice de **confiance**, qui mesure la force (pertinence) de la règle d'association, en calculant le nombre de fois où les éléments de Y apparaissent dans une transaction contenant X :

$$c(XY) = \frac{\sigma(X \cup Y)}{\sigma(X)} \quad (7.2)$$

Vu que la base de données est souvent large et que l'utilisateur ne s'intéresse qu'aux éléments fréquents, deux autres paramètres seuils sont définis. Il s'agit du *support minimal* et de la *confiance minimale*, relatifs respectivement aux indices de support et de confiance, en dessous desquels les règles sont supposées non intéressantes et sont éliminées.

Comme nous voyons que la meilleure méthode pour expliquer un problème donné est de le projeter sur un exemple, nous considérons le Tableau 7.2, étant une base de données qui présente les achats d'un client "W", où $D = \{T_1, T_2, T_3, T_4, T_5\}$ et $I = \{\text{pain, bonbons, fromage, lait, coca - cola}\}$. La règle d'association " $\text{pain} \Rightarrow \text{fromage}$ " se voit plus intéressante que la règle " $\text{coca - cola} \Rightarrow \text{pain}$ " car $s(\text{pain} \Rightarrow \text{fromage}) > s(\text{coca - cola} \Rightarrow \text{pain})$ et $c(\text{pain} \Rightarrow \text{fromage}) > c(\text{coca - cola} \Rightarrow \text{pain})$ comme le montre le Tableau 7.3. De même, la règle " $\text{fromage} \Rightarrow \text{pain}$ " est plus intéressante que la règle " $\text{pain} \Rightarrow \text{fromage}$ ", car même si $s(\text{fromage} \Rightarrow \text{pain}) = s(\text{pain} \Rightarrow \text{fromage})$, $c(\text{fromage} \Rightarrow \text{pain}) > c(\text{pain} \Rightarrow \text{fromage})$. Ceci est logique parce que l'achat du fromage est accompagné avec l'achat du pain aussi (transactions T_1, T_2 et T_3 du Tableau 7.2), alors que l'achat du pain peut ne pas être accompagné par l'achat du fromage (transaction T_4 du Tableau 7.2).

Référence de la transaction	Éléments
T_1	pain, bonbons, fromage
T_2	pain, fromage
T_3	pain, lait, fromage
T_4	coca-cola, pain
T_5	coca-cola, lait

TABLE 7.2 – Exemple des achats d'un client.

7.3.2.2 Fouille des patterns séquentiels (Sequential pattern mining)

Loin du pain et du fromage, la fouille de données permet également de découvrir les régularités dans les données se présentant sous forme de séquences, dont chacune se voit comme une liste ordonnée d'événements (Figure 7.7).

La fouille des patterns séquentiels est formellement définie par [Agrawal and Srikant \[1995\]](#) comme suit :

Règle d'association	Indice de support (s)	Indice de confiance (c)
$\text{pain} \Rightarrow \text{fromage}$	0.60	0.75
$\text{fromage} \Rightarrow \text{pain}$	0.60	1.00
$\text{coca} - \text{cola} \Rightarrow \text{pain}$	0.20	0.50
$\text{fromage} \Rightarrow \text{bonbons}$	0.20	0.33
$\text{bonbons} \Rightarrow \text{fromage}$	0.20	1.00
$\text{bonbons} \Rightarrow \text{lait}$	0.00	0.00

TABLE 7.3 – Indices de support et de confiance pour les différentes règles d'association de la base de données du Tableau 7.2.



FIGURE 7.7 – Exemple d'une séquence d'événements. Noter que $\langle (a) (b) (d) (e\ g) \rangle$ est une sous-séquence de $\langle (a) (b\ c) (d) (e\ f\ g) \rangle$.

Définition 7.2. Soient $D = \{T_1, T_2, \dots, T_n\}$ la base de données des transactions d'un client, $\text{itemset}(T_i)$ l'ensemble des éléments contenus dans la transaction T_i et $\langle \text{itemset}(T_1) \text{itemset}(T_2) \dots \text{itemset}(T_n) \rangle$ la séquence client. Sachant qu'une séquence $\langle a_1 a_2 \dots a_n \rangle$ est contenue dans une autre séquence $\langle b_1 b_2 \dots b_n \rangle$ s'il existe des entiers $\langle i_1 < i_2 < \dots < i_n$ tel que $a_1 \subseteq b_{i_1}, a_2 \subseteq b_{i_2}, \dots, a_n \subseteq b_{i_n}$. Et sachant que dans un ensemble de séquences, une séquence s est dite *maximale* si elle n'est contenue dans aucune des autres séquences. La fouille des patterns séquentiels consiste à trouver les séquences *maximales* parmi toutes les séquences dont l'indice de support est supérieur ou égal à l'indice de support minimal (étant le seuil pré-défini).

Pour expliciter cette définition, nous considérons le Tableau 7.4 qui contient cinq séquences. Pour un support minimum (comme décrit dans la Section 7.3.2.1) de 25% par exemple, nous identifions deux patterns séquentiels : $\langle (c) (i) \rangle$ et $\langle (c) (d\ g) \rangle$. En effet, le premier pattern $\langle (c) (i) \rangle$ est supporté par les séquences S_1 et S_4 car il est présent dans ces deux séquences¹¹¹ parmi les cinq du Tableau 7.4, et donc a un indice de support égal à 0.40 (en appliquant l'équation 7.1) soit de 40%. De même, le deuxième pattern $\langle (c) (d\ g) \rangle$ est supporté par les séquences S_2 et S_4 , et a donc un indice de support de 40%. La séquence $\langle (a\ b) (c) \rangle$ n'est, en revanche, pas vue comme pattern séquentiel parce que son indice de support de 20% est inférieur à 25% étant le seuil considéré. De plus, la séquence $\langle (c) \rangle$, qui malgré le fait qu'elle ait un indice de support supérieur à 25%, n'est pas considérée comme pattern séquentiel parce qu'elle n'est pas maximale, car elle est une sous-séquence (comme expliqué dans la Figure 7.7) de la séquence $\langle (c) (i) \rangle$ entre autres.

Référence	Séquence
S_1	$\langle (c) (i) \rangle$
S_2	$\langle (a\ b) (c) (d\ f\ g) \rangle$
S_3	$\langle (c\ e\ g) \rangle$
S_4	$\langle (c) (d\ g) (i) \rangle$
S_5	$\langle (i) \rangle$

TABLE 7.4 – Exemple de base de données séquentielles contenant cinq séquences, adapté de [Agrawal and Srikant, 1995].

Plusieurs techniques sont développées pour permettre la fouille des patterns séquentiels, et sont présentées

¹¹¹. Noter que l'existence de l'élément $\langle (d\ g) \rangle$ dans la séquence S_4 du Tableau 7.4 ne pose pas de problème si nous cherchons des patterns qui ne sont pas nécessairement contigus.

et comparées dans les travaux de [Zhao and Bhowmick \[2003b\]](#), [Han et al. \[2007\]](#) et [Mabroukeh and Ezeife \[2010\]](#) entre plusieurs autres. De plus, il existe plusieurs autres techniques permettant la fouille et l'exploration des données, venant de plusieurs autres domaines, notamment les statistiques, l'intelligence artificielle, l'apprentissage automatique et les systèmes de base de données. Le lecteur pourrait satisfaire sa curiosité en jetant un coup d'œil sur les travaux de [James et al. \[2013\]](#) et [Kudyba \[2014\]](#), entre autres.

7.3.3 Techniques de l'apprentissage automatique (Machine learning)

Comme décrit dans la Section 7.3.1, l'apprentissage automatique vise à construire des programmes informatiques "able to construct new knowledge or to improve already possessed knowledge by using input information" [[Kodratoff and Michalski, 2014](#)]. Le point de départ de l'apprentissage automatique est un ensemble d'enregistrement de données (également appelés jeux de données, instances, observations ou cas) ayant une des trois formes présentées dans les tableaux 7.5, 7.6 et 7.7. Une instance est décrite par un vecteur d'attributs ou de caractéristiques (*features* en Anglais) à n dimensions : $X = (X_1, X_2, \dots, X_n)$ et possède, éventuellement¹¹², un attribut cible Y appelé *classe* ou *label*. La plupart des problèmes d'apprentissage tombent dans une des quatre catégories suivantes : l'apprentissage supervisé, l'apprentissage non-supervisé, l'apprentissage semi-supervisé ou l'apprentissage par renforcement. Chacune de ces quatre catégories est brièvement présentée ci-après.

Instance	X (vecteur d'entrée ou de caractéristiques)			Y (classe)
	X_1	X_2	X_i	
1	X_{11}	X_{21}	X_{n1}	Y_1
2	X_{12}	X_{22}	X_{n2}	Y_2
3	X_{13}	X_{23}	X_{n3}	Y_3
m	X_{1m}	X_{2m}	X_{nm}	Y_m

TABLE 7.5 – La forme d'un jeu de données où deux paramètres sont connus : X étant le vecteur de caractéristiques et Y étant la classe à laquelle appartient X .

Instance	X (vecteur d'entrée ou de caractéristiques)		
	X_1	X_2	X_i
1	X_{11}	X_{21}	X_{n1}
2	X_{12}	X_{22}	X_{n2}
3	X_{13}	X_{23}	X_{n3}
m	X_{1m}	X_{2m}	X_{nm}

TABLE 7.6 – La forme d'un jeu de données où seul X est connu.

Instance	X (vecteur d'entrée ou de caractéristiques)			Y (classe)
	X_1	X_2	X_i	
1	X_{11}	X_{21}	X_{n1}	-
2	X_{12}	X_{22}	X_{n2}	Y_2
3	X_{13}	X_{23}	X_{n3}	-
m	X_{1m}	X_{2m}	X_{nm}	Y_m

TABLE 7.7 – La forme d'un jeu de données où X et quelques Y sont connus.

112. Car l'existence du label dépend de la nature du jeu de données et du type d'apprentissage. En effet, le label existe lorsqu'il s'agit de l'apprentissage supervisé ou semi-supervisé, il est en revanche non présent lorsqu'il s'agit de l'apprentissage non-supervisé.

7.3.3.1 Apprentissage supervisé (Supervised learning)

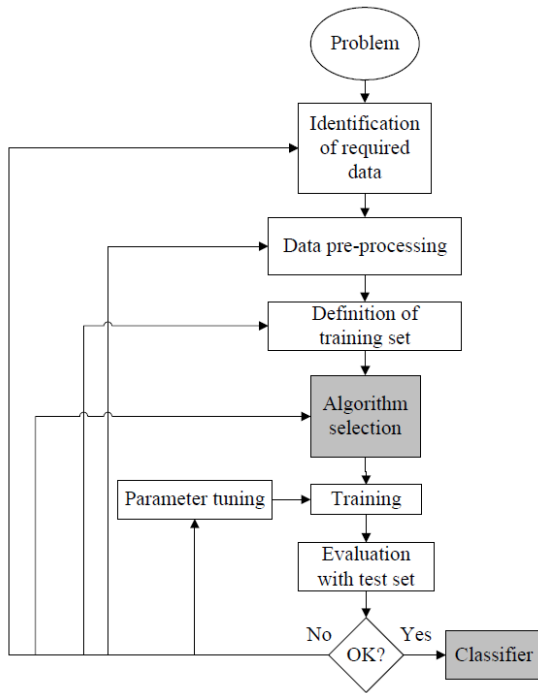
Ce type d'apprentissage est dit supervisé parce que le label Y , du jeu de données dont la structure générale est présentée dans le Tableau 7.5, est connu. Le processus de l'apprentissage supervisé passe par plusieurs étapes comme le montre la Figure 7.8.a :

- La première étape consiste à définir le problème d'apprentissage (prédire si un e-mail est considéré comme spam, par exemple) et le jeu de données approprié (l'ensemble des e-mails précédents, leurs caractéristiques (date, expéditeur, etc.) et leur nature (spam ou non spam), par exemple).
- Ayant un jeu de données bien défini, il est possible de le pré-traiter pour enlever les instances n'apportant aucun intérêt et considérées comme bruit, les techniques permettant de détecter ce bruit sont présentées et discutées par [Hodge and Austin \[2004\]](#), entre autres. Le but du pré-traitement est aussi d'optimiser l'apprentissage en enlevant les caractéristiques non intéressantes ou redondantes (l'adresse postale, par exemple, ne se voit pas comme caractéristique intéressante pour prédire si un e-mail est considéré comme spam).
- Ayant un jeu de données pré-traité, l'utilisateur en sélectionne la partie, appelée *training set*, qui sera utilisée pour entraîner l'algorithme d'apprentissage supervisé considéré (en laissant le label Y connu), et celle qui aura pour objectif de tester cet algorithme (*test set* où le label Y est masqué en remplaçant sa valeur par un point d'interrogation "?"). En effet, deux phases principales sont présentes dans ce type d'apprentissage comme le montre la Figure 7.8.b. D'abord la phase d'entraînement où l'algorithme sélectionné est entraîné par le jeu de données d'entraînement. Cet algorithme est ensuite testé avant d'être définitivement pris en compte, le test est réalisé en utilisant l'algorithme pour prédire le label -déjà connu mais masqué- du jeu de données de test (Figure 7.8.b).
- L'évaluation de l'algorithme est souvent basée sur l'exactitude de prédiction, vue comme le pourcentage de prédictions correctes divisées par le nombre total de prédictions. Si le taux d'erreur de prédiction n'est pas satisfaisant, l'utilisateur doit examiner une variété de facteurs : vois si les caractéristiques pertinentes du problème sont pas prises en compte, considérer un jeu de données d'entraînement plus important, réduire la dimensionnalité (le nombre de variables considérées) du problème, sélectionner un autre algorithme d'apprentissage plus approprié ou encore régler les paramètres d'apprentissage (Figure 7.8.a).
- Si l'évaluation de l'algorithme est en revanche satisfaisante, ce dernier sera ensuite validé et utilisé par la phase de prédiction pour prédire le label d'un ensemble de données réelles [[Han et al., 2011](#)] (Figure 7.8.b)

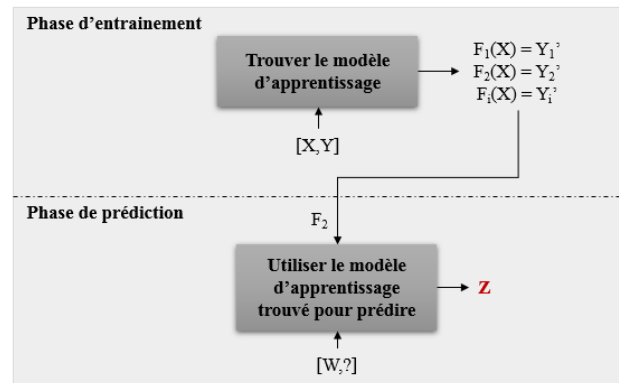
L'apprentissage supervisé comporte deux tâches principales : la classification et la régression. La première tâche concerne les jeux de données où le label Y est un ensemble discret et fini comme vrai/faux, oui/non, homme/femme, etc. La classification a pour objectif de prédire la classe à laquelle appartient chaque instance d'un ensemble de données réel, comme le montre la Figure 7.9.a. La régression, quant à elle, concerne les jeux de données où le label Y est un nombre continu. Elle a pour objectif de trouver la fonction permettant d'estimer le label sachant la caractéristique, la Figure 7.9.b présente un exemple de régression où la fonction, liant le prix de la voiture et la caractéristique x , est linéaire.

7.3.3.2 Apprentissage non-supervisé (Unsupervised learning)

L'apprentissage non-supervisé se voit le plus proche de ce qu'on appelle l'intelligence "artificielle" parce que l'ordinateur, ici, a la capacité d'apprendre à identifier les patterns complexes, présents dans un jeu de données, sans avoir besoin d'une aide humaine pour le guider en indiquant le label Y associé à chaque vecteur d'entrée X . En d'autres mots, ce type d'apprentissage, dont la tâche la plus populaire est le partitionnement (clustering), concerne les jeux de données -dits non labellisés- où le label Y n'est pas connu ou n'existe pas comme le montre le Tableau 7.6. Son objectif est d'identifier les catégories ou groupes significatifs appelés "clusters" [[Albalade et Minker, 2013](#)], de façon à ce que les données appartenant au même cluster aient des propriétés similaires et les données appartenant à des clusters différents aient des propriétés très différentes (Figure 7.10.a).

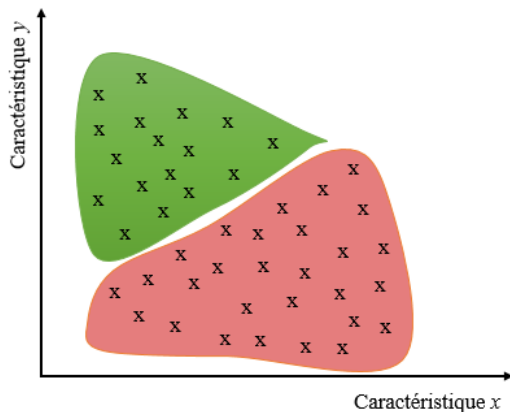


7.8.a. Processus général de l'apprentissage supervisé selon Kotsiantis [2007]

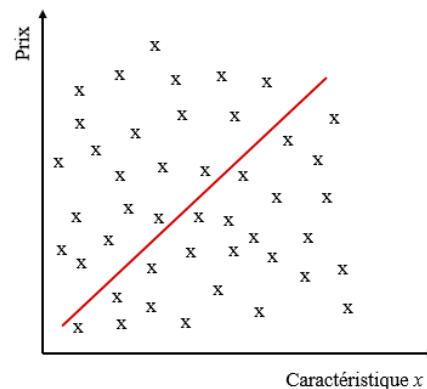


7.8.b. Représentation IDEF0 des deux phases de l'apprentissage supervisé

FIGURE 7.8 – Processus et phases de l'apprentissage supervisé.



7.9.a. Exemple de la classification des e-mails selon certaines caractéristiques x et y . La classe en vert regroupe les e-mails non-spam et la classe en rouge regroupe les e-mails spam



7.9.b. Exemple de l'estimation du prix d'une voiture selon une caractéristique x

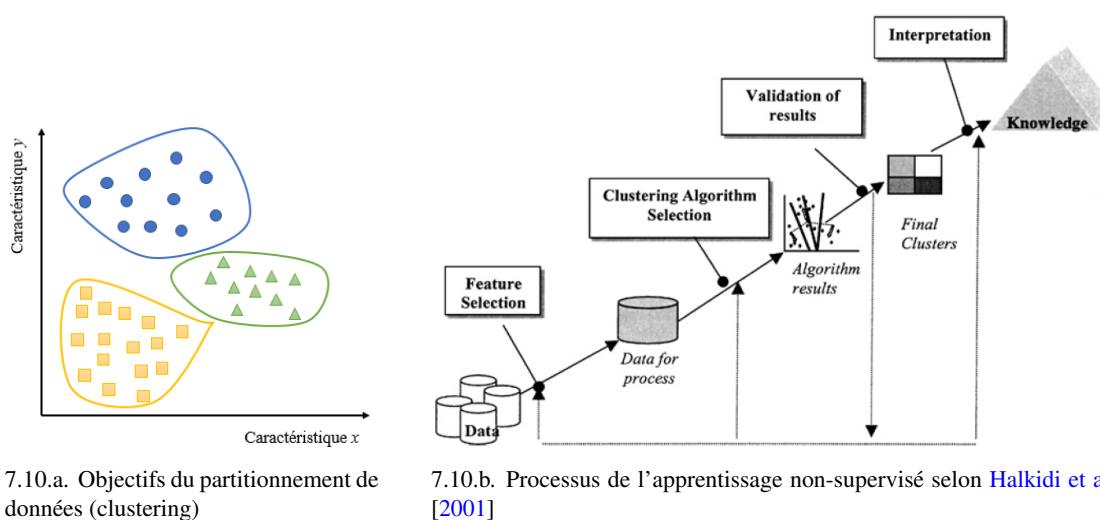
FIGURE 7.9 – Processus et phases de l'apprentissage supervisé.

L'apprentissage non-supervisé suit le processus de la Figure 7.10.b :

- Sélectionner les caractéristiques qui se voient les plus intéressantes pour la tâche d'apprentissage non-supervisé en question. Ceci permet d'efficacement trouver les groupes existants, de comprendre mieux les données et de réduire leur taille, entre autres [Dash and Koot, 2009],
- Sélectionner l'algorithme de clustering le plus adapté au jeu de données et à la tâche de partitionnement en question. Deux caractéristiques principales permettent de comparer l'efficacité des algorithmes de partitionnement : la mesure de proximité et le critère de partitionnement [Halkidi et al., 2001]. La première caractéristique permet de quantifier la similitude entre deux points de données (deux vecteurs de caractéristiques X_1 et X_2 par exemple) et la deuxième caractéristique définit le

critère de partitionnement comme le type de clusters attendus et leur nombre, entre autres [Halkidi et al., 2001].

- Valider les résultats obtenus. En effet, n'étant pas guidé pendant la phase de partitionnement et ayant des clusters qui ne sont pas connus a priori, l'utilisateur doit vérifier l'exactitude des résultats obtenus à l'aide des techniques appropriées [Halkidi et al., 2001],
- Interpréter les résultats du partitionnement en les associant à d'autres analyses et expériences afin de tirer la conclusion la plus juste possible.



7.10.a. Objectifs du partitionnement de données (clustering)

7.10.b. Processus de l'apprentissage non-supervisé selon Halkidi et al. [2001]

FIGURE 7.10 – Processus et résultat de l'apprentissage non-supervisé.

L'apprentissage non-supervisé se voit souvent comme une des premières étapes de la classification supervisée [Jain et al., 1999]. En effet, ayant un large jeu de données non labellisées, les analystes utilisent les techniques de clustering pour identifier les groupes existants, puis associent les données aux patterns appropriés. Le lecteur peut consulter les travaux de Jain et al. [1999], Albalade and Minker [2013] et Grira et al. [2004], entre autres, pour avoir une idée sur les approches et techniques considérées dans ce type d'apprentissage.

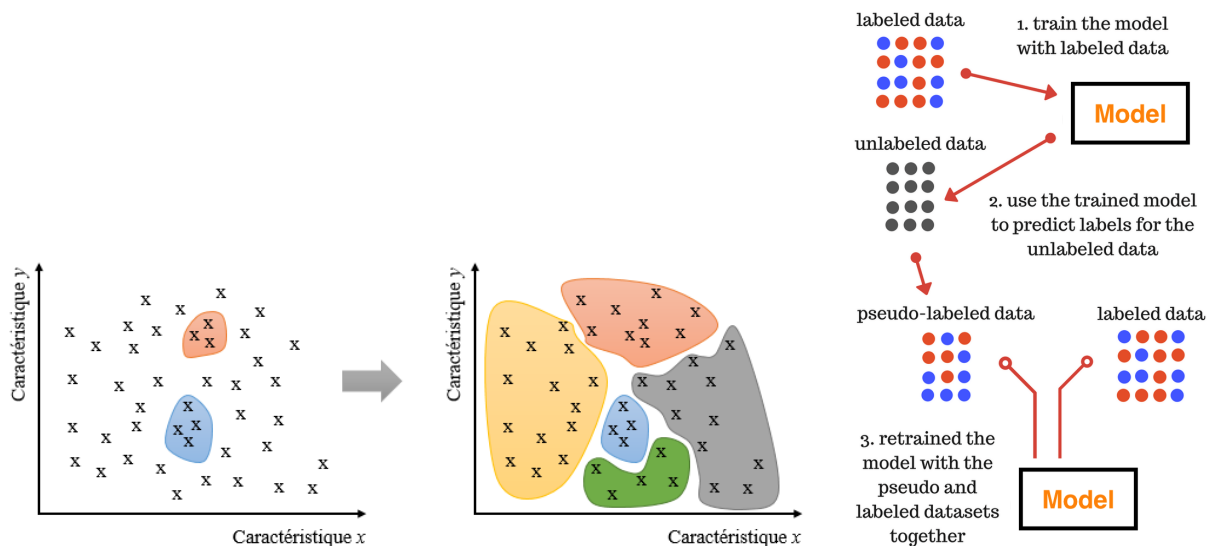
7.3.3.3 Apprentissage semi-supervisé (semi-supervised learning)

L'apprentissage supervisé, comme a été discuté dans la Section 7.3.3.1, nécessite un large volume de données labellisées pour bien entraîner l'algorithme afin d'augmenter l'efficacité et la qualité de la prédiction. La préparation d'un tel jeu de données, qui comprend l'acquisition des données et leur étiquetage, est un processus difficile, long et coûteux [Zhu, 2006]. De plus, un étiquetage excessif pourrait introduire un biais humain au modèle d'apprentissage [Castle, 2018]. Pour pallier ces problèmes, a été introduit l'apprentissage semi-supervisé qui considère un jeu de données composé de quelques données labellisées et de nombreuses données non labellisées (Tableau 7.7). Ces dernières peuvent améliorer considérablement la précision du modèle d'apprentissage tout en réduisant le temps et les coûts associés à sa construction [Zhu, 2006; Chapelle et al., 2010].

L'apprentissage semi-supervisé comporte deux tâches comme les montrent les figures 7.11 et 7.12. La première tâche concerne la classification semi-supervisée où les classes, relatives aux données labellisées, sont préalablement connues (Figure 7.11.a) et dont la manière comment les données non labellisées sont exploitées, dépend de la méthode adoptée. Nous citons par exemple la méthode de la pseudo-labellisation, discutée plus en détail dans le travail de Lee [2013], qui consiste à utiliser les données labellisées pour entraîner le modèle d'apprentissage, qui est ensuite utilisé pour classifier les données non labellisées donnant lieu, ainsi, à un ensemble de données dites pseudo-labellisées. Ces dernières ainsi que les données labellisées sont utilisées ensemble pour ré-entraîner le modèle d'apprentissage (Figure 7.11.b).

La deuxième tâche de l'apprentissage semi-supervisé concerne le clustering semi-supervisé, appelé aussi clustering sous contraintes. Cette tâche suit le même processus du clustering supervisé discuté dans la Section 7.3.3.2, et a la particularité d'avoir en plus de la mesure de proximité, des informations à priori comme les contraintes "must-link" qui indiquent que deux points de données doivent appartenir au même groupe et les contraintes "cannot-link" indiquant que les deux points doivent appartenir à deux groupes différents (Figure 7.12).

Les différentes techniques de la classification et le clustering semi-supervisés sont présentées et discutées dans les travaux de Krishnapuram et al. [2005], Zhu [2006] et Grira et al. [2004] entre plusieurs autres.



7.11.a. Objectif de la classification semi-supervisée avec deux classes connues à priori (orange et bleu)

7.11.b. Processus de la pseudo-étiquetage, une méthode de la classification semi-supervisée [SHUBHAM, 2017]

FIGURE 7.11 – Objectif et exemple de processus de la classification semi-supervisée.

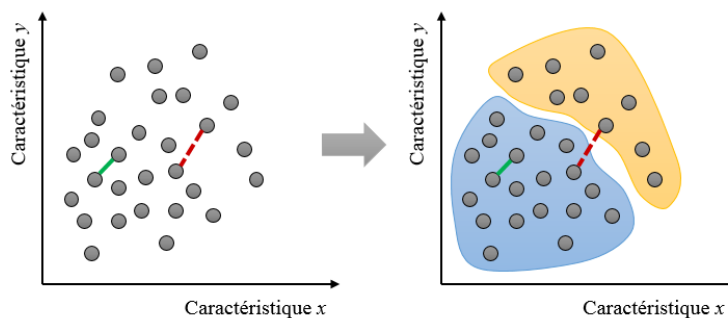


FIGURE 7.12 – Le partitionnement semi-supervisé avec les contraintes *must-link* (en vert continu) et *cannot-link* (en rouge pointillé).

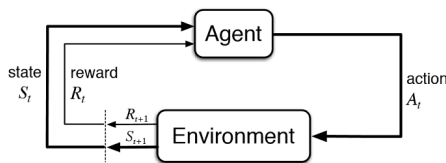
7.3.3.4 Apprentissage par renforcement (Reinforcement learning)

Les apprentissages supervisé, non-supervisé et semi-supervisé permettent de résoudre un nombre important de problèmes intéressants, allant de la classification des images, à l'estimation du prix d'un produit, à la détection des anomalies au marketing ciblé. Ces techniques se trouvent en revanche limitées, voire non appropriées, lorsqu'il s'agit des problèmes d'interaction avec l'environnement extérieur comme jouer au

parcheesi¹¹³ ou encore apprendre à un robot comment adapter sa façon de marcher en fonction de l'état du terrain. En effet, prenons le deuxième problème comme exemple et soit le jeu de données où X est l'état du terrain et Y est l'ensemble des actions optimales pouvant être exécutées, au niveau de l'état X :

- Les jeux de données de ce type n'existent pas pour tous les environnements (terrains) possibles, et sont coûteux voire parfois impossibles à construire si aucune information sur les actions optimales n'est connue,
- Il n'est pas possible d'améliorer les actions futures du robot car les effets de ses actions ne sont pas examinées (si à un état donné le robot exécute une action, aucun signal de récompense positif (note positive indiquant que l'action est bonne) ou négatif n'est reçu),
- En considérant le jeu de données construit, le robot apprendra à imiter un expert humain sans apprendre à chercher la meilleure stratégie pour résoudre un problème donné tenant compte de l'environnement.

L'apprentissage par renforcement permet de surmonter ces difficultés en offrant, à une machine, la possibilité d'identifier les meilleures actions par "essai-erreur" compte tenu d'une situation/d'un environnement donné de façon à ce que le signal de récompense soit maximisé. En effet, les algorithmes de l'apprentissage par renforcement, contrairement aux autres types d'apprentissage, ne sont pas informés des actions à prendre et n'ont ni des données d'entraînement ni des labels, ils doivent ainsi découvrir les actions engendrant une récompense maximale en les essayant.



7.13.a. Processus de l'apprentissage par renforcement selon Sutton et al. [1998]

environment	You are in state 65. You have 4 possible actions.
agent	I'll take action 2.
environment	You have received a reward of 7 units. You are now in state 15. You have 2 possible actions.
agent	I'll take action 1.
environment	You have received a reward of -4 units. You are now in state 65. You have 4 possible actions.
agent	I'll take action 2.
environment	You have received a reward of 5 units. You are now in state 44. You have 5 possible actions.
...	...

7.13.b. Exemple de l'interaction d'un agent avec son environnement [Kaelbling et al., 1996]

FIGURE 7.13 – Processus et exemple de l'apprentissage par renforcement.

Selon Kaelbling et al. [1996], un modèle d'apprentissage par renforcement se compose généralement d'un :

- Ensemble d'états possibles S , avec s_0 comme état initial,
- Ensemble discret d'actions possibles A ,
- Ensemble de signaux de renforcement R (comme $\{0,1\}$ par exemple),

Le scénario typique de ce type d'apprentissage se voit comme une boucle dont les étapes sont les suivantes (Figure 7.13.b) :

- L'agent reçoit l'état initial s_0 de son environnement,
- L'agent sélectionne l'action A_0 qui, selon son expérience maximise la récompense ou qu'il veut essayer si aucune information sur la récompense n'est connue,
- L'environnement passe au nouvel état $s_{suivant}$ et envoie un signal de récompense r à l'agent,
- L'agent reçoit la récompense r et $s_{suivant}$ et recommence la boucle dont l'exécution continue permet d'améliorer le comportement de l'agent.

Plusieurs techniques ont été développées dans le cadre de l'apprentissage par renforcement, elles sont discutées et comparées dans plusieurs travaux. Nous citons par exemple ceux de Kaelbling et al. [1996], Kulkarni [2012] et Wiering and van Otterlo [2012], entre autres.

113. C'est un jeu de société d'origine indienne : <http://www.ludoteka.com/parcheesi.html>

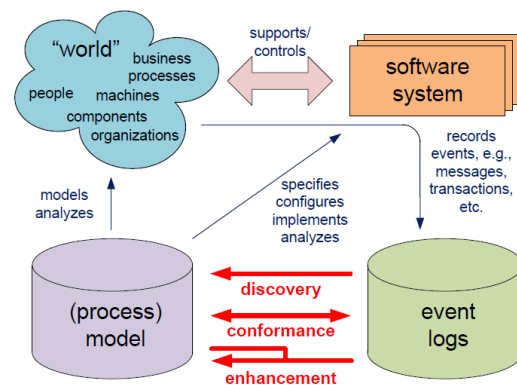
7.3.4 Techniques de fouille de processus (Process mining)

Les entreprises reconnaissent l'importance de la traçabilité des données, discutée plus en détail dans la Section 6.2, et visent à recorder les informations relatives à toutes leurs activités via les systèmes d'information et notamment les PAISs (Process-Aware Information Systems). Dumas et al. [2005] définissent le PAIS comme étant un "software system that manages and executes operational processes involving people, applications, and/or information sources on the basis of process models." ; il est ainsi clair que l'objectif de ce genre de systèmes est de tracer les données relatives aux événements et processus qui ont lieu dans une organisation. La trace issue des PAISs¹¹⁴ est appelée "journal des événements" (Figure 7.14.a) et constitue le point d'entrée des techniques de fouille de processus.

Comme discuté dans la Section 7.3.1, la fouille de processus permet d'explorer, de vérifier et d'améliorer les processus dans une organisation. En effet, en partant d'un journal des événements, cette discipline assure trois tâches comme le montre la Figure 7.14.b : la "découverte des processus" en générant automatiquement leurs modèles, le "contrôle de conformité" entre un modèle de processus existant et la réalité et puis "l'amélioration des processus" en enrichissant le modèle avec les données identifiées dans le journal des événements. Ces trois tâches sont brièvement introduites ci-après.

case id	activity id	originator	timestamp
case 1	activity A	John	9-3-2004:15.01
case 2	activity A	John	9-3-2004:15.12
case 3	activity A	Sue	9-3-2004:16.03
case 3	activity B	Carol	9-3-2004:16.07
case 1	activity B	Mike	9-3-2004:18.25
case 1	activity C	John	10-3-2004:9.23
case 2	activity C	Mike	10-3-2004:10.34
case 4	activity A	Sue	10-3-2004:10.35
case 2	activity B	John	10-3-2004:12.34
case 2	activity D	Pete	10-3-2004:12.50
case 5	activity A	Sue	10-3-2004:13.05
case 4	activity C	Carol	11-3-2004:10.12
case 1	activity D	Pete	11-3-2004:10.14
case 3	activity C	Sue	11-3-2004:10.44
case 3	activity D	Pete	11-3-2004:11.03
case 4	activity B	Sue	11-3-2004:11.18
case 5	activity E	Clare	11-3-2004:12.22
case 5	activity D	Clare	11-3-2004:14.34
case 4	activity D	Pete	11-3-2004:15.56

7.14.a. Exemple d'un journal des événements (Event log) [Van der Aalst, 2005]



7.14.b. Les trois tâches de la fouille de processus selon Van Der Aalst [2012]

FIGURE 7.14 – Les trois tâches de la fouille de processus.

7.3.4.1 Découverte des processus (Process discovery)

Comme discuté dans la Section 6.1, les entreprises tentent à capturer leur savoir-faire, leurs expériences et bonnes pratiques sous la forme de modèles de processus, qui décrivent les objectifs de ses processus métier, les activités que contient chaque processus et les ressources utilisées, entre autres. Ces modèles permettent de spécifier, documenter et analyser les processus [van der Aalst, 2011], et servent de base pour l'adaptation ou le développement de nouveaux processus. La complexité croissante de ces derniers et les changements occurrents qu'ils peuvent subir, font que les modèles de processus sont difficilement construits à la main. La première tâche de la fouille de processus, étant la découverte des processus, permet de pallier ce problème en permettant la génération automatique des modèles de processus à partir des journaux des événements, relatifs aux exécutions passées des processus, qui sont capturées par les systèmes d'information, notamment les PAISs.

114. Nous citons comme exemple de PAISs, les systèmes ERP (Enterprise Resource Planning) comme le SAP, les systèmes CRM (Customer Relationship Management) comme Microsoft Dynamics CRM et les systèmes PDM (Product Data Management) comme Windchill, entre autres [van der Aalst and Gunther, 2007; van der Aalst et al., 2008].

Le point de départ de cette tâche, comme toutes les autres tâches de la fouille des processus d'ailleurs, est un journal des événements ayant la forme présentée dans la Figure 7.14.a. À partir de ce dernier, est automatiquement générée la représentation graphique du processus qui, selon le choix de l'utilisateur, peut être sous la forme de modèles UML [OMG, 2011], BPEL¹¹⁵ [OASIS, 2007], BPMN [OMG, 2013] et réseaux de Pétri [Murata, 1989] entre plusieurs autres formalismes introduits dans le travail de van der Aalst [2011]. En plus de la perspective processus qui se concentre plus sur les processus en permettant la génération de leurs modèles graphiques, deux autres perspectives sont identifiées dans cette tâche. Il s'agit de la perspective organisationnelle qui se concentre plus sur l'aspect humain en identifiant et classifiant les personnes exécutant les processus en termes de rôles et unités organisationnelles, et en identifiant la façon comment elles interagissent lors de l'exécution des processus. L'autre perspective concerne les traces, et a pour objectif de répondre à la question "Quoi" et identifiant les propriétés d'une trace, notamment sa localisation dans le processus, les personnes qui l'exécutent, le temps de son exécution, etc.

Pour illustrer ce qui vient d'être dit, la Figure 7.15.a présente le modèle réseaux de Pétri généré automatiquement à partir du journal des événements de la Figure 7.14.a. Il est clair que la perspective processus, ayant permis de générer ce modèle, répond à la question "Comment" est exécuté le processus, sans tenir compte des personnes qui l'ont exécuté et à l'aide de quelles ressources il a été exécuté. La perspective organisationnelle permet, en revanche, de découvrir les personnes qui sont impliquées lors de l'exécution du processus, ainsi que leurs rôles (Figure 7.15.b) répondant ainsi à la question "Qui" a exécuté le processus. De plus, toujours dans la perspective organisationnelle, il est possible d'identifier les relations entre les personnes exécutant les processus, notamment le transfert de travail comme le montre le sociogramme de la Figure 7.15.c.

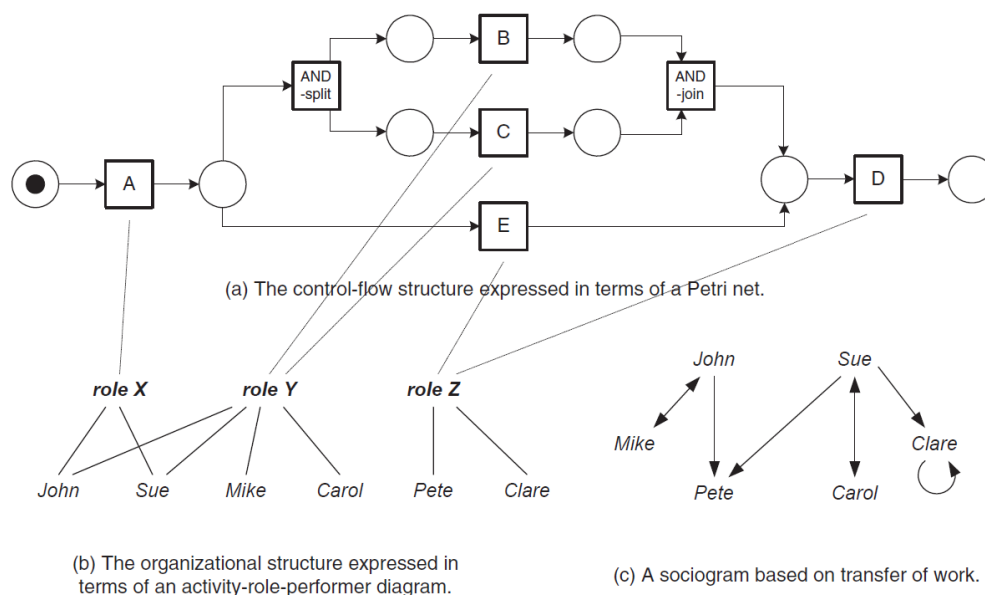


FIGURE 7.15 – Résultats de la découverte des processus : (a) est le processus (exprimé sous la forme d'un réseau de Pétri) découvert à partir du journal des événements de la Figure 7.14.a, (b) et (c) sont respectivement la structure organisationnelle et le transfert d'activités relatifs au même journal des événements.

7.3.4.2 Contrôle de conformité (Conformance checking)

Les modèles générés par la tâche de la découverte des processus ou les modèles prédéfinis dans une organisation peuvent ne pas refléter la réalité d'une façon fidèle, à cause des changements qui affectent les

115. Ou Business Process Execution Language, c'est un langage basé sur le standard XML, qui permet aux services web d'interconnecter et de partager les données dans une architecture orientée services (Service-Oriented Architecture - SOA).

processus, ou si si l'ingénieur compte sur son expertise plutôt que sur un modèle prédéfini, ou encore si le modèle a été généré à partir de très anciennes traces. En effet, pour concevoir un produit donné, un ingénieur doit -selon un modèle prédéfini- commencer par exécuter l'activité "A" ensuite "B" ensuite "C"; cependant l'ingénieur peut exécuter en réalité les activités "A", "X", "B" et "C", violant ainsi l'ordre d'exécution prédéfini. La tâche du contrôle de conformité se voit très intéressante dans ce contexte. Elle vise à vérifier si le modèle prédéfini et l'exécution réelle du processus, capturée sous la forme de journaux des événements, sont conformes en identifiant et localisant les déviations -si elles existent- puis en mesurant leur sévérité [van der Aalst and Gunther, 2007]. Pour mieux expliquer le principe de cette tâche, nous commençons par définir ce qu'est une "relation d'ordre" :

Définition 7.3. Soient T un ensemble de tâches (activités de processus), $\sigma \in T^*$ une trace d'événements et $W \in \mathcal{P}(T^*)$ un journal des événements. Soient $a, b \in T$, les relations d'ordre pouvant exister dans un journal des événements sont définies comme suit :

- $a >_w b$ (a précède b dans w) si et seulement si il existe une trace $\sigma = t_1 t_2 \dots t_{n-1}$ et $i \in \{1, \dots, n-2\}$ telle que $\sigma \in W$ et $t_i = a$ et $t_{i+1} = b$,
- $a \rightarrow_w b$ (a cause b) si et seulement si $a >_w b$ et $b \not>_w a$,
- $a \#_w b$ (a est non-parallèle avec b) si et seulement si $a \not>_w b$ et $b \not>_w a$, et
- $a \parallel_w b$ (a est parallèle avec b) si et seulement si $a >_w b$ et $b >_w a$.

a. T^* est l'ensemble de toutes les séquences qui sont composées de zéro ou plusieurs tâches de T
 b. $\mathcal{P}(T^*)$ est l'ensemble puissance (appelé aussi powerset ou l'ensemble des sous-ensembles) de T^* , $W \in \mathcal{P}(T^*)$ revient à dire que $W \subseteq T^*$

Pour illustrer cette définition, nous reprenons le modèle (a) de la Figure 7.15. De ce modèle et comme expliqué dans la définition 7.3, nous identifions le journal des événements $W = \{ABCD, ACBD, AED\}$. De ce dernier, et en considérant la correspondance entre les relations d'ordre et les concepts relatifs aux réseaux de Pétri (Figure 7.16), sont inférées les relations d'ordre suivantes :

- Précédence : $A >_w B, A >_w C, A >_w E, B >_w C, B >_w D, C >_w B, C >_w D$ et $E >_w D$,
- Causalité : $A \rightarrow_w B, A \rightarrow_w C, A \rightarrow_w E, B \rightarrow_w D, C \rightarrow_w D$ et $E \rightarrow_w D$,
- Parallélisme : $B \parallel_w C$ et $C \parallel_w B$.

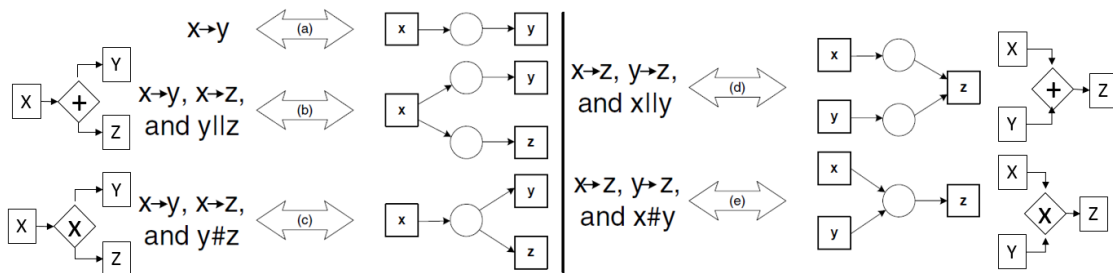


FIGURE 7.16 – Correspondance entre les relations d'ordre, les concepts des réseaux de Pétri et les logiques AND et XOR (adapté de [Van der Aalst and de Medeiros, 2005]).

Maintenant que les relations d'ordre sont présentées et illustrées, nous considérons le modèle prédéfini de la Figure 7.17.a où $W = \{AFGI, AGFI, AGBEDHI, AGBDEHI, AGBCEHI, AGBECHI, ABGDHI\}$ est un journal des événements possible. Le contrôle de conformité permet de vérifier si ce modèle est conforme avec les exécutions réelles capturées par les journaux des événements 7.17.b, 7.17.c, 7.17.d et 7.17.e. Le journal des événements 7.17.b contient quatre traces (appelées aussi cas ou case), dans la première trace sont enregistrés quatre événements (appelés aussi activités ou tâches) : A, G, F et I. En jouant le "jeu de jetons" des réseaux de Pétri, nous nous rendons compte que cette séquence d'événements est tout à fait possible dans le modèle prédéfini parce que $A >_w G, A >_w F$ et $F \parallel_w G$, pareillement, toutes les autres traces du journal des événements 7.17.b sont possibles dans le modèle prédéfini. Les traces du journal

7.17.c sont aussi possibles dans le modèle mais n'en représente qu'une partie (la partie basse qui comporte les activités A , F , G et I). La première trace du journal 7.17.d n'est pas possible à cause de l'absence de G dans la trace alors que dans le modèle $F \parallel_w G$ et donc le jeton émis de la place de départ doit se diviser pour occuper les places p_1 et p_2 en parallèle, donnant ainsi lieu à l'exécution des activités F et G . De la même façon, la quatrième trace du même journal n'est pas possible dans le modèle à cause de l'absence de l'activité G . Passons maintenant au journal des événements 7.17.e, les deux premières traces ne sont pas possibles dans le modèle parce que l'activité B n'y est pas présente alors que $B >_w E$ dans le modèle.

Le contrôle montre que le modèle prédéfini de la Figure 7.17.a n'est pas conforme aux journaux des événements reflétant la réalité et que deux problèmes sont identifiés dans ce modèle prédéfini. Le premier problème concerne le "underfitting" où le journal des événements capture un comportement non présent dans le modèle prédéfini (comme dans le cas des journaux 7.17.d et 7.17.e). Le deuxième problème concerne le "overfitting" où le journal contient des comportements possibles dans le modèle prédéfini mais qu'une partie des comportements décrits par ce dernier ne sont pas capturés par le journal des événements (cas du journal 7.17.c).

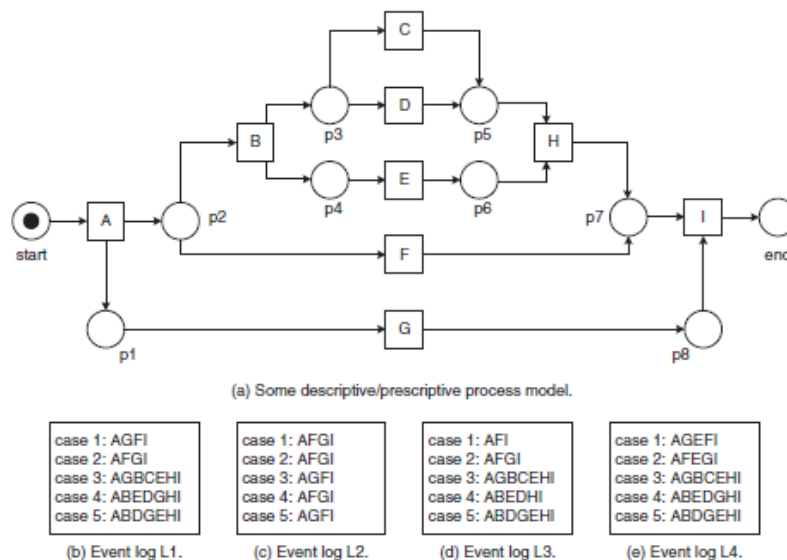


FIGURE 7.17 – Contrôle de conformité entre un processus prédéfini (a) et quelques exécutions réelles capturées dans les journaux des événements (b,c,d et e) [Van der Aalst, 2005]

7.3.4.3 Amélioration des processus (Enhancement)

Gardons toujours en tête la notion de modèle prédéfini ; ce dernier -s'il n'est pas régulièrement entretenu- peut ne plus être à jour. La troisième tâche de la fouille de processus intervient ici pour étendre ce modèle prédéfini, ou n'importe quel modèle a-priori, en introduisant de nouveaux aspects et perspectives [Van Der Aalst, 2012]. Le but de cette tâche n'étant pas de mesurer l'alignement entre un modèle et la réalité comme l'assure le contrôle de conformité, mais plutôt d'utiliser les informations relatives à l'exécution réelle (capturées dans les journaux des événements) pour enrichir ou améliorer un modèle donné [van der Aalst and Gunther, 2007]. van der Aalst [2011] identifie deux types d'amélioration, la première -étant la réparation- consiste à modifier le modèle pour mieux refléter la réalité. Par exemple, le modèle où deux activités sont modélisées séquentiellement mais qui peuvent en réalité se produire en parallèle, peut être modifié pour refléter cela (Figure 7.18). Le deuxième type d'amélioration est l'extension, c'est-à-dire l'ajout d'une nouvelle perspective au modèle de processus en le corrélant avec le journal des événements. Un exemple de ce type est l'extension d'un modèle de processus avec les données de performance pour identifier et localiser les goulots d'étranglement, le temps de traitement, les ressources et les règles de décision, entre autres [van der Aalst, 2011].

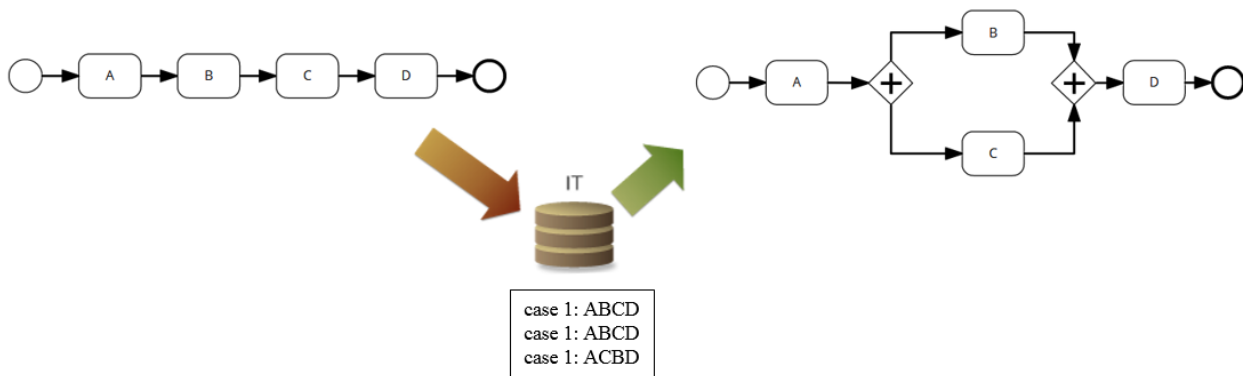


FIGURE 7.18 – La troisième tâche de la fouille de processus : amélioration d'un modèle de processus existant (adapté de <http://fluxicon.com/>).

7.3.5 Synthèse et discussion de l'état de l'art sur les techniques de l'intelligence artificielle et de la business intelligence supportant la prise de décision

Dans la Section 7.2, nous avons défini l'aide à la décision dans notre contexte, en distinguant deux grands types :

- L'aide à la décision **globale** : elle concerne l'identification et la proposition du ou des patterns (exécution d'un processus du début à la fin) les plus adaptés à un contexte de décision donné,
- L'aide à la décision **locale** : elle concerne l'identification et la proposition des paramètres (ressources et solutions admissibles/finales (Figure 7.4)) les plus significatifs dans un contexte de décision donné.

De plus, dans la Section 7.3, nous avons fait un tour d'horizon sur les techniques de l'intelligence artificielle que nous supposons être les plus basiques. En effet, nous avons introduit les quatre techniques de l'apprentissage automatique, les trois techniques de la fouille de processus, et malgré l'existence de plusieurs techniques de fouille de données, nous nous sommes limités à l'étude des règles d'association et de la fouille de séquences qui se voient les plus populaires et les plus pertinents quant à notre contexte de recherche. Bien que les autres techniques de fouille de données -comme la fouille des itemsets [Goethals and Zaki, 2004]¹¹⁶ et la fouille des épisodes [Leemans and van der Aalst, 2015]¹¹⁶ - sont également aussi populaires, elles ne peuvent pas être appliquées dans notre contexte. En effet, la fouille des itemsets se voit souvent comme une des premières étapes de la fouille des règles d'association que nous avons étudiée [Agrawal et al., 1993]. De plus, un épisode, quant à lui, se voit comme un ensemble partiellement¹¹⁷ ordonné des activités [Leemans and van der Aalst, 2015]. Par conséquent, la fouille des épisodes n'est pas, non plus, adaptée à notre contexte parce que l'ordre défini des activités dans un pattern nous est très intéressant. En effet, le pattern contenant les activités $\{A, B, C, D\}$ est différent de celui qui contient les activités $\{A, C, B, D\}$.

Passons maintenant à l'étude de la possibilité de l'utilisation des techniques que nous avons introduites, pour cela nous considérons le modèle de processus de la Figure 7.19 qui contient cinq différents patterns $P_1 = \{A_1, A_2, A_3, A_4, A_8, A_9\}$, $P_2 = \{A_1, A_2, A_3, A_4, A_9, A_8\}$, $P_3 = \{A_1, A_2, A_3, A_5, A_8, A_9\}$, $P_4 = \{A_1, A_2, A_3, A_5, A_9, A_8\}$ et $P_5 = \{A_6, A_7\}$. Chacune des activités des différents patterns peut être exécutée en utilisant une des deux ressources disponibles R_1 et R_2 . Une technique est dite applicable dans notre contexte si et seulement si elle permet -automatiquement et sans interaction avec l'environnement- l'aide à la décision aux niveaux global (identifier lequel des patterns $P_{1 \Rightarrow 5}$ est le plus significatif pour un contexte de décision donné) et local (identifier laquelle des ressources, entre R_1 et R_2 , est la plus adaptée au contexte de la décision), comme discuté dans la Section 7.2. Le résultat de la comparaison des différentes techniques par rapport à nos deux critères est présenté dans la Figure 7.20, qui est synthétisée et discutée ci-dessous.

¹¹⁶. Cette référence n'est pas celle qui a premièrement introduit le concept, nous l'avons citée parce qu'elle définit ce dernier, étant sujet à plusieurs interprétations, d'une façon simple et pertinente.

¹¹⁷. Contrairement à Wu et al. [2013] qui voit la fouille des épisodes comme un ensemble totalement ordonné d'activités, et conformément à Mannila et al. [1997] qui considèrent l'ordre partiel dans la définition.

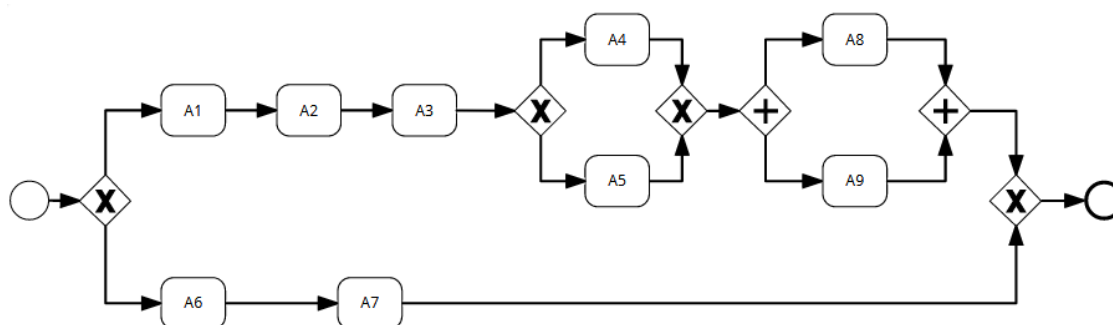


FIGURE 7.19 – Exemple d'un processus à cinq patterns différents (exprimé en BPMN [OMG, 2013]).

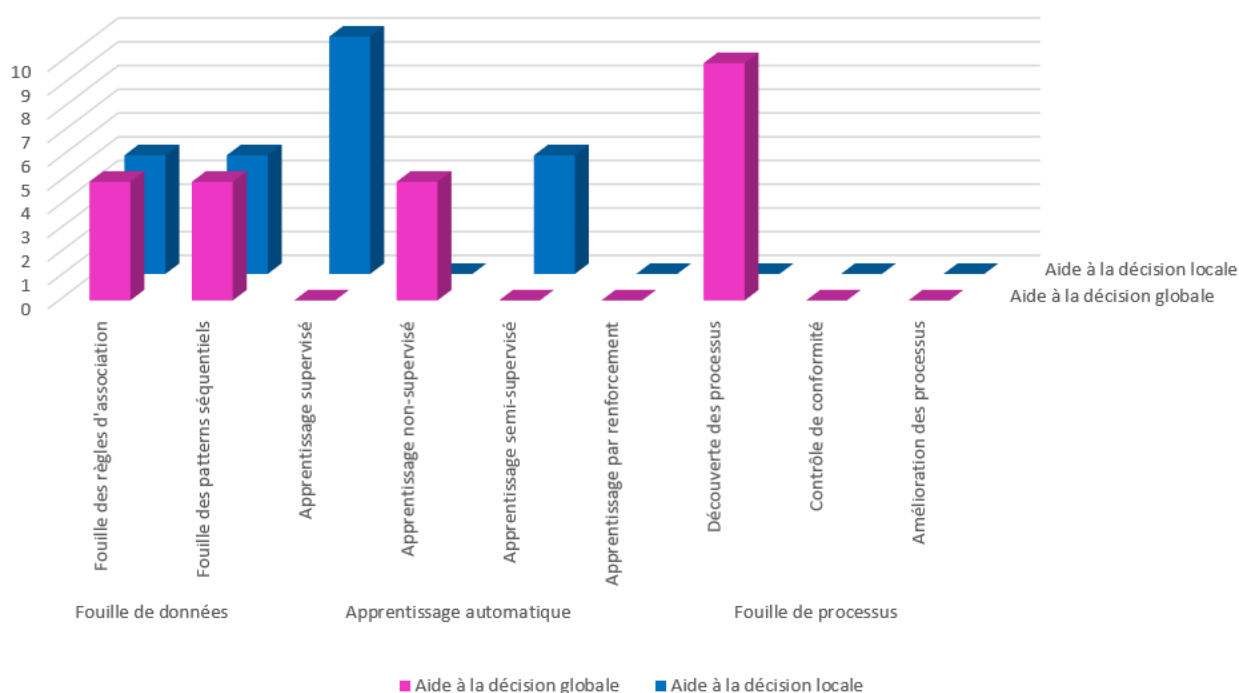


FIGURE 7.20 – Résultat de l'étude de l'application des techniques étudiées dans notre contexte (La note 0 veut dire que la technique n'est pas applicable, 5 pour dire qu'elle est difficilement applicable sans efforts d'adaptation et 10 pour dire qu'elle est tout à fait applicable).

Concernant les techniques de la fouille de données : La fouille des règles d'association se voit difficilement applicable aux deux niveaux de l'aide à la décision. En effet, les règles d'association dans ce contexte peuvent être de la forme :

- $A_1 \Rightarrow A_2$ avec $s = 0.95$ et $c = 0.95$ par exemple ¹¹⁸,
- $A_2 \Rightarrow A_3$ avec $s = 1.00$ et $c = 0.90$,
- $A_1 \Rightarrow R_1$ avec $s = 0.85$ et $c = 0.90$,
- $A_1 \Rightarrow R_2$ avec $s = 0.05$ et $c = 0.10$, etc.

Il est donc possible de découvrir les différents patterns, en **associant** les règles pour identifier les ensembles ordonnés des activités. De la même façon, il est aussi possible de déterminer quelle ressource est impliquée par quelle activité. Cependant, L'exploration des patterns et des paramètres devient beaucoup plus compliquée et très couteuse en temps lorsqu'il s'agit des processus contenant un nombre important d'activités qui elles aussi impliquent l'utilisation de plusieurs paramètres en même temps (par exemple, la ressource humaine "x" utilise le logiciel "y" et la machine "z" pour exécuter l'activité "A").

118. Le calcul des indices de support "s" et de confiance "c" n'est pas rigoureux, les valeurs sont données à titre d'exemple.

La fouille des patterns séquentiel se voit aussi difficilement applicable dans notre contexte. En effet, considérons la base de données du Tableau 7.4, les patterns séquentiels qui peuvent être déduits de cette base sont : $\langle (c) (i) \rangle$ et $\langle (c) (d g) \rangle$ pour un support minimum de 25%, comme expliqué dans la Section 7.3.2.2. Il est intéressant de noter que ces deux patterns séquentiels ne reflètent pas les différents patterns existants, à savoir, les cinq séquences de la Section 7.3.2.2, par conséquent l'utilisation de ce type de fouille de données nécessite des efforts d'adaptation non négligeables. Concernant les paramètres des activités, c'est vrai qu'il est possible de les formuler sous la forme de séquences $\langle (R_1) (R_1 R_2) (R_3) \dots \rangle$, mais la correspondance entre les activités et les paramètres nécessite d'être travaillée en plus des efforts d'adaptation qui doivent être faits pour s'assurer que les patterns séquentiels soient bien les patterns relatifs à l'exécution des processus du début à la fin.

Concernant les techniques de l'apprentissage automatique : La classification supervisée se voit tout à fait possible pour prédire les paramètres des activités les plus significatifs dans un contexte de décision donné. En effet, ayant l'activité en question et ses anciennes traces d'exécution, le problème peut être formulé comme suit :

- Entraîner un modèle d'apprentissage en considérant le vecteur d'entrée X qui se compose du nom de l'activité et du nom de celui qui l'exécute, entre autres et le label Y étant le paramètre d'intérêt (comme la ressource logicielle utilisée par exemple) qui est déjà connu dans les anciennes traces,
- Après avoir entraîné et validé le modèle, l'utiliser pour prédire le label (ressource logicielle en l'occurrence) le plus adapté tenant compte du contexte.

L'apprentissage supervisé se voit, en revanche, difficilement applicable à la découverte des processus, parce que la découverte des patterns -contrairement à la prédiction des paramètres- n'est associée à aucune information à-priori.

L'apprentissage non-supervisé n'est pas du tout le bon choix pour prédire les paramètres des activités, parce que sa tâche principale est le partitionnement qui ne permet absolument pas la prédiction des situations futures. Cependant, il peut être vu comme un candidat intéressant pour identifier les différents patterns dont chacun fait référence à un des clusters; les efforts de vérification des patterns identifiés doivent tout de même se multiplier pour s'assurer d'avoir un résultat correct. Bien qu'il nécessite des efforts d'adaptation, l'apprentissage semi-supervisé peut, lui aussi, être vu comme un candidat intéressant pour la prédiction des paramètres à travers la classification semi-supervisée, mais ne se voit pas applicable pour découvrir les patterns, parce que le partitionnement semi-supervisé nécessite des informations à priori.

L'apprentissage par renforcement n'est pas applicable dans notre contexte, parce que l'identification des patterns et des paramètres se fait par essais/erreurs et à l'aide de l'interaction avec l'environnement, chose que nous souhaitons pas avoir (l'utilisateur n'intervient qu'au niveau de la validation ou le refus de la décision proposée, qui doit quand même avoir un certain niveau de certitude).

Concernant les techniques de fouille de processus : La fouille des processus ne s'intéresse pas aux paramètres des activités comme matière pour la fouille, ceci dit que les trois tâches ne sont pas applicables à l'aide à la décision locale. Cependant, la découverte des processus est un excellent candidat pour découvrir les différents patterns dans un processus via les perspectives "processus" et "trace" permettant d'identifier les modèles de processus, les patterns et d'étudier les propriétés des différentes traces. Le contrôle de conformité et l'amélioration des processus ne sont pas vraiment faits pour découvrir les patterns, mais sont plutôt orientés vers le contrôle et la mise à jour des modèles de processus.

De la Figure 7.20 et de la synthèse ci-haut, nous considérons par la suite les techniques de l'apprentissage automatique, notamment la classification supervisée, pour l'aide à la décision locale et la fouille de processus, notamment la découverte des processus, pour l'aide à la décision globale. Bien que ce couplage de la fouille des processus et de l'apprentissage automatique est à notre connaissance nouveau dans le domaine de la conception et la supervision des systèmes, il a déjà été considéré dans quelques domaines de recherche. Nous citons par exemple le travail de Reimann et al. [2010] qui proposent un outil permettant de supporter l'écriture collaborative d'un document électronique. L'outil utilise les techniques

de l'apprentissage automatique pour extraire les documents et les techniques de fouille des processus pour explorer les journaux des événements, qui capturent le comportement de l'utilisateur, en générant le modèle de processus associé. Un autre travail paraît intéressant et très proche du notre, il s'agit de celui de [Crierie et al. \[2009\]](#) qui proposent une méthode de découverte des règles métier. La fouille de processus est d'abord utilisée pour analyser les informations du journal des événements en identifiant la catégorie des utilisateurs qui exécutent l'activité. L'apprentissage automatique est ensuite utilisé pour analyser les informations de contexte, existant dans le journal des événements, afin de découvrir l'ensemble des paramètres possibles pour une activité donnée.

Plus généralement, les travaux qui couplent ces deux domaines utilisent les techniques de l'apprentissage automatique pour extraire les connaissances et trouver les relations entre elles. Ensuite, ils utilisent les techniques de fouille de processus pour identifier le modèle de processus à partir du journal des événements enregistrés. Notre travail de recherche adopte la même logique pour une double analyse de processus, comme cela est détaillé dans la section suivante.

7.4 Module d'aide à la décision : Aide à la décision globale et locale

7.4.1 Processus de l'aide à la décision et exemple d'illustration

Pour pouvoir aider les utilisateurs aux niveaux local et global identifiés, le module de l'aide à la décision doit [[Es-Soufi et al., 2016](#)] :

- **Exigence 1** : commencer par identifier les différents patterns possibles à partir des traces relatives aux anciennes exécutions des processus et ensuite découvrir les règles de décision qui contrôlent leur choix (étapes 1, 2 et 3 de la Figure 7.21),
- **Exigence 2** : proposer à l'utilisateur le ou les patterns les plus adaptés au contexte de la décision (étape 4 de la Figure 7.21),
- **Exigence 3** : après avoir eu la réponse de l'utilisateur (étape 5 de la Figure 7.21), prédire les paramètres les plus adaptés au contexte de la décision pour chaque activité du pattern choisi, et les proposer à l'utilisateur (étapes 6, 7, 8, 9 et 10 de la Figure 7.21).

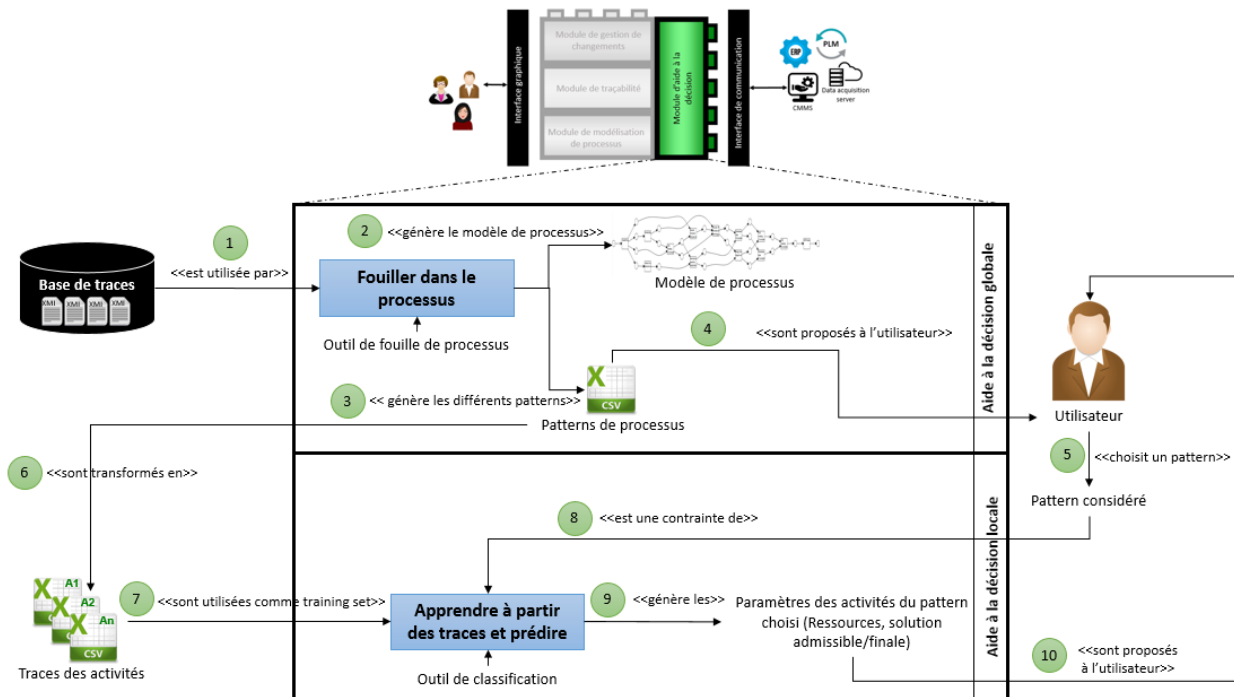


FIGURE 7.21 – Processus général du module de l'aide à la décision ($A \xrightarrow{\ll\text{utilise}\gg} x$ se lit "A utilise x").

Chacune de ces trois étapes sera présentée et expliquée plus en détails dans les sous-sections suivantes. Pour permettre une explication simple et pertinente, nous introduisons un exemple de référence, que nous appelons exemple d'illustration par la suite. Cet exemple illustratif, étant construit à la main¹¹⁹, décrit le processus le plus général possible¹²⁰ que nous pouvons rencontrer lors de la conception ou la supervision d'un système (Figure 7.22). De plus, nous le considérons comme **notre référence pour comparer les algorithmes que nous étudions par la suite**.

! L'algorithme qui répondra à nos exigences doit avoir la capacité de retrouver les règles de décision, contrôlant le choix des différents patterns de notre processus de référence, **exactement** comme écrit en rouge dans la Figure 7.22.

Le Tableau 7.8 présente les différents patterns du processus illustratif, dont chacun est une trace de format XMI, issue¹²¹ du module de traçabilité.

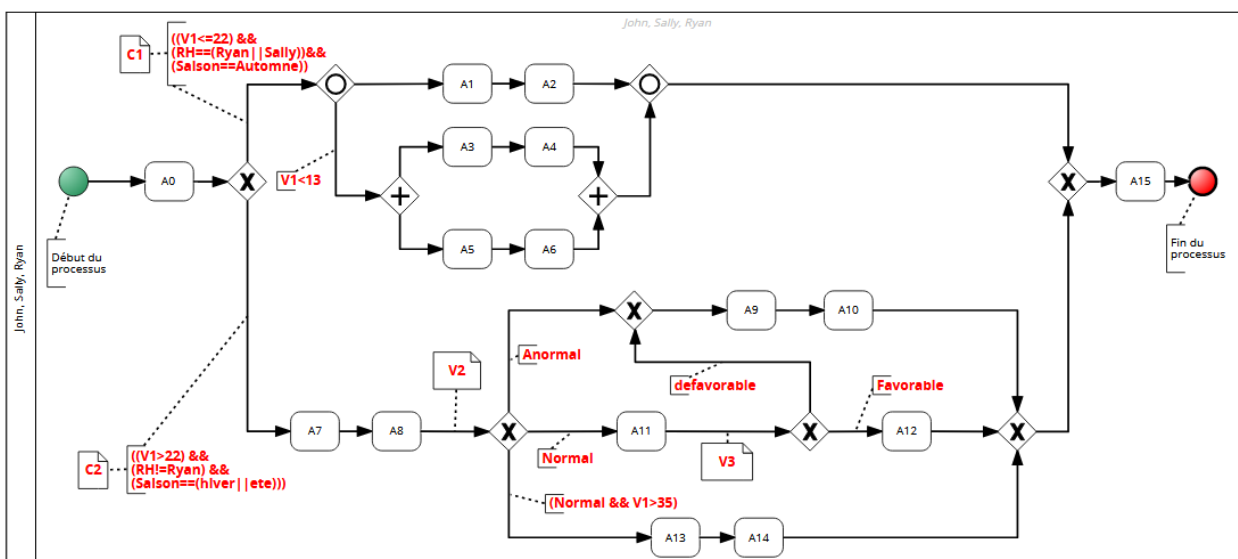


FIGURE 7.22 – Processus illustratif où les trois jonctions (And, Or et Xor) sont présentes (exprimé en BPMN [OMG, 2013]).

Notre processus (Figure 7.22) est contrôlé par cinq paramètres (que nous appelons **variables de contexte** par la suite) : *RH* (Ressource Humaine¹²², de type chaîne de caractères pouvant prendre plusieurs valeurs selon les personnes exécutant le processus, ici nous nous limitons à trois), *saison* (de type chaîne de caractères, prenant une des valeurs : automne, hiver ou été), V_1 (de type réel), V_2 (de type chaîne de caractères, prenant une des deux valeurs : Anormal ou Normal) et V_3 (de type chaîne de caractères, prenant une des deux valeurs : Favorable ou Défavorable). En effet, trois utilisateurs exécutent le processus, il s'agit de *John*, *Ryan* et *Sally*. Après avoir exécuté l'activité A_0 , deux choix exclusifs possibles se présentent (logique Xor) dont **un et un seul** doit être pris en compte :

119. Nous avons construit ce modèle de processus en adoptant BPMN comme langage de modélisation et en utilisant l'outil Signavio : <https://www.signavio.com/>.

120. En effet, notre exemple d'illustration considère toutes les jonctions possibles : And, Or et Xor comme décrit dans le Tableau 6.1, donnant ainsi lieu à plusieurs patterns dont chacun fait référence à un choix d'exécution possible. De plus, notre exemple illustratif considère la possibilité de la présence des mêmes activités dans des patterns différents (activités A_9 et A_{10} dans la Figure 7.22).

121. Les traces (ou patterns) du processus illustratif de la Figure 7.22 sont instanciées à partir du méta-modèle de trace, dans l'environnement Eclipse (Section 6.2)

122. Il est aussi possible que la variable de contexte concernant la ressource humaine ne soit pas son nom mais plutôt sa compétence principale ou son rôle, entre autres.

- Si la condition " $C_1 = ((V_1 \leq 22) \ \&\& \ (RH == (Ryan \ || \ Sally)) \ \&\& \ (Saison == Automne))$ "¹²³ est vérifiée, c'est la partie haute du processus qui est prise en compte. Là, plusieurs autres chemins sont possibles, dont **un ou plusieurs** peuvent être pris en compte (logique *Or*). Les activités A_1 et A_2 seront, dans tous les cas, exécutées parce que C_1 est vérifiée. Si, maintenant, en plus de C_1 , le paramètre ($V_1 < 13$), les activités A_3, A_4, A_5 et A_6 seront exécutées en parallèle avec A_1 et A_2 . La jonction "*Or*" s'est ainsi transformée en "*And*" comme le montre la Figure 7.23. Le parallélisme au niveau de l'exécution est contraint par la précedence des activités, en effet, A_2 par exemple ne peut pas s'exécuter avant A_1 , ni A_4 avant A_3 , ni A_6 avant A_5 . En revanche, A_3 peut bien s'exécuter avant A_2 , A_5 avant A_4 , et ainsi de suite. La partie haute du processus donne lieu aux patterns $P_{5 \rightarrow 50}$ définis dans le Tableau 7.8. Notons que rien que cette partie du processus illustratif peut s'exécuter de 94 façons différentes, d'où l'intérêt d'être assisté lors de la prise de décision.
- Si la condition " $C_2 = ((V_1 > 22) \ \&\& \ (RH! = Ryan) \ \&\& \ (Saison == (hiver \ || \ ete)))$ "¹²³ qui est vérifiée, la partie basse du processus sera prise en compte. Après avoir exécuté les activités A_7 et A_8 , trois chemins exclusifs sont possibles, dont un seul sera pris en compte, selon la valeur du paramètre V_2 issu (mesuré ou calculé) de l'exécution de A_8 . Si ($V_2 == Anormal$), les activités A_9 et A_{10} peuvent être exécutées (P_1 du Tableau 7.8), sinon, si ($V_2 == Normal$), l'activité A_{11} peut s'exécuter, et donne lieu après à deux autres chemins possibles contrôlés par le paramètre V_3 (A_{12} s'exécute si ($V_3 == Favorable$)) donnant lieu au pattern P_2 du Tableau 7.8, sinon les activités A_9 et A_{10} s'exécutent (P_3 du Tableau 7.8). Si maintenant en plus de ($V_2 == Normal$), le paramètre ($V_1 > 35$), alors ce sont les activités A_{13} et A_{14} qui seront exécutées après A_8 (P_4 du Tableau 7.8).
- Après l'exécution d'une des deux parties exclusives haute ou basse (comme expliqué ci-haut), l'activité A_{15} s'exécute et le processus arrive à sa fin.

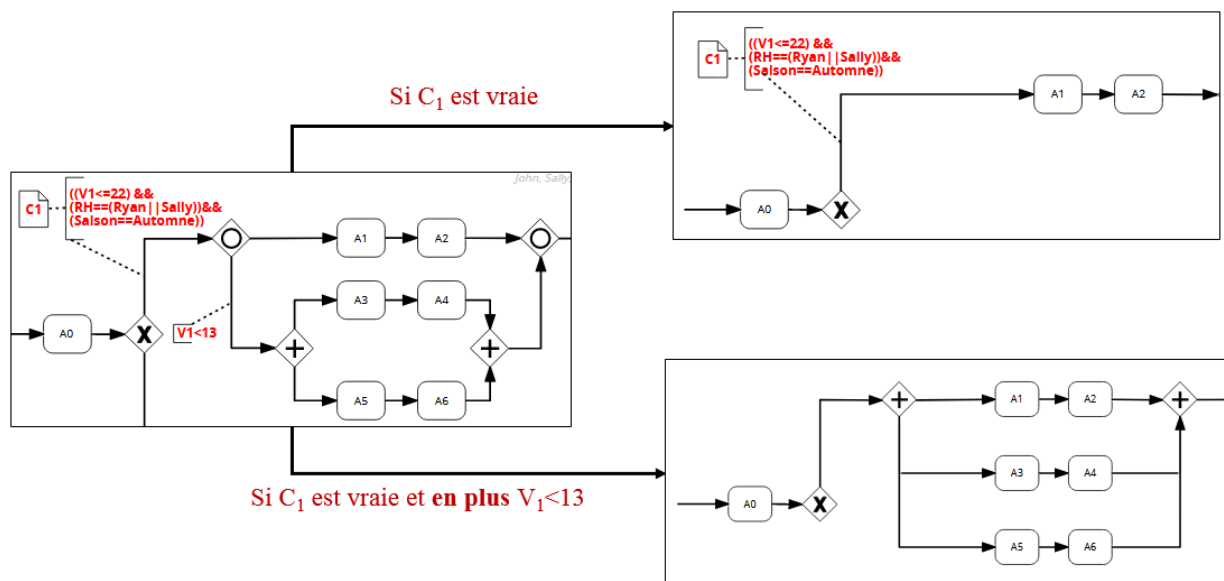


FIGURE 7.23 – Transformation de la jonction "*Or*" en "*And*" si toutes ses conditions sont vérifiées (Extrait du processus de la Figure 7.22).

Maintenant que nous avons introduit le processus illustratif qui nous aidera à expliquer notre proposition et choisir l'algorithme à adopter, nous passons à la préparation du journal des événements relatif aux exécutions passées du processus illustratif (Figure 7.22). Ensuite, nous passons à la discussion des deux étapes générales du processus de l'aide à la décision (Figure 7.21).

123. " $\&\&$ " est le "*et*" logique : la condition ($a \ \&\& \ b$) est vraie si et seulement si a est vrai et b est vrai. " $\|$ " est le "*ou*" logique : ($a \ \| \ b$) est vraie si un des deux paramètres est vrai. " $==$ " s'utilise pour vérifier l'égalité contrairement à " $=$ " qui s'utilise pour affecter une variable à une autre. ($a! = b$) revient à dire que $a \neq b$.

Pattern	Activités						
P_1	A_0	A_7	A_8	A_9	A_{10}	A_{15}	
P_2	A_0	A_7	A_8	A_{11}	A_{12}	A_{15}	
P_3	A_0	A_7	A_8	A_{11}	A_9	A_{10}	A_{15}
P_4	A_0	A_7	A_8	A_{13}	A_{14}	A_{15}	
P_5	A_0	A_1	A_2	A_{15}			
P_6	A_0	A_1	A_2	A_3	A_4	A_5	A_6
P_7	A_0	A_1	A_2	A_3	A_5	A_4	A_6
P_8	A_0	A_1	A_2	A_3	A_5	A_6	A_4
P_9	A_0	A_1	A_2	A_5	A_6	A_3	A_4
P_{10}	A_0	A_1	A_2	A_5	A_3	A_6	A_4
P_{11}	A_0	A_1	A_2	A_5	A_3	A_4	A_6
P_{12}	A_0	A_3	A_4	A_1	A_2	A_5	A_6
P_{13}	A_0	A_3	A_4	A_1	A_5	A_2	A_6
P_{14}	A_0	A_3	A_4	A_1	A_5	A_6	A_2
P_{15}	A_0	A_3	A_4	A_5	A_6	A_1	A_2
P_{16}	A_0	A_3	A_4	A_5	A_1	A_6	A_2
P_{17}	A_0	A_3	A_4	A_5	A_1	A_2	A_6
P_{18}	A_0	A_5	A_6	A_1	A_2	A_3	A_4
P_{19}	A_0	A_5	A_6	A_1	A_3	A_2	A_4
P_{20}	A_0	A_5	A_6	A_1	A_3	A_4	A_2
P_{21}	A_0	A_5	A_6	A_3	A_4	A_1	A_2
P_{22}	A_0	A_5	A_6	A_3	A_1	A_4	A_2
P_{23}	A_0	A_5	A_6	A_3	A_1	A_2	A_4
P_{24}	A_0	A_1	A_3	A_2	A_4	A_5	A_6
P_{25}	A_0	A_1	A_3	A_4	A_2	A_5	A_6
P_{26}	A_0	A_1	A_3	A_4	A_5	A_2	A_6
P_{27}	A_0	A_1	A_3	A_4	A_5	A_6	A_2
P_{28}	A_0	A_1	A_3	A_2	A_5	A_4	A_6
P_{29}	A_0	A_1	A_3	A_5	A_2	A_4	A_6
P_{30}	A_0	A_1	A_3	A_5	A_4	A_2	A_6
P_{31}	A_0	A_1	A_3	A_5	A_4	A_6	A_2
P_{32}	A_0	A_1	A_3	A_2	A_5	A_6	A_4
P_{33}	A_0	A_1	A_3	A_5	A_2	A_6	A_4
P_{34}	A_0	A_1	A_3	A_5	A_6	A_2	A_4
P_{35}	A_0	A_1	A_3	A_5	A_6	A_4	A_2
P_{36}	A_0	A_1	A_5	A_2	A_6	A_3	A_4
P_{37}	A_0	A_1	A_5	A_6	A_2	A_3	A_4
P_{38}	A_0	A_1	A_5	A_6	A_3	A_2	A_4
P_{39}	A_0	A_1	A_5	A_6	A_3	A_4	A_2
P_{40}	A_0	A_1	A_5	A_2	A_3	A_6	A_4
P_{41}	A_0	A_1	A_5	A_3	A_2	A_6	A_4
P_{42}	A_0	A_1	A_5	A_3	A_6	A_2	A_4
P_{43}	A_0	A_1	A_5	A_3	A_6	A_4	A_2
P_{44}	A_0	A_1	A_5	A_2	A_3	A_4	A_6
P_{45}	A_0	A_1	A_5	A_3	A_2	A_4	A_6
P_{46}	A_0	A_1	A_5	A_3	A_4	A_2	A_6
P_{47}	A_0	A_1	A_5	A_3	A_4	A_6	A_2
P_{48}	A_0	A_3	A_1	A_4	A_2	A_5	A_6
P_{49}	A_0	A_3	A_1	A_2	A_4	A_5	A_6
P_{50}	A_0	A_3	A_1	A_2	A_5	A_4	A_6

TABLE 7.8 – 50 patterns, parmi les 94 possibles, de l'exemple d'illustration de la Figure 7.22

Il est intéressant de noter que les seules données que peut avoir le module d'aide à la décision, sont les traces sous le format XMI (Figure 7.21), et que le modèle de processus illustratif de la Figure 7.22, ainsi que ses patterns définis dans le Tableau 7.8 et les règles de décision contrôlant leur choix, ne sont pas connus à-priori. Nous les avons introduits pour **expliquer** la logique de la contribution et pour **vérifier** l'exactitude des résultats des algorithmes de la fouille de processus. En effet, l'algorithme -dont les règles de décision identifiées sont les plus proches de celles que nous avons considérées dans la Figure 7.22- est choisi.

7.4.2 Préparation du journal des événements

Comme cela a été déjà discuté dans la Section 7.3.4, le point d'entrée des outils de fouille de processus est un ensemble d'informations sur les exécutions passées des processus, enregistrées dans un journal des événements ayant généralement un des deux formats spécifiques : MXML (Mining eXtensible Markup Language) et XES (eXtensible Event Stream), dont les méta-modèles sont respectivement introduits dans les travaux de [van Dongen and Van der Aalst \[2005\]](#) et [Gunther and Verbeek \[2014\]](#). Cependant, les traces issues du module de traçabilité -et qui sont sauvegardées **séparément** dans une base de traces (Figure 7.21), dont chacune est une ancienne exécution d'un processus donné- ont le format XMI. La question qui se pose à ce stade là est :



Verrou technologique 1

Comment passer d'un **ensemble de traces**, enregistrées séparément sous la forme de fichiers **XMI**, à un **seul journal des événements** exprimé sous forme d'un fichier **MXML** ou **XES** ?

Avant d'essayer de lever ce premier verrou technologique ¹²⁴, nous devons d'abord choisir quel langage considérer, entre MXML et XES, pour construire le journal des événements. Les deux langages sont compatibles avec les outils de fouille de processus existants dans le marché. MXML, étant basé sur la syntaxe XML ¹²⁵ et dont le méta-modèle est présenté dans la Figure 7.24.a, est apparu en 2003 ¹²⁶ et a été considéré, depuis, comme le format standard des journaux des événements. En 2016, un autre format, basé également sur la syntaxe XML mais permettant de mieux capturer le comportement des systèmes ¹²⁷, a été considéré comme le nouveau standard par le "IEEE-SA Standards Board" ¹²⁸. Il s'agit du format XES, dont le méta-modèle est présentée dans la Figure 7.24.b et qui est considéré comme moins restrictif et plus extensible que le format MXML. Par conséquent, nous faisons le choix de considérer le format XES pour construire nos journaux des événements, car en plus qu'il soit extensible et moins restrictif, il est largement documenté et considéré par la communauté de la fouille des processus.

L'objet principal du langage XES est le **Log** (Figure 7.24.b) qui est le journal des événements contenant toutes les traces relatives à un processus spécifique. En effet, le log du processus "Préparer une tarte au chocolat" est différent du log relatif au processus "Nettoyer le moteur d'une lamborghini aventador sv". Un log peut contenir une ou plusieurs **traces** d'un processus donné, sachant qu'une trace est une exécution spécifique du processus ¹²⁹, et qu'elle peut contenir un ou plusieurs **événements** (que nous avons appelés activités de processus). Un événement peut contenir un ou plusieurs **attributs** pouvant être de différents types (chaîne de caractères, entier, réel, booléen, etc.) et ayant une clé qui l'identifie (**key** sur la Figure 7.24.b) ainsi qu'une **valeur** (`<typeAttribut key="nomAttribut" value="valeurAttribut"/>`). La norme XES ne définit pas un ensemble spécifique d'attributs pouvant exister dans le log pour ne pas limiter l'utilisateur. La sémantique des attributs peut donc être ambiguë, impactant ainsi l'interprétation des données. Le concept d'**extension** permet de pallier ce problème car il permet de définir un ensemble d'attributs en lui attachant une sémantique. Par exemple, l'extension "**cycle de vie**" ¹³⁰ permet de spécifier la transition de cycle de vie représentée par chaque événement du log : l'attribut **schedule** veut dire que l'activité doit être exécutée, l'attribut **start** veut dire que l'exécution de l'activité commence et l'attribut **complete** veut dire que l'exécution de l'activité est terminée. Un autre concept se voit très utile pour le langage XES, il s'agit du classificateur (**classifier** dans la Figure 7.24.b) qui a pour but d'affecter une identité à chaque événement pour lui permettre d'être comparable aux autres événements du log. Par exemple, le classificateur dont le nom est `ActivityClassifier` et les clés sont `ressource` et `statut`, considère les événements e_1 et e_2 , dont $nom_{e_1} = nom_{e_2}$ et $statut_{e_1} = statut_{e_2}$, comme des événements égaux.

Il est parfois possible d'avoir des activités qui durent très longtemps (une semaine par exemple) ou dont la durée d'exécution est supérieure à l'intervalle de temps dans lequel le système d'information génère la trace, ceci donne lieu à ce qu'on appelle les "traces incomplètes". Le standard XES permet de refléter cela à l'aide des attributs de transition "start" et "complete" de l'extension "cycle de vie" ¹³⁰. Une

124. Il est important de noter la différence entre un verrou scientifique et un verrou technologique. Le premier est lié à une limite conceptuelle (comme l'absence d'une méthode permettant de résoudre une problématique donnée), tandis que le deuxième est lié à une limite technologique (comme l'absence d'une application permettant l'implémentation de la méthode identifiée.)

125. eXtensible Markup Language : <https://www.w3.org/TR/xml/>

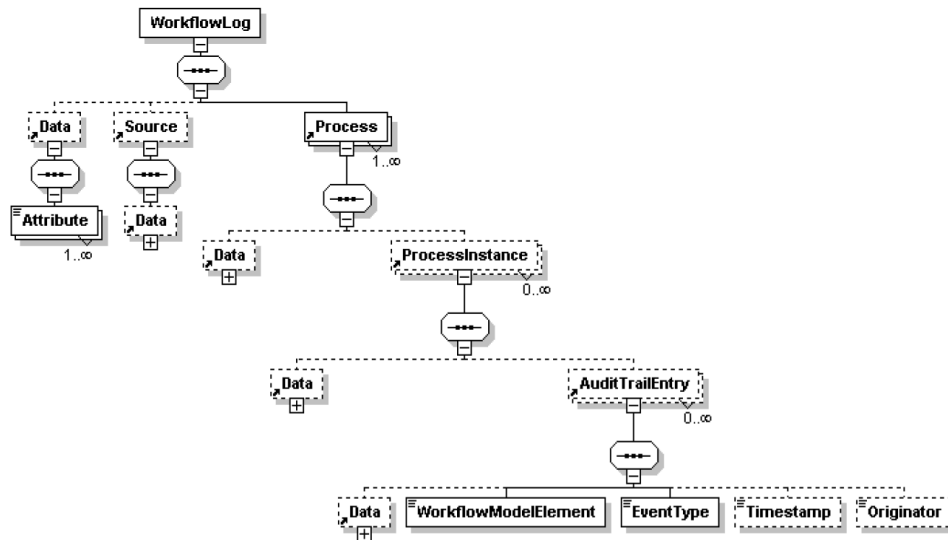
126. <http://www.processmining.org/logs/mxml>

127. <http://www.xes-standard.org/>

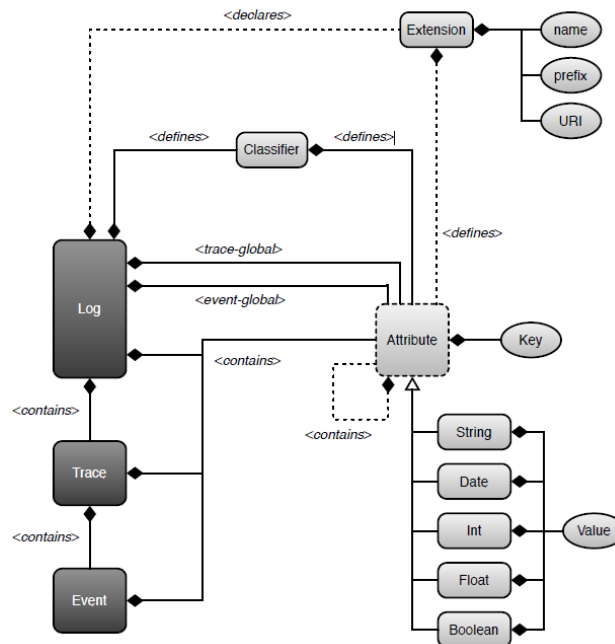
128. <https://standards.ieee.org/about/sasb/index.html>

129. Comme exemple de traces du processus "Préparer une tarte au chocolat", nous citons : "Préparer une tarte ronde au chocolat et à trois couches en suivant la méthode de Widad la pâtissière" et "Préparer une tarte carrée au chocolat".

130. <http://www.xes-standard.org/lifecycle.xesext>



7.24.a. Méta-modèle du langage MXML [van Dongen and Van der Aalst, 2005]



7.24.b. Méta-modèle du langage XES [Gunther and Verbeek, 2014]

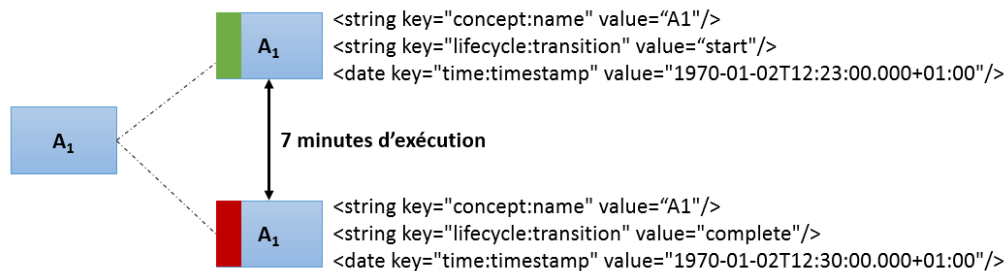
FIGURE 7.24 – Méta-modèles des deux langages compatibles avec les outils de fouille de processus.

activité A_1 -ou événement A_1 selon le méta-modèle XES- se voit, en effet, un ensemble de deux instances de la même activité où l'attribut de transition de la première instance est égale à "start" et celui de la deuxième est égale à "complete", comme le montre la Figure 7.25.a. Dans le présent travail de recherche, nous souhaitons simplifier cette représentation en ne considérant que la deuxième instance de l'activité dont la transition est "complete". Cette simplification est justifiée comme suit :

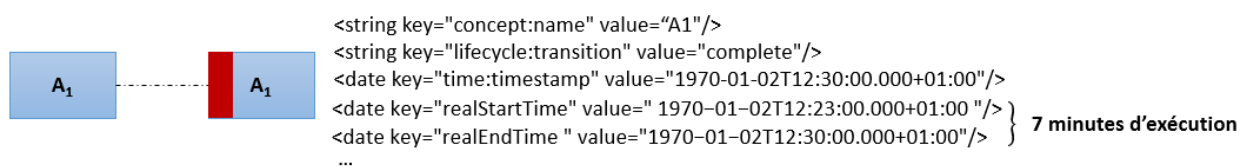
- Les algorithmes de l'apprentissage apprennent à partir d'un ensemble de traces **finies**, dont le résultat de l'exécution de leurs activités est bien **connu**,
- Le fait de considérer des traces qui sont encore en cours d'exécution **impacte négativement** le résultat de la fouille et de l'apprentissage (en étant considérées comme un bruit face aux traces complètes).

De ce fait, nous adoptons la représentation simplifiée de la Figure 7.25.b pour une activité A_1 . Notons que

la trace issue du module de traçabilité est conforme au méta-modèle de trace (Figure 6.15), ceci dit que tous les éléments de ce méta-modèle, qui sont relatifs à l'activité, doivent être présents dans le journal des événements (d'où la présence des "realStartTime" et "realEndTime" dans la Figure 7.25.b).



7.25.a. A₁ vue par le standard XES



7.25.b. Représentation de A₁ dans notre travail de recherche

FIGURE 7.25 – Représentation, d'une activité, adoptée par le standard XES et celle adoptée par notre travail de recherche qui ne prend en compte que les activités dont l'exécution est terminée.

Après avoir choisi le format du journal des événements et défini la forme de ses activités, nous revenons maintenant au premier verrou technologique que nous avons identifié au début de la section. L'ingénierie dirigée par les modèles (MDE) se voit un moyen efficace permettant de lever ce verrou, à travers les notions de transformation et de projection¹³¹. Comme illustré sur la Figure 7.26, une transformation ATL^{??} ainsi qu'une projection "XMI2XML" ont été développées, pour permettre le passage automatique d'un modèle de trace XMI à un modèle XES (journal des événements) en trois grandes étapes :

- Nous commençons d'abord par la transformation¹³¹ des traces XMI, étant conformes au méta-modèle de traces de la Figure 6.15, à un ensemble de traces XMI conformes au méta-modèle XES de la Figure 7.24.b. Notons que nous sommes toujours dans l'espace technique de modélisation MDA¹³² (Figure 7.27).
- Nous concaténons, ensuite, ces nouvelles traces XMI en un seul fichier XMI (GlobalXes.xmi sur la Figure 7.26), en étant toujours dans l'espace technique de modélisation.
- Pour générer le document "EventLog.xes", respectant la syntaxe XES, nous appliquons la projection "XMI2XML" au fichier GlobalXes.xmi. Cette projection permet le passage de l'espace technique de modélisation (modèle conforme à un méta-modèle, MDA¹³² sur la Figure 7.27), à l'espace technique XML (document conforme à une schéma XSD¹³³) comme le montre la Figure 7.27.

Par conséquent, pour construire le journal des événements qui contient toutes les anciennes traces complètes du processus exemple de la Figure 7.22, nous appliquons la conversion "XMI2XES", sur les 1572 traces XMI que nous avons construites à la main, en instanciant le méta-modèle de trace de la Figure 6.15, selon la logique présentée dans le Tableau 7.9. En effet :

131. La transformation permet le passage d'un modèle conforme à un certain méta-modèle "x", à un autre modèle conforme au même méta-modèle "x". En revanche, la projection permet de passer d'un modèle conforme à un certain méta-modèle "x", à un autre modèle conforme à un autre méta-modèle "y" [Bézivin, 2005].

132. Model Driven Architecture, c'est une approche de développement des systèmes logiciels, proposée par l'OMG (<https://www.omg.org/mda/>). La différence entre la MDA et la MDE est bien discutée ici : <https://modeling-languages.com/clarifying-concepts-mbe-vs-mde-vs-mdd-vs-mda/>

133. ou XML Schema Definition, il permet de représenter l'inter-relation entre les attributs et les éléments d'un objet XML (comme un document par exemple) : https://www.w3schools.com/xml/schema_intro.asp.

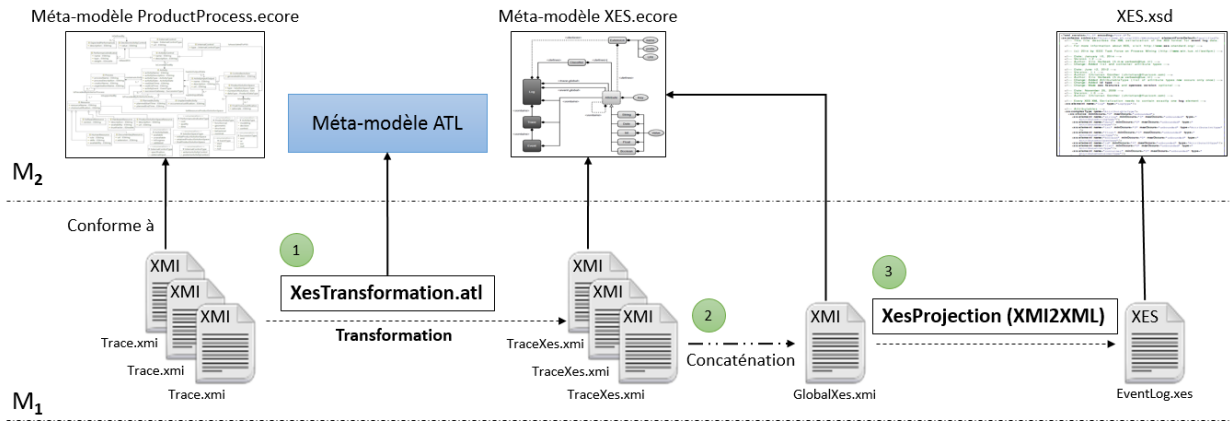


FIGURE 7.26 – Conversion XMIs2XES : Passage d'un ensemble de traces, de format XMI, à un seul journal des événements de format XES.

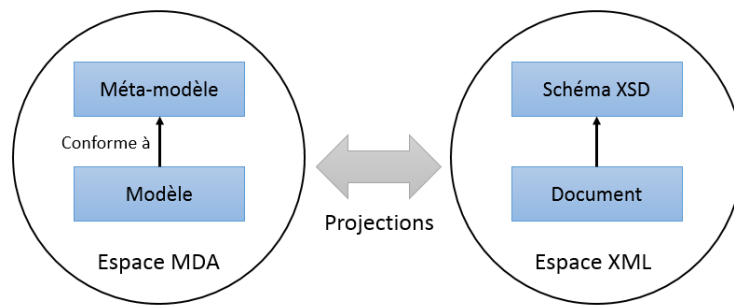


FIGURE 7.27 – Projections entre les deux espaces techniques MDA et XML (adapté de [Kurtev et al., 2002]).

- Les traces des patterns P_1, P_2, P_3 et P_4 sont contrôlées par la totalité des variables de contexte. V_1 prend 40 valeurs réelles possibles ($V_1 \in [23, 32]$ pour $P_{1 \rightarrow 3}$ et $V_1 \in [35, 44]$ pour P_4), RH prend deux valeurs possibles : *John* ou *Sally*, saison prend deux valeurs possibles : hiver ou été, V_2 prend les valeurs : normal ou anormal et V_3 prend les valeurs : favorable ou défavorable. En construisant les traces des patterns $P_{1 \rightarrow 4}$ de cette façon, nous obtenons 160 traces par patterns, soit 640 traces différentes pour les quatre patterns.
- Les traces des patterns P_5 et $P_{8 \rightarrow 50}$ ne sont contrôlées que par V_1 qui prend 10 valeurs réelles possibles ($V_1 \in [3, 12]$), RH qui prend les valeurs : *Ryan* et *Sally* et la saison qui ne peut être que l'automne. P_6 et P_7 sont instanciés pour 14 et 12 valeurs de V_1 respectivement ($V_1 \in [0, 12]$ pour P_6 et $V_1 \in [1, 12]$ pour P_7), en plus des valeurs de RH (*Ryan* et *Sally*) et saison (automne). Ceci donne lieu à 20 traces par pattern pour P_5 et $P_{8 \rightarrow 50}$, 24 traces pour P_7 et 28 traces pour P_6 . Au total, nous avons 932 traces différentes pour tous les patterns $P_{5 \rightarrow 50}$.

Pattern	Variables de contexte					Total des traces
	V_1	RH	Saison	V_2	V_3	
$P_{1 \rightarrow 4}$	40	2	2	1	1	$(40 \times 2 \times 2 \times 4) = 640$
P_5	10	2	1	-	-	$(10 \times 2) = 20$
P_6	14	2	1	-	-	$(14 \times 2) = 28$
P_7	12	2	1	-	-	$(12 \times 2) = 24$
$P_{8 \rightarrow 50}$	10	2	1	-	-	$(10 \times 2 \times 43) = 860$
1572						

TABLE 7.9 – Logique de construction des traces du processus exemple de la Figure 7.22

Ayant le journal des événements (contenant les 1572 traces¹³⁴) en format XES, il est maintenant possible de procéder à sa fouille en utilisant un des logiciels de fouille de processus existant dans le marché. Nous citons, entre autres, MyInvenio¹³⁵, ProM¹³⁶ et Disco¹³⁷. Dans le présent travail de recherche, nous faisons le choix d'utiliser l'outil ProM¹³⁶ car en plus d'être simple d'utilisation, il est gratuit, open source et permet l'hébergement¹³⁸ de nouveaux plugins de fouille de processus (au cas où aucun des plugins existants ne répond à nos besoins). De plus, ProM permet la découverte des modèles de processus ainsi que les différentes règles de décision et les différents patterns, existants dans un journal des événements, via la perspective "trace".

Le processus d'aide à la décision globale est à deux phases comme introduit au début de la Section 7.4.1 : la phase de l'exploration des patterns (étape 4 de la Figure 7.28) et la phase de l'identification des règles de décision (étapes 2 et 3 de la Figure 7.28). La deuxième phase est, comme le montre la figure, constituée de deux sous-phases [van der Aalst, 2011] : d'abord la phase de la "génération du modèle réseau de Pétri" à partir du journal des événements (étape 2 de la Figure 7.28); ce modèle décrit la simultanéité et la synchronisation des activités dans les traces et identifie les points de décision qui donnent lieu à d'autres choix. Ensuite, ayant le journal des événements ainsi que le modèle réseau de Petri correspondant, la sous-phase de la "fouille de décision" (Decision mining, (étape 3 de la Figure 7.28)) peut être exécutée pour identifier les règles de décision contrôlant le choix des patterns. Maintenant, ayant les patterns et les règles de décision, le module de l'aide à la décision procède au filtrage et propose à l'utilisateur les patterns qui satisfont les règles de décision et le nouveau contexte de décision. Les étapes 2, 3, 4 et 5 de la Figure 7.28 seront détaillées dans les sections suivantes. Notons que les étapes {2, 3} et {4} peuvent être exécutées en parallèle.

7.4.3 Phase d'exploration des différents patterns existants dans un journal des événements

Lors de l'exécution d'un processus métier, il est possible de considérer des patterns différents (comme ceux du Tableau 7.8), comme il est tout à fait possible de suivre plusieurs fois le même pattern (par exemple, toujours exécuter le pattern P_6 -si C_1 est vérifiée et ($V_1 > 13$)- au lieu de suivre les patterns P_5 ou $P_{7 \rightarrow 50}$ du Tableau 7.8). Cela se traduit par l'existence des traces similaires, en termes de nature et d'ordre d'activités, dans le journal des événements enregistré (la date de l'exécution ne peut bien évidemment être la même). L'exploration des patterns a donc pour objectif d'identifier les variantes de traces et leurs doublons, où chaque variante de trace fait référence à un pattern qui est une exécution de processus linéaire de son début à sa fin [Es-Soufi et al., 2016].

Différents travaux de recherche se sont intéressés à l'extraction de variantes de traces. Song et al. [2008] proposent une approche basée sur la partitionnement de traces (trace clustering), qui regroupe les traces similaires en sous-ensembles homogènes selon plusieurs perspectives. Bose and van der Aalst [2009] proposent un plugin ProM appelé "Pattern abstraction", permettant d'explorer les patterns d'exécution de bas niveau (séquences étant des sous-ensembles de patterns) dans un journal des événements. Ces patterns de bas niveau peuvent être fusionnés pour générer les patterns de processus les plus fréquents qui peuvent ensuite être exportés dans un seul fichier CSV. La perspective ProM appelée "Explore Event Log (Trace Variants/Searchable/Sortable)" permet d'explorer les patterns en mentionnant les différentes variantes de trace ainsi que le nombre et les noms des traces dupliquées. Ces variantes peuvent être exportées dans des

134. Ce nombre paraît tout à fait acceptable pour tester la proposition et la précision des algorithmes de fouille de processus choisis. Cependant, pour découvrir les processus réels avec un niveau supérieur de certitude, il faut considérer un nombre de trace beaucoup plus important (13087 traces sont considérées pour étudier le processus de demande de prêt : <https://data.4tu.nl/repository/uuid:3926db30-f712-4394-aebc-75976070e91f>).

135. <https://www.my-invenio.com/>

136. <http://www.promtools.org/>

137. <https://fluxicon.com/disco/>

138. En effet, les nouveaux plugins développés peuvent être intégrés à l'environnement ProM. Cette intégration permet de combiner les plugins développés avec les autres plugins proposés par ProM. En outre, elle permet de visualiser, graphiquement, les résultats des plugins en choisissant une des perspectives proposées par ProM.

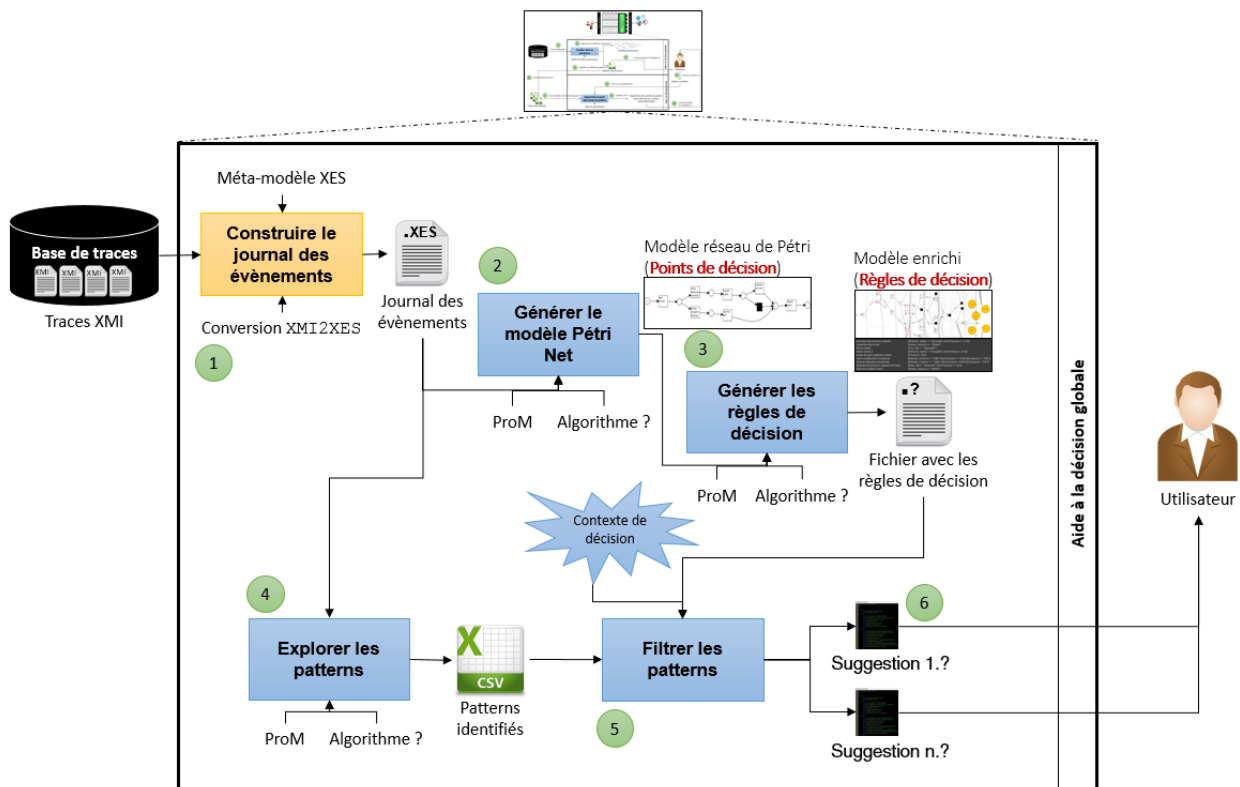


FIGURE 7.28 – Processus de l'aide à la décision globale. Les activités en jaune sont introduites pour lever les verrous identifiés.

fichiers CSV distincts, où chaque fichier contient la variante de trace, c'est-à-dire le pattern de processus, ainsi que les traces dupliquées associées. Ces travaux sont comparés selon les critères suivants :

- Déterminisme : nous souhaitons que les patterns identifiés par l'algorithme soient exactement les patterns existant dans le journal des événements. Nous exigeons ainsi une certitude de 100%.
- Capacité d'identifier **tous** les patterns différents,
- Capacité d'identifier les patterns dupliqués,
- Capacité d'identifier la fréquence de chaque pattern dans le journal des événements,
- Capacité d'exporter les patterns différents, ainsi que leur fréquence, dans un seul fichier.

Le résultat de la comparaison des différents travaux étudiés est présenté dans le Tableau 7.10.

En effet, nous ne voyons pas que l'approche de Song et al. [2008] est pertinente dans notre contexte car notre objectif est de découvrir les patterns d'une façon déterministe et non pas d'une façon approximative ou probabiliste. Le plugin "Pattern abstraction" ainsi que la perspective ProM "Explore Event Log (Trace Variants/Searchable/Sortable)" sont considérés comme pertinents dans notre contexte. De plus, étant donné qu'aucune des deux techniques que nous venons de garder ne permet d'exporter un fichier CSV contenant uniquement les variantes de trace et leur fréquence, nous choisissons d'utiliser la perspective ProM "Explore Event Log (Trace Variants/Searchable/Sortable)", car les patterns identifiés sont relatifs à l'exécution des processus du début à la fin, contrairement aux séquences étant une partie des patterns. De plus la perspective tient compte de l'ordre des activités contrairement au plugin "Pattern abstraction".

La perspective "Explore Event Log (Trace Variants/Searchable/Sortable)", identifie **correctement** tous les patterns (Tableau 7.8) de notre journal des événements relatif au processus illustratif de la Figure 7.22. Une partie des patterns identifiés est présentée dans la Figure 7.29.

	Approche de [Song et al., 2008]	Plugin "Pattern abstraction"	"Pattern perspective "Explore Event Log"
Déterminisme	-	✓	✓
Identification de tous les patterns différents	~	✓	✓
Identification des patterns dupliqués	-	✓	✓
Identification de la fréquence de chaque pattern	-	✓	✓
Export des patterns et leur fréquence dans un seul fichier	-	-	-

TABLE 7.10 – Résultat de comparaison des algorithmes de fouille de pattern étudié. ✓ (resp. -) veut dire "oui" (resp. "non") et ~ veut dire "pas vraiment".

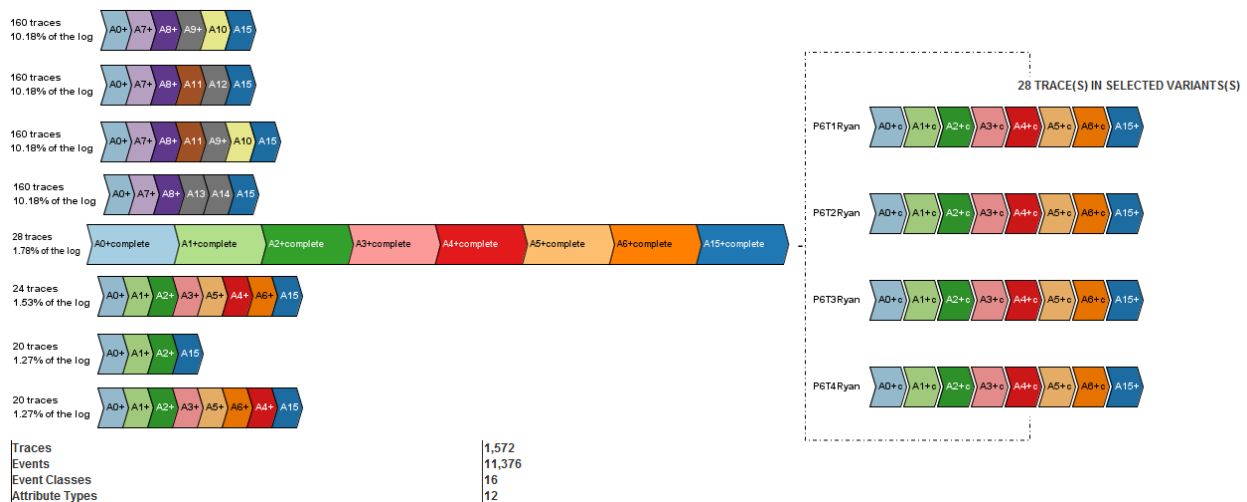


FIGURE 7.29 – Partie des patterns identifiés par la perspective ProM "Explore Event Log (Trace Variants/Searchable/Sortable)".

Bien qu'elle soit très pertinente, la perspective "Explore Event Log (Trace Variants/Searchable/Sortable)" ne permet pas d'exporter les variantes de patterns ainsi que leur fréquence dans un seul fichier. Ceci nous pousse à poser la question suivante :

Verrou technologique 2



Comment passer d'un ensemble de fichiers CSV, où chacun contient la variante du pattern et ses doublons, à **un seul** fichier CSV ne contenant que les variantes de **pattern** et leur **fréquence** ?

Le deuxième verrou technologique peut être levé en développant une application permettant le post-traitement des patterns identifiés (Figure 7.30) comme suit :

- Calculer la fréquence d'apparition de chaque pattern,
- Supprimer les doublons,
- Générer un fichier contenant les différents patterns non dupliqués, classés par ordre décroissant selon leur fréquence d'apparition.

Ce post-traitement est implémenté sous le nom "Patterns2RankedPatterns" en utilisant le langage de programmation Java. Les patterns post-traités sont maintenant prêts à être exploités et il ne reste qu'à découvrir les règles de décision contrôlant leur choix.

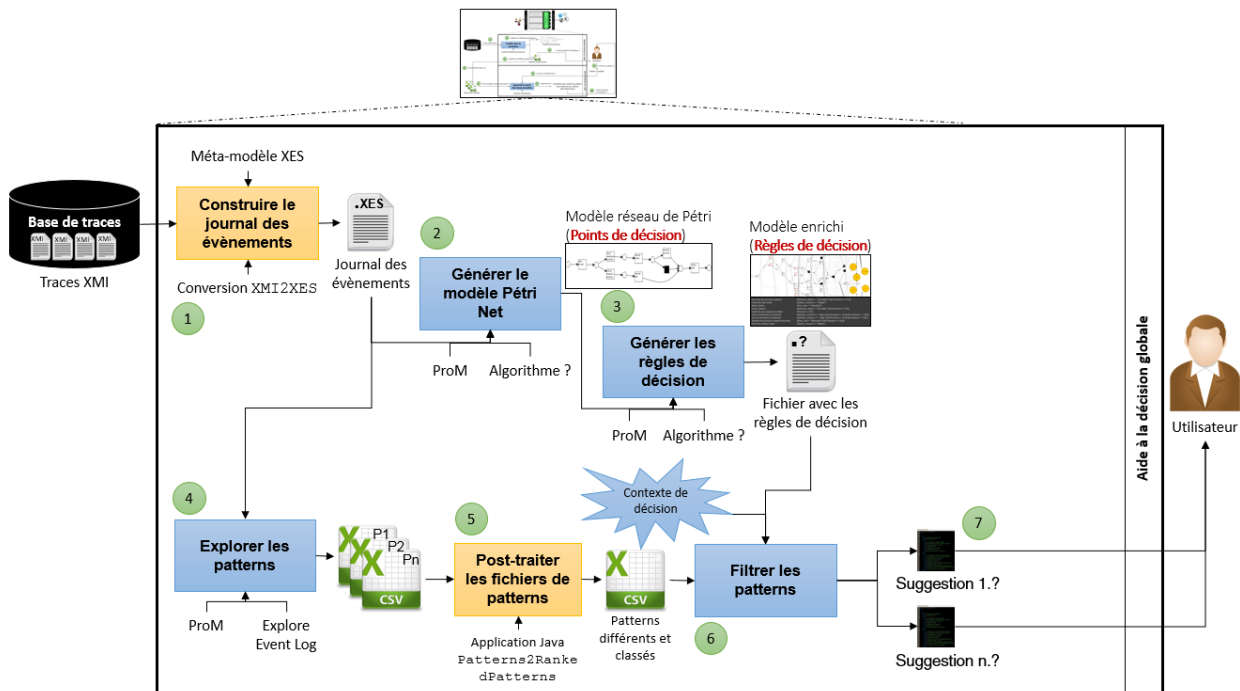


FIGURE 7.30 – Processus de l'aide à la décision globale amélioré. Les activités en jaune sont introduites pour lever les verrous identifiés.

7.4.4 Phase de la découverte des règles de décision

7.4.4.1 Sous-phase 1 : découverte du modèle réseau de Pétri à partir du journal des événements

Différents algorithmes ont été proposés pour générer automatiquement un modèle réseau de Pétri à partir d'un journal des événements. Nous citons par exemple l'algorithme "alpha", l'algorithme "alpha ++" (discutés et comparés par van Dongen et al. [2009]), le "inductive visual miner" [Leemans et al., 2014] et le "inductive miner" (appelé aussi B') [Leemans et al., 2013]. L'algorithme que nous choisirons doit produire un modèle réseau de Petri correct en identifiant les trois types de jonctions considérés : Xor, Or et And. De plus, le modèle produit doit être conforme¹³⁹ au journal des événements que nous avons construit à partir de notre processus de référence de la Figure 7.22.

La comparaison des différents algorithmes est présentée dans le Tableau 7.11. Ce dernier montre que seul le Inductive Miner (IM) répond à nos besoins. En effet, tous les algorithmes permettent la représentation des logiques "Xor" et "And" (le lecteur peut se référer à la Figure 7.16 pour faire la correspondance entre les concepts des réseaux de Petri et ceux relatifs aux logiques "Xor" et "And"), mais seul le IM peut représenter la logique "Or".

Pour expliciter ceci, nous prenons par exemple les modèles générés par les algorithmes alpha et alpha++¹⁴⁰ (Figure 7.31.a dont la traduction en BPMN est présentée dans la Figure 7.32.a), en suivant la logique des réseaux de Petri, le jeton permettant de franchir l'activité A_0 se divise en trois pour occuper les places en rouge dans la Figure 7.31.a. Ceci déclenche l'exécution parallèle des activités A_1 , A_3 et A_5 (si nous

139. La conformité entre un modèle et un journal des événements est discutée dans Section 7.3.4.2.

140. Qui est introduit pour lever les limites de l'algorithme alpha.

	Alpha	Alpha ++	IVM	IM
And	✓	✓	✓	✓
Xor	✓	✓	✓	✓
Or	-	-	-	✓
Conformité	-	-	-	✓

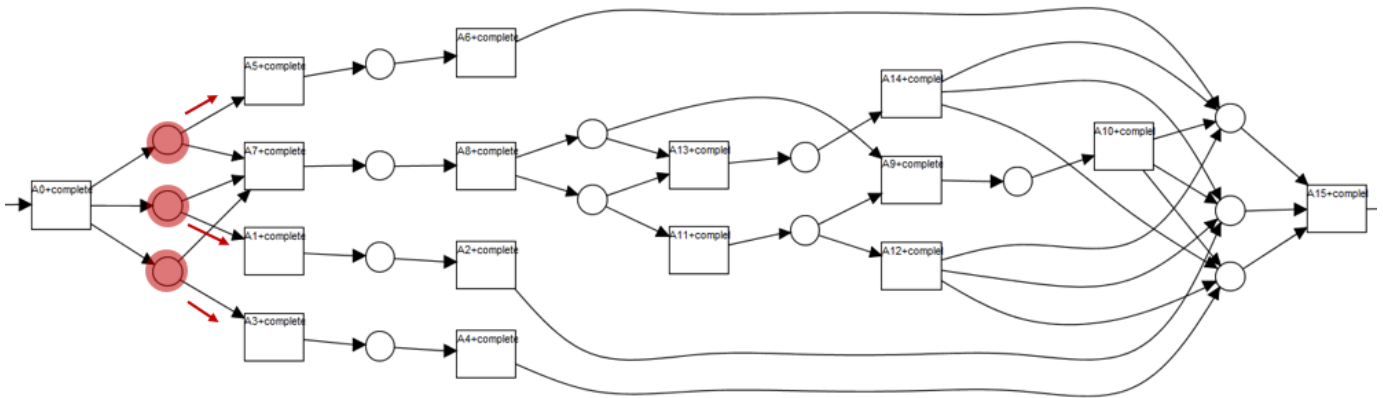
TABLE 7.11 – Résultat de la comparaison des algorithmes permettant la génération automatique du modèle réseau de Petri à partir de notre journal des événements

supposons que c'est la règle "C₁" de la Figure 7.22 qui est vérifiée); ce parallélisme n'est par contre vérifié que si $V_1 < 13$.

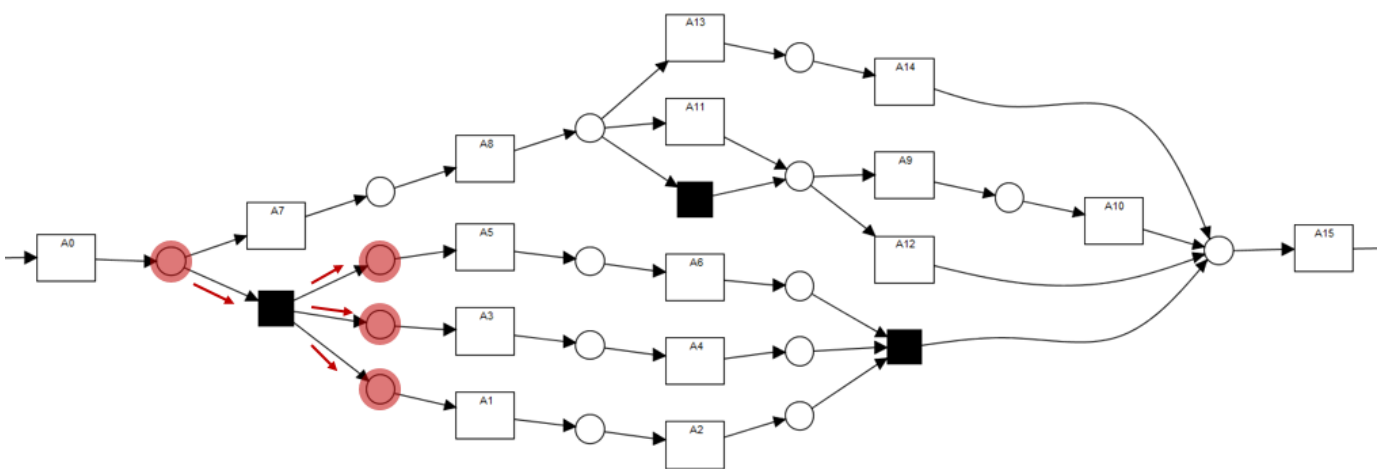
De même, si nous prenons l'exemple du modèle généré par le IVM (Figure 7.31.b dont la traduction en BPMN est présentée dans la Figure 7.32.b), le jeton issu de A_0 passe par une activité dite invisible¹⁴¹ (représentée par un carré noir dans la Figure 7.31.b) et se divise ensuite en trois pour activer l'exécution parallèle des activités A_1 , A_3 et A_5 . Passons maintenant au modèle généré par le IM (Figure 7.31.c dont la traduction en BPMN est présentée dans la Figure 7.32.c), il est possible de décrire ici la logique "Or" grâce à la présence des activités invisibles¹⁴¹ qui sont exclusives avec A_3 et A_5 . En effet, le jeton issu de A_0 franchit l'activité invisible¹⁴¹ (nous supposons que "C₁" est toujours vérifiée), et se divise en trois jetons. Un des trois jetons franchit directement l'activité A_1 alors que les deux restant doivent occuper une des deux activités (A_5 ou l'activité invisible¹⁴¹ pour le premier jeton et A_3 ou l'activité invisible¹⁴¹ pour le deuxième), en franchissant les deux activités invisibles¹⁴¹, seule la suite des activités A_1 et A_2 est exécutée, décrivant ainsi la logique "Or" de la Figure 7.22 à travers l'exécution du pattern P_5 (Tableau 7.8).

Ayant maintenant le modèle réseau de Petri généré par l'algorithme Inductive Miner, étant le plus pertinent dans notre contexte, nous passons à la section suivante où nous explorant les techniques permettant l'identification des règles de décision.

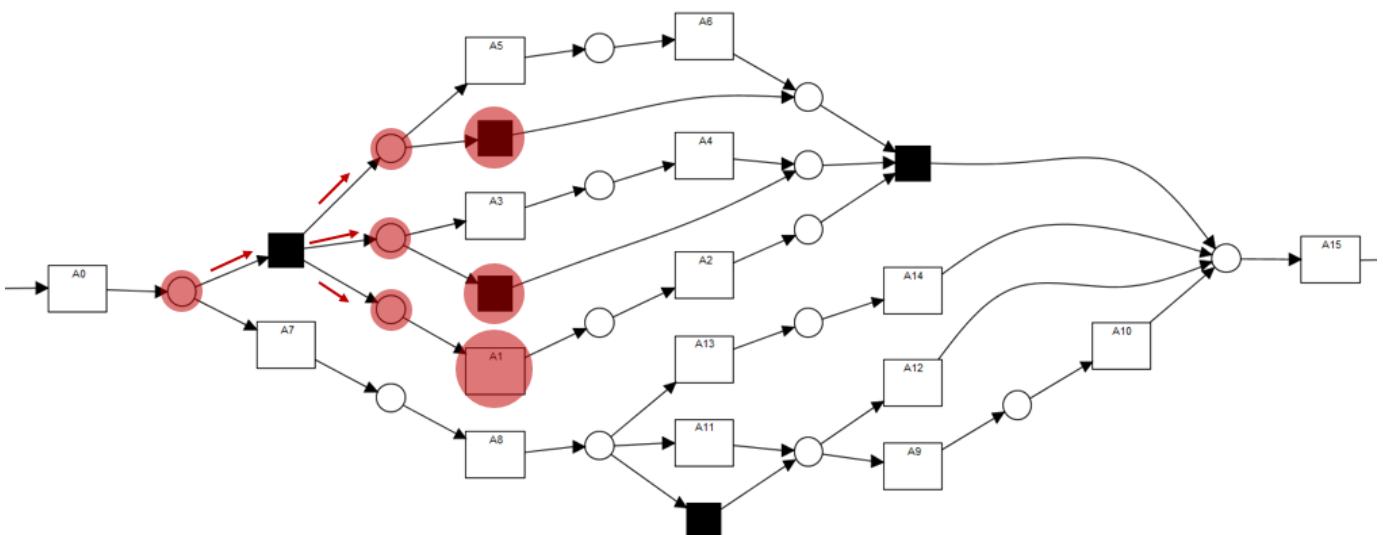
¹⁴¹. Appelée aussi *Tau split*, c'est une activité fantôme qui permet d'exprimer la logique "Or" tout en permettant au modèle réseau de Petri d'être conforme à son méta-modèle.



7.31.a. Modèle généré par l'algorithme alpha (l'algorithme alpha++ génère le même modèle)

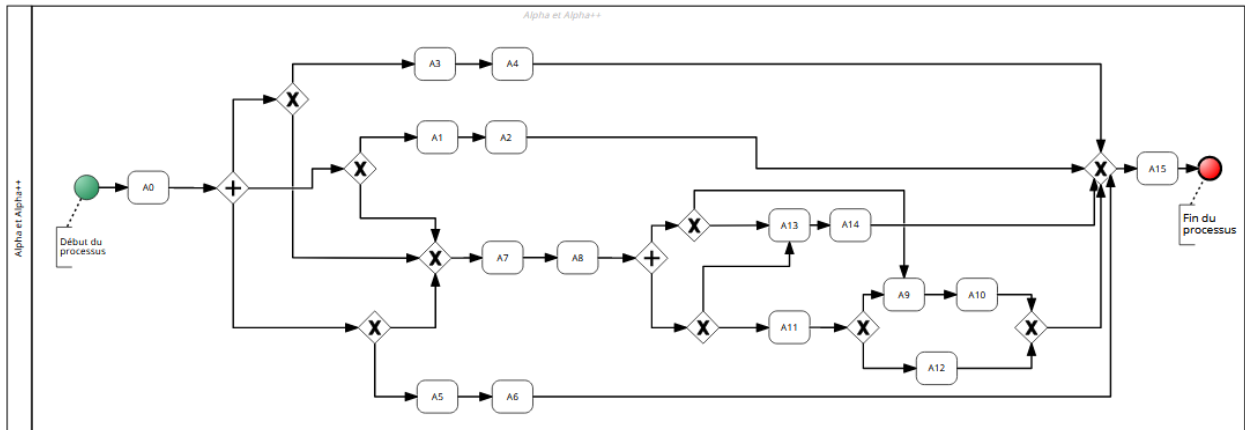


7.31.b. Modèle généré par le Inductive Visual Miner (IVM)

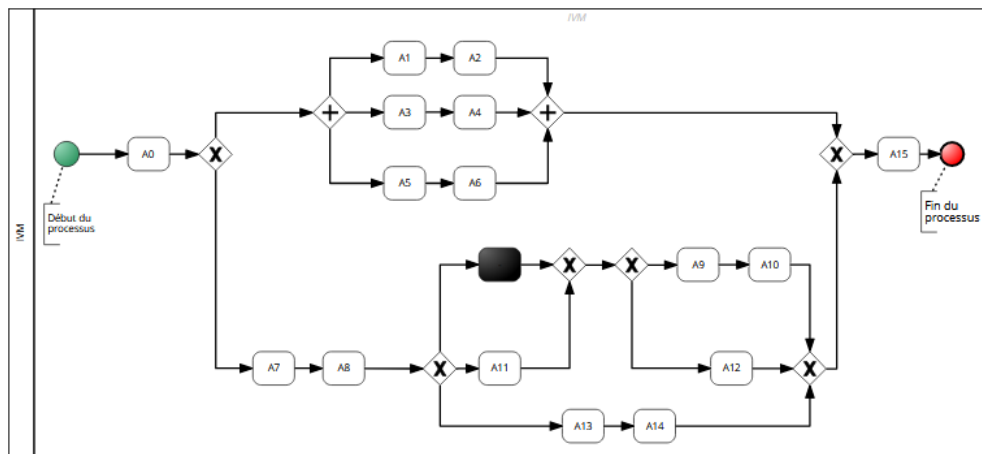


7.31.c. Modèle généré par le Inductive Miner (IM)

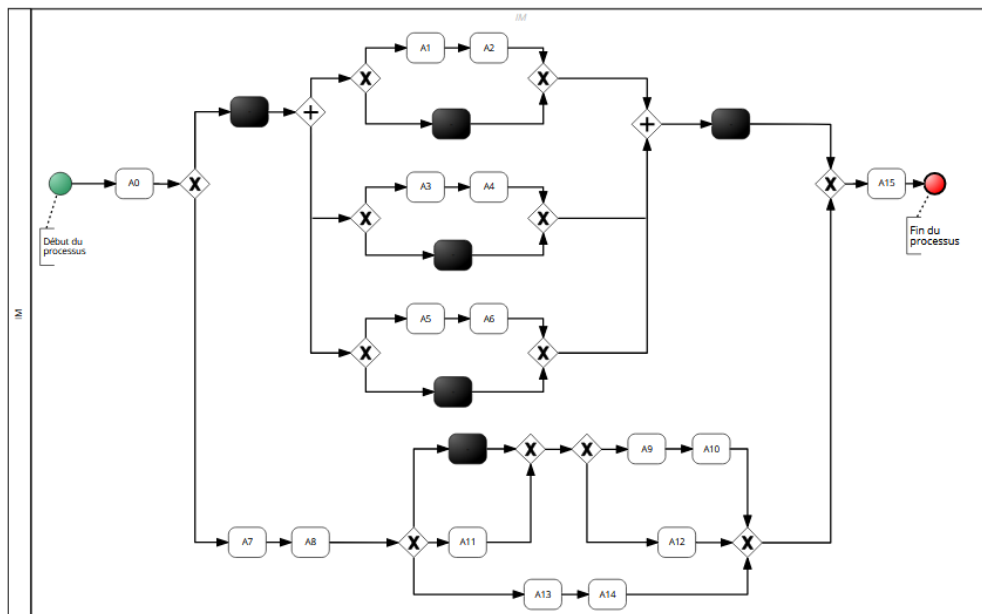
FIGURE 7.31 – Génération du modèle réseau de Petri par les algorithmes Alpha (Alpha++), IVM et IM



7.32.a. Représentation BPMN du modèle généré par les algorithmes alpha et alpha++ (présenté dans la Figure 7.31.a)



7.32.b. Représentation BPMN du modèle généré par le Inductive Visual Miner (présenté dans la Figure 7.31.b)



7.32.c. Représentation BPMN du modèle généré par le Inductive Miner (présenté dans la Figure 7.31.c)

FIGURE 7.32 – Traduction des modèles générés, par les algorithmes Alpha (Alpha++), IVM et IM, en modèles BPMN.

7.4.4.2 Sous-phase 2 : fouille des règles de décision (Decision Mining)

De nombreux travaux de recherche s'intéressent à la fouille de décision, branche de la fouille des processus, qui concerne l'identification des règles de décision. Rozinat and Aalst [2006] proposent un algorithme, appelé "Decision Point Analysis", qui permet d'identifier les points de décision représentant les divisions de choix dans un processus. Ensuite, au niveau de chaque point de décision, est identifiée la règle de décision exclusive (règle Xor) sous la forme " $v \text{ op } c$ ", où " v " est une variable, " op " est un opérateur de comparaison et " c " est une constante; cette règle permet à une activité parmi les autres de s'exécuter. L'algorithme "Decision Point Analysis" est implémentée en tant que plugin et est hébergé par ProM.

De Leoni et al. [2013] proposent une technique qui améliore l'algorithme "Decision Point Analysis" en permettant de découvrir des règles de décision beaucoup plus complexes pour les points de décision de type Xor. La technique proposée est basée sur la découverte d'invariants¹⁴² qui prend en compte plusieurs variables, c'est-à-dire sous la forme " $v_1 \text{ op } c$ " ou " $v_1 \text{ op } v_2$ ", où " v_1 " et " v_2 " sont des variables. Cette technique est implémentée sous la forme d'un outil nommé "Branch Miner"¹⁴³.

Catalkaya et al. [2014] proposent une technique qui intègre les règles de décision dans les modèles de processus en transformant la jonction Xor en une jonction Xor basée sur des règles qui déterminent automatiquement la solution optimale en termes de performances (coût et temps entre autres) pendant l'exécution. Cette technique n'est par contre pas encore implémentée.

Bazhenova and Weske [2016] proposent une approche pour dériver des modèles de décision à partir de modèles de processus en utilisant la fouille de décision améliorée. En effet, les règles de décision sont découvertes à l'aide de l'algorithme "Decision Point Analysis" [Rozinat and Aalst, 2006], puis améliorées en prenant en compte les prédictions de mesures de performance de processus (temps, entre autre) liées aux différents résultats de décision. Cette approche n'est pas encore implémentée.

Dunkl et al. [2015] proposent une méthode qui étend le "Decision Point Analysis" qui permet d'analyser uniquement des valeurs uniques [Rozinat and Aalst, 2006]. La méthode proposée prend en compte les séries de données temporelles (c'est-à-dire la séquence des points de données listées dans l'ordre chronologique) et permet de générer des règles de décision complexes avec plusieurs variables. La méthode est implémentée mais n'est pas partagée publiquement.

Ghattas et al. [2014] proposent une technique d'exploration de processus qui permet d'identifier le chemin de processus le plus performant en extrayant des règles de décision basées sur les relations entre le contexte passé (c'est-à-dire la situation dans laquelle les décisions antérieures ont été prises), les décisions qui ont été prises au cours de l'exécution du pattern et la performance de processus (temps, coût, qualité). L'approche n'est pas encore implémentée.

De Leoni and van der Aalst [2013] introduisent une technique qui prend en entrée le modèle réseau de Petri du processus, le journal des événements et le résultat de l'alignement (indiquant si le modèle et le journal sont conformes, comme discuté dans la Section 7.3.4.2), et produit un modèle réseau de Petri avec les règles de décision inclusives et exclusives identifiées. Cette technique est implémentée en tant que plugin ProM appelé "Data Flow Discovery". Une autre variante de ce plugin, nécessitant uniquement le journal des événements et le modèle réseau Petri correspondant, est également implémentée.

Mannhardt et al. [2016] proposent une technique qui vise à découvrir des règles de décision inclusives et exclusives même si elles se chevauchent à cause de la présence des traces incomplètes. Cette technique est implémentée dans le plugin Multi-Perspective Explorer [Mannhardt et al., 2015] hébergé par ProM.

Sarno et al. [2013] proposent une approche pour explorer les règles de décision inclusives à l'aide de l'algorithme "Decision Point Analysis" [Rozinat and Aalst, 2006]. L'approche consiste à modifier manuellement le modèle de réseau de Petri en transformant la jonction OR en une jonction AND suivie d'une jonction XOR dans chacun de ses arcs sortants.

Avant de comparer les travaux qui nous semblent pertinents dans notre contexte de recherche, nous écartons les propositions de Bazhenova and Weske [2016], de Catalkaya et al. [2014] et de Ghattas et al. [2014] parce qu'elles ne sont pas encore implémentées et donc nous ne sommes pas capables de les tester pratiquement (la

142. Ce sont les propriétés ou les règles qui ne changent pas dans un journal des événements.

143. <http://sep.cs.ut.ee/Main/BranchMiner>

théorie est essentielle, certes, mais nécessite d'être testée pour être validée). Nous écartons aussi la méthode de [Dunkl et al. \[2015\]](#) car malgré le fait qu'elle soit implémentée, elle n'est pas partagée publiquement et donc non testée dans notre contexte. L'approche de [Sarno et al. \[2013\]](#) ne nous semble pas pratique dans notre contexte parce qu'elle nécessite une intervention humaine pour convertir les jonctions OR en XOR dans le modèle réseau de Petri généré, avant de passer à la découverte des règles de décision. De plus, le "Decision Point Analysis" de [Rozinat and Aalst \[2006\]](#) et le "Branch Miner" de [De Leoni et al. \[2013\]](#) ne calculent que les règles de décisions relatives à la jonction XOR. Par conséquent, il nous reste donc à comparer le "Data Flow Discovery" [[De Leoni and van der Aalst, 2013](#)] et le "Multi-Perspective Explorer" [[Mannhardt et al., 2015](#)] qui nous paraissent pertinents dans notre contexte. La comparaison se fait selon nos critères que nous définissons comme suit :

- Capacité de trouver les règles relatives aux points de décision (ou jonctions) AND, OR et XOR. En effet, malgré le fait de choisir le *Inductive Miner* qui permet d'identifier toutes ces jonctions lors de la génération du modèle réseau de Pétri, nous souhaitons quand même vérifier que l'algorithme, générant les règles de décision, arrive à correctement identifier les règles au niveau de chaque jonction.
- Capacité de découvrir automatiquement les règles de décision, sans intervention humaine.
- Précision : nous souhaitons trouver l'algorithme qui génère les règles de décision d'une façon précise (c-à-d, exactement comme nous les avons présentées en rouge dans la Figure 7.22). Nous avons introduit ce critère parce parfois, pour deux patterns exclusifs par exemple, l'algorithme ne découvre que la règle R_1 relative au premier pattern, et considère que la règle contrôlant l'autre chemin exclusif est \bar{R}_1 , sans refaire le calcul.

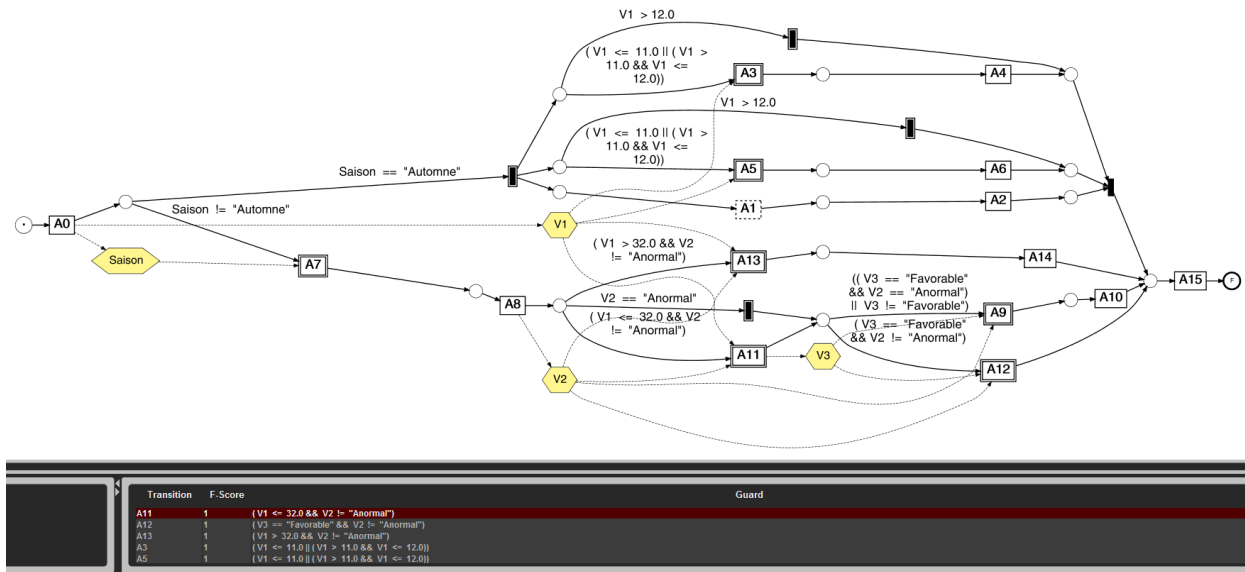
Le résultat de la comparaison des travaux étudiés est présenté dans le Tableau 7.12. Les deux algorithmes, "Data Flow Discovery" (DFD) [[De Leoni and van der Aalst, 2013](#)] et Multi-Perspective Explorer (MPE) [[Mannhardt et al., 2015](#)], permettent de générer les règles décision au niveau des trois types de jonctions, mais sont en revanche **imprécis**.

	Data Flow Discovery [De Leoni and van der Aalst, 2013]	Multi-Perspective Explorer [Mannhardt et al., 2015]
AND	✓	✓
XOR	✓	✓
OR	✓	✓
Précision	-	-

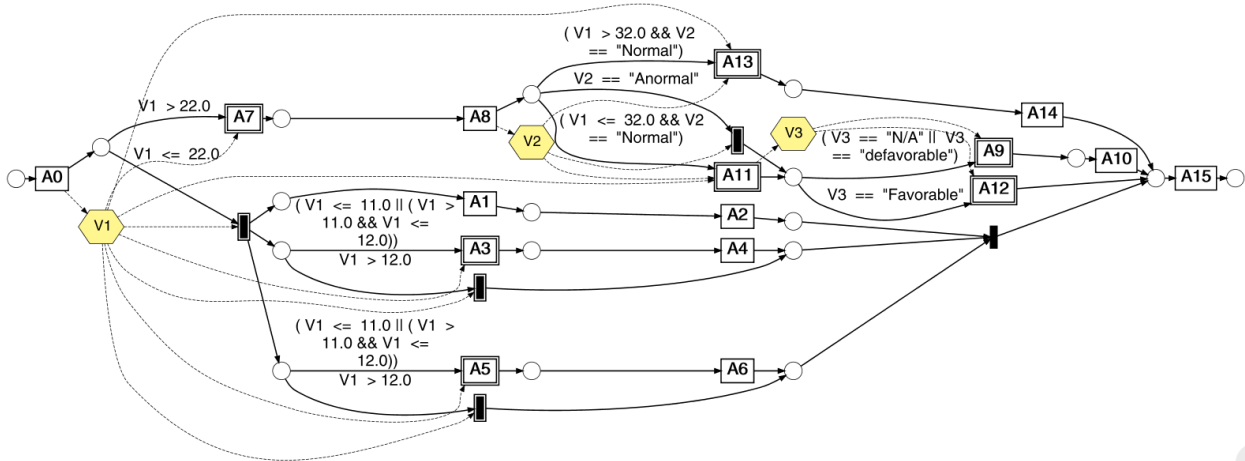
TABLE 7.12 – Résultat de la comparaison des deux algorithmes de fouille de décision que nous avons gardés.

En effet, après avoir appliqué les deux algorithmes sur le journal des événements que nous avons construit ainsi que le modèle réseau de Petri correspondant (généré en utilisant le *Inductive Miner*) nous remarquons que les règles de décision découvertes (Figure 7.33.a pour le DFD et Figure 7.33.b pour le MPE) ne correspondent pas parfaitement à celles de la Figure 7.22. En effet :

- Selon le DFD, la règle qui contrôle le choix des patterns $P_{1 \rightarrow 4}$ est ($Saison \neq Automne$). Cette règle est bien différente de celle de la Figure 7.22 ($C_2 = ((V_1 > 22) \ \&\& \ (RH \neq Ryan) \ \&\& \ (Saison == (hiver \ || \ ete))))$) parce que, premièrement, les conditions sur les valeurs de V_1 ne sont pas identifiées à ce niveau mais plutôt un peu loin dans le processus, et donc il suffit que la saison soit égale à l'automne pour proposer l'activité A_1 , cette dernière qui ne doit être proposée que si en plus ($V_1 \leq 22$). Deuxièmement, aucune règle sur les ressources humaines n'est identifiée. Troisièmement, la règle portant sur la variable de contexte "Saison" n'est pas juste, car pour proposer les patterns $P_{1 \rightarrow 4}$, il faut que ($Saison == (hiver \ || \ ete)$), ces patterns ne doivent donc pas être proposés pour une ($Saison == printemps$).
- Selon le MPE, la valeur de V_1 est bien prise en compte au début du processus comme noté sur la Figure 7.22. Par contre, il n'est pas précis car aucune règle sur la saison ou les ressources humaine n'est identifiée.



7.33.a. Modèle réseau de Petri enrichi avec les règles de décision, généré en utilisant l’algorithme "Data Flow Discovery".



7.33.b. Modèle réseau de Petri enrichi avec les règles de décision, généré en utilisant le Multi-Perspective Explorer.

FIGURE 7.33 – Génération des règles de décision en utilisant les deux algorithmes DFD et MPE.

Cette imprécision des deux algorithmes, au niveau de la génération des règles de décision, nous amène à la question suivante :



Verrou scientifique 2

Comment avoir des règles de décision **justes** et **précises** en utilisant un des deux algorithmes DFD et MPE, **sans modifier** les propriétés (type et nom, entre autres) des **variables** dans le journal des évènements ?

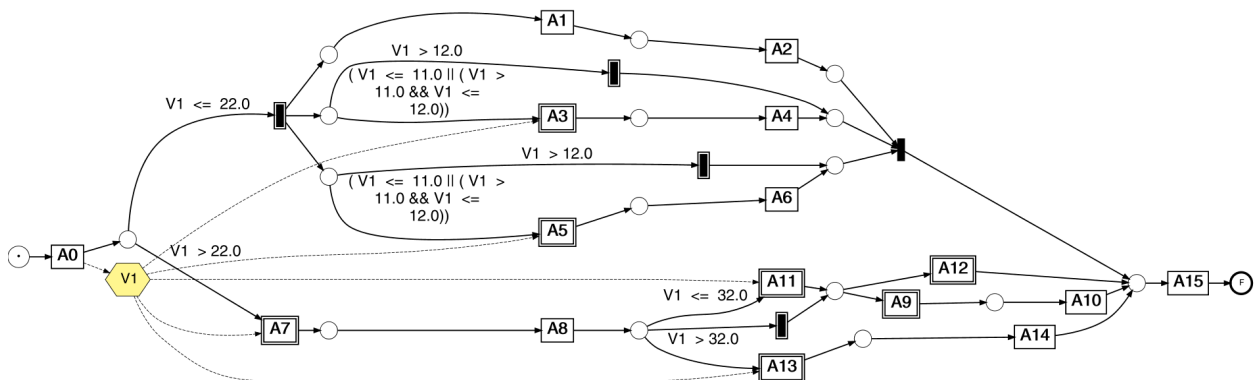
En effet, nous ne souhaitons pas lever ce deuxième verrou scientifique en modifiant le journal des évènements pour trois raisons :

- La modification des journaux des évènements, relatifs aux processus industriels complexes, sera très couteuse en temps et en argent,
- Les journaux des évènements modifiés peuvent ne plus refléter fidèlement le comportement réels des processus,
- Il est difficile d’implémenter les modifications pour les automatiser, car les journaux des évènements diffèrent d’un processus à un autre.

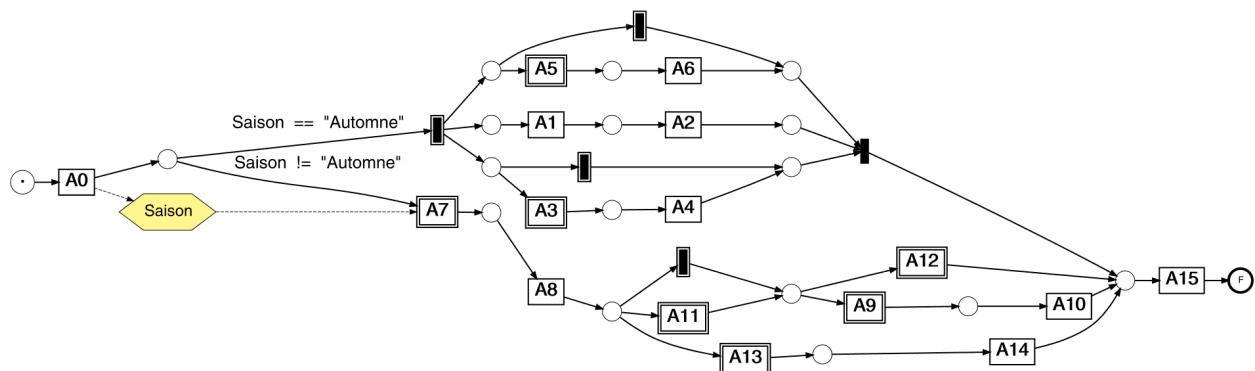
La solution qui nous paraît pertinente, concerne la génération des règles de décision autant de fois que de variable de contexte, en considérant chaque fois une seule variable pour éviter toute interférence entre les variables de contexte. Pour n variables de contexte, par exemple, nous aurons n modèles réseau de Petri, dont chacun est enrichi par la règle de décision relative à la variable de contexte considérée. Ces modèles sont ensuite post-traités pour avoir un seul modèle réseau de Petri où toutes les règles de décision sont présentes. Ce post-traitement, étant implémenté en Java sous le nom `Pnm1s2Pnm1`, consiste généralement à :

- parcourir tous les fichiers PNML (ce sont des fichiers où le modèle réseau de Petri est représenté textuellement en se basant sur la syntaxe XML) et récupérer les règles et les activités que contrôlent ces règles,
- Agréger ces règles au bon endroit en étudiant le lien logique approprié ("&&" ou "||", entre autres.) et générer le modèle PNML contenant toutes les règles de décision.

Pour tester l'efficacité de notre solution, permettant de lever le deuxième verrou, nous l'appliquons sur notre journal des événements et le modèle réseau de Petri correspondant en utilisant les deux algorithmes que nous avons gardés, le DFD et le MPE. En adoptant notre solution, le DFD arrive à identifier la règle relative à la variable V_1 au bon endroit (Figure 7.34.a). Cependant, la règle de décision relative à la variable *Saison* n'est toujours pas améliorée (Figure 7.34.b). L'algorithme DFD n'est donc pas précis, et ceci est une raison pertinente pour le rejeter. Passons maintenant à l'algorithme MPE, nous avons déjà montré que la règle de décision sur la variable V_1 était bien identifiée d'une façon précise et au bon endroit (Figure 7.33.b). En adoptant notre solution, les règles relatives aux variables *Saison* (Figure 7.35.a) et *RH* (Figure 7.35.b) sont désormais identifiées d'une façon précise, comme noté sur la Figure 7.22, lorsqu'elles sont identifiées séparément.

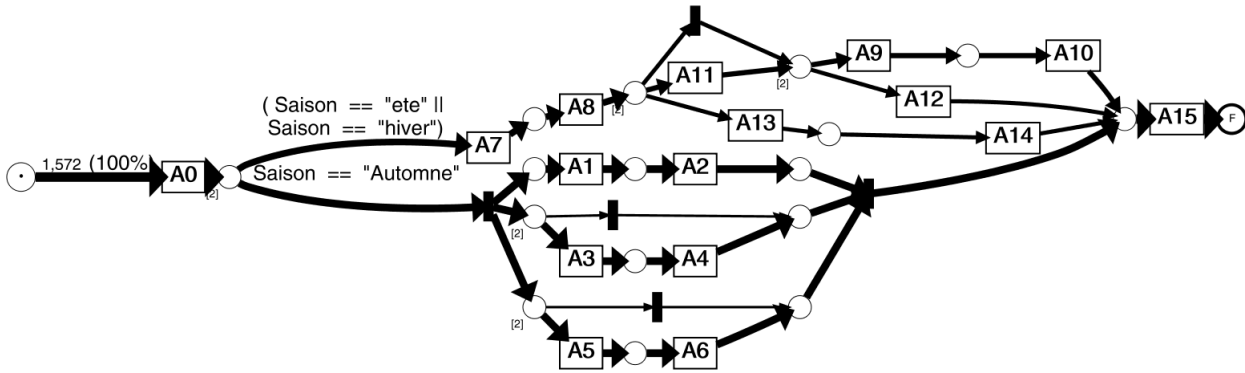


7.34.a. Règle de décision générée pour la variable de contexte V_1 , en utilisant le DFD

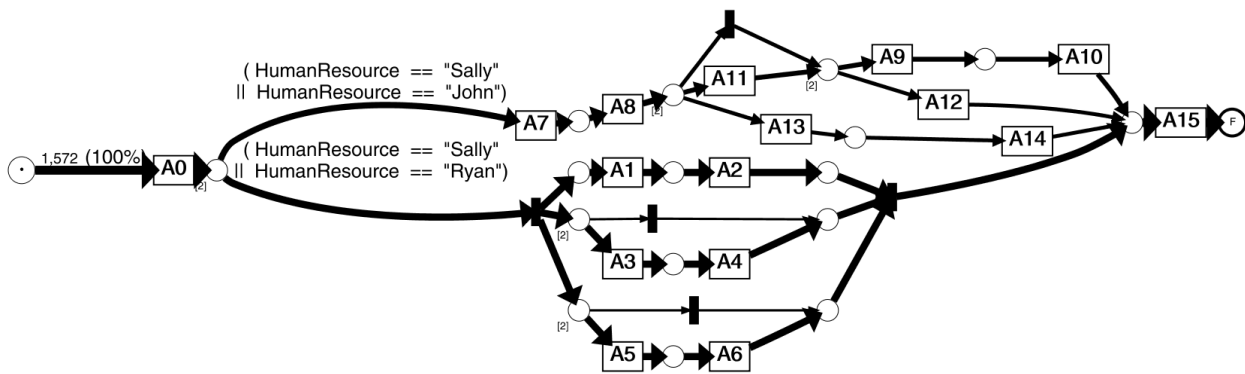


7.34.b. Règle de décision générée pour la variable de contexte *Saison*, en utilisant le DFD

FIGURE 7.34 – Génération des règles de décision relatives aux variables V_1 et *Saison* en utilisant le DFD.



7.35.a. Règle de décision générée pour la variable de contexte *Saison*, en utilisant le MPE



7.35.b. Règle de décision générée pour la variable de contexte *RH*, en utilisant le MPE

FIGURE 7.35 – Génération des règles de décision relatives aux variables *RH* et *Saison* en utilisant le MPE.

La Figure 7.36 représente le modèle réseau de Pétri final contenant toutes les règles de décision agrégées comme le préconise notre solution permettant de lever le troisième verrou.

Transition	Guard
A11	((V1<=32.0)&&(V2=="Normal"))
A12	(V3=="Favorable")
A13	((V1>32.0)&&(V2=="Normal"))
A3	((V1<=11.0) ((V1>11.0)&&(V1<=12.0)))
A5	((V1<=11.0) ((V1>11.0)&&(V1<=12.0)))
A7	(((((HumanResource=="Sally") ((HumanResource=="John"))&&((Saison=="ete") ((Saison=="hiver"))))&&(V1>22.0)))
A9	((V3=="N/A") ((V3=="defavorable")))
tau from tree	((V1<=32.0)&&(V2=="Anormal"))
tau from tree	((V1<=32.0)&&(V2=="Anormal"))
tau from tree	((V1<=32.0)&&(V2=="Anormal"))
tau split	(((((HumanResource=="Sally") ((HumanResource=="Ryan"))&&(Saison=="Automne"))&&(V1<=22.0)))

FIGURE 7.36 – Les règles de décision du modèle réseau de Petri résultant après l'agrégation des règles de décision identifiées séparément par le MPE.

Les règles de décision sont justes, précises et bien placées au niveau de toutes les jonctions, cependant, la présence des activités invisibles ¹⁴¹ (activités en noir) pose un problème qui malgré le fait qu'il ne soit pas très important lorsque nous visualisons le modèle réseau de Petri enrichi de la Figure 7.36, il est à résoudre pour permettre un filtrage correct pour ainsi proposer les patterns à l'utilisateur avec un niveau de certitude qu'il juge bon. En effet, pour expliquer ce problème, nous considérons le modèle réseau de Petri de la Figure 7.37 et nous nous concentrons sur la jonction AND contrôlée par la règle C₁. Nous avons expliqué que pour décrire la logique de la jonction OR, le Inductive Miner crée une activité dite invisible ¹⁴¹ (appelée aussi *Tau split*) représentée par un carré noir sur la Figure 7.37, et que les deux chemins qui viennent après

l'activité A_0 , sont exclusifs, ceci dit que l'activité A_7 est contrôlée par C_2 et l'activité invisible ¹⁴¹ (qui se situe juste en dessus de A_7) est contrôlée par C_1 . Ceci nous pose un problème parce que cette activité invisible ¹⁴¹ n'existe physiquement pas parmi les activités des patterns identifiés, la règle C_1 est ainsi perdue lors du filtrage, et les activités A_5 , A_3 et A_1 seront vues comme automatiquement exécutables vu qu'elles ne sont contrôlées par aucune règle de décision. Ceci n'est pas correct parce qu'il aura fallu vérifier C_1 avant d'exécuter A_5 , A_3 et A_1 comme noté sur la Figure 7.22. Nous identifions ainsi un nouveau verrou :



Verrou technologique 3

Comment **transmettre automatiquement** les **règles** de décision situées au niveau des **activités invisibles**, aux **activités postérieures** qui y sont connectées ?

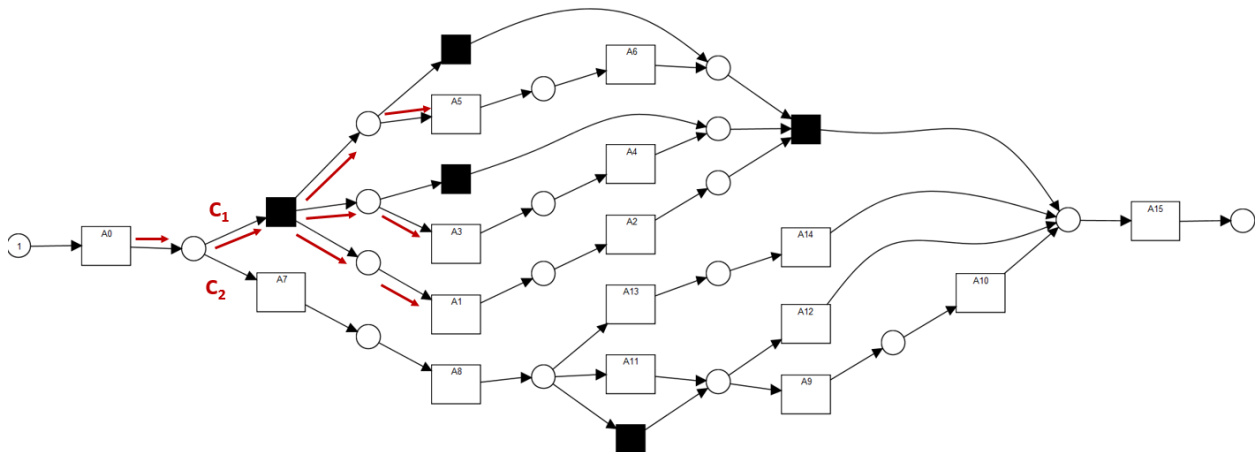


FIGURE 7.37 – Nécessité de propager les règles de décision situées au niveau des activités invisibles aux activités postérieures voisines.

Pour lever ce verrou, nous proposons un programme appelée `PnmlRulesPropagation`, implémentée en Java, permettant de :

- Lire le fichier Pnml d'entrée et créer un fichier pnml de sortie,
- Identifier les activités invisibles et stocker leurs règles de décision,
- Identifier les activités postérieures qui sont connectées aux activités invisibles,
- Créer le tag relatif aux règles de décision et copier les règles dedans.

En appliquant ce programme sur le fichier pnml relatif au modèle réseau de Petri de la Figure 7.36, nous obtenons un nouveau modèle réseau de Petri (Figure 7.38), où les règles de décision sont correctement propagées et sont ainsi identiques à celles de la Figure 7.22.

Étant sûrs que nos solutions ont permis de lever les verrous identifiés, et qu'elles permettent ainsi de garantir l'identification des règles de décision justes et précises, nous mettons à jour le processus d'aide à la décision globale en ajoutant une boucle, se répétant autant de fois que de variable de contexte, sur l'activité de la génération des règles de décision et une activité de post-traitement, permettant d'agrèger les règles de décision séparément identifiées et de propager les règles de décision relatives aux activités invisibles (Figure 7.39). Notons que l'activité A_1 est maintenant contrôlée par la condition C_1 .

Ayant maintenant les différents patterns identifiés ainsi que les règles de décision contrôlant leur choix, il est possible d'assister l'utilisateur lors d'une prochaine exécution du processus. En effet, le fichier des différents patterns classés est automatiquement filtré (étape 8 de la Figure 7.39) pour satisfaire les règles de décision identifiées **ainsi que** le contexte de l'utilisateur (son nom, la date de l'exécution, type de processus, etc.). En d'autres termes, les patterns qui ne satisfont pas les règles de décision et ceux qui, par exemple, sont exécutés par un utilisateur autre que celui qui exécute actuellement le processus, sont désactivés sans être

Transition	Guard
A1	(((HumanResource==" Sally"))((HumanResource=="Ryan"))&&(Saison=="Automne"))&&(V1<=22.0))
A11	((V1<=32.0)&&(V2=="Normal"))
A12	(V3=="Favorable")
A13	((V1>32.0)&&(V2=="Normal"))
A3	((V1<=11.0) ((V1>11.0)&&(V1<=12.0)))
A5	((V1<=11.0) ((V1>11.0)&&(V1<=12.0)))
A7	(((HumanResource==" Sally"))((HumanResource=="John"))&&((Saison=="ete") ((Saison=="hiver"))))&&(V1>22.0))
A9	((V3=="N/A") ((V3=="defavorable"))
tau from tree	((V1<=32.0)&&(V2=="Anormal"))
tau from tree	((V1<=32.0)&&(V2=="Anormal"))
tau from tree	((V1<=32.0)&&(V2=="Anormal"))
tau split	(((HumanResource==" Sally"))((HumanResource=="Ryan"))&&(Saison=="Automne"))&&(V1<=22.0))

FIGURE 7.38 – Les règles de décision du modèle réseau de Petri résultant après la propagation des règles de décisions à travers les jonctions.

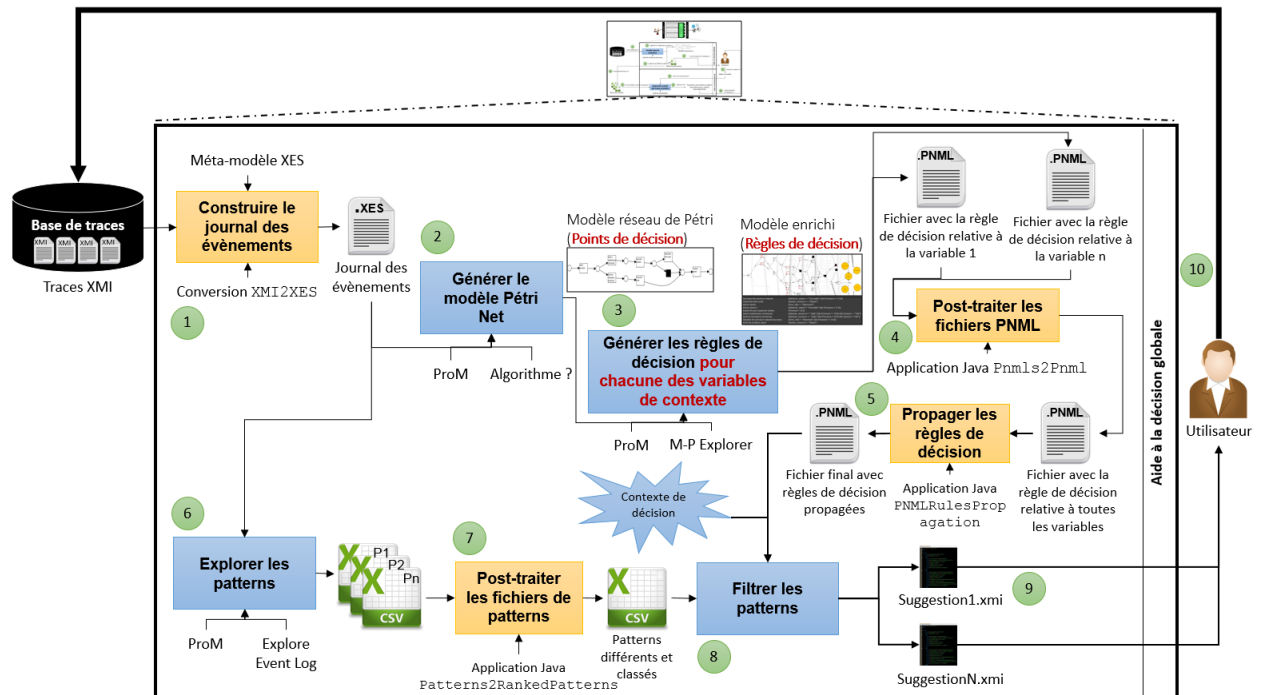


FIGURE 7.39 – Processus de l'aide à la décision globale amélioré pour permettre la génération des règles de décision précises [Es-Soufi et al., 2017b]. Les activités en jaune sont introduites pour lever les verrous identifiés.

supprimés pour ne pas limiter l'utilisateur qui peut, selon son expertise, faire le choix d'exécuter le pattern même si la règle qui contrôle son choix n'est pas vérifiée. Par conséquent, un classement de suggestions (c'est-à-dire les patterns les mieux adaptés au contexte de décision actuel) est proposé à l'utilisateur (étape 9 de la Figure 7.39). Le pattern, sélectionné par l'utilisateur, est ensuite capturé et stocké afin d'enrichir la base des traces et donc le journal des événements (étape 10 de la Figure 7.39).

7.4.5 Prédiction des paramètres des activités les plus adaptés au contexte de la décision

Il s'agit ici de prédire les ressources à utiliser et/ou la décision à prendre en compte, pour chaque activité du pattern choisi à travers le sous-module de l'aide à la décision globale. Nous avons montré, dans la Section 7.3.5, que la technique d'apprentissage la plus pertinente dans ce contexte, est la classification supervisée dont l'objectif est d'apprendre des valeurs connues (paramètres des activités des anciennes traces) pour en prédire de nouvelles. Avant d'aller choisir l'algorithme de la classification supervisée, nous devons d'abord choisir un outil permettant cette technique. Il existe, en effet, plusieurs outils

d'apprentissage automatique prenant en charge la classification supervisée. Nous choisissons l'outil "Weka" ¹⁴⁴ car il est facile d'utilisation, gratuit, open source, simple à utiliser et **bien documenté et considéré par la communauté de l'apprentissage automatique**.

Comme il n'est possible de considérer, à chaque fois, qu'une seule activité du pattern choisi lors de l'application des algorithmes de la classification supervisée en utilisant Weka, nous nous trouvons dans l'obligation de construire un training set par activité (Figure 7.40) afin de démarrer le processus de classification supervisée (présenté dans la Figure 7.41). En effet, pour un journal des événements contenant trois patterns $P_1 = \{A_1, A_2, A_3, A_4\}$, $P_2 = \{A_1, A_2, A_3, A_5\}$ et $P_3 = \{A_1, A_2, A_5, A_4\}$, nous extrayons les différentes exécutions des activités A_1, A_2, A_3, A_4 et A_5 et nous les enregistrons dans des journaux d'activités séparés $J_{A_1}, J_{A_2}, J_{A_3}, J_{A_4}$ et J_{A_5} ayant le format CSV.

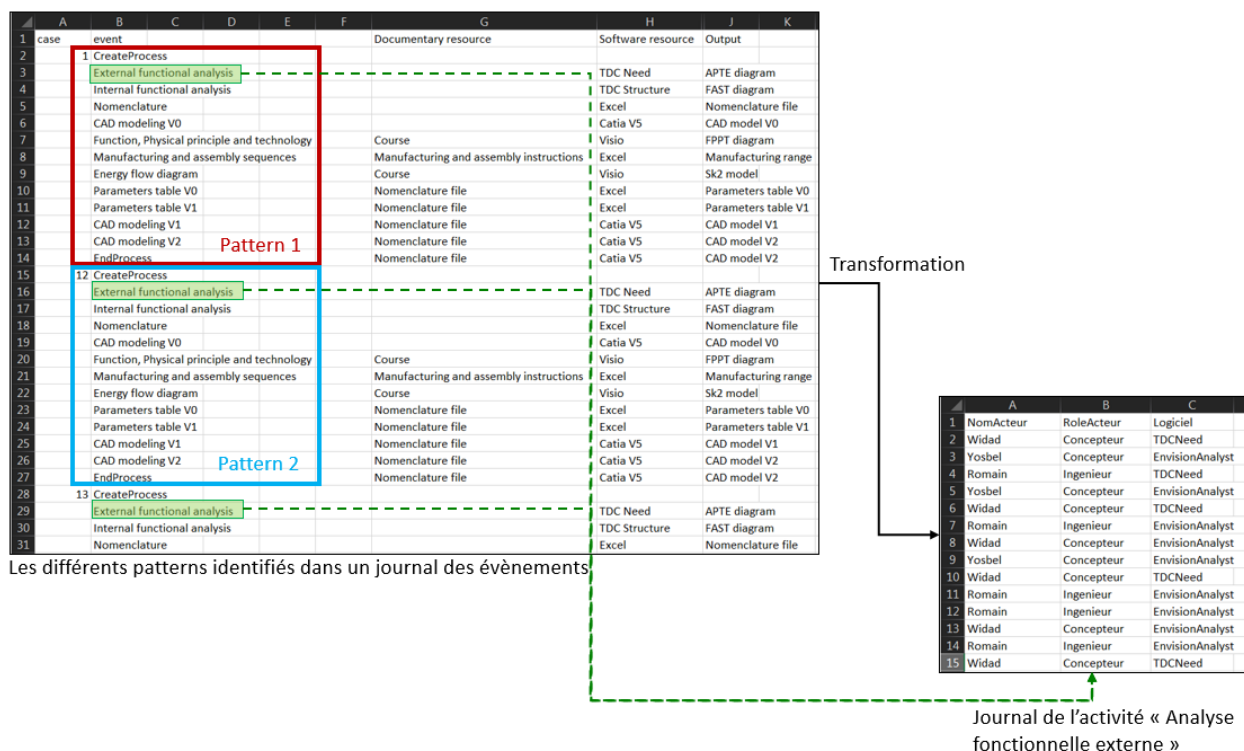


FIGURE 7.40 – Transformation d'un journal de patterns (issu de l'exécution des étapes 6 et 7 de la Figure 7.39) contenant toutes les activités des patterns identifiés d'un journal des événements, à un journal par activité.

Ayant le journal de l'activité considérée, le training set est, ensuite, construit comme présenté dans le Tableau 7.5, où X est le vecteur d'attributs d'entrée qui décrit les propriétés de l'activité en question et Y est la variable de sortie (étape 1 de la Figure 7.41). Le training set peut être pré-traité (étape 2 de la Figure 7.41) si certains attributs du vecteur X doivent être éliminés [Witten et al., 1999]. Ensuite, l'algorithme de classification supervisée, appelé aussi classifieur, doit être choisi parmi les arbres de décision, le SVM (Support Vector Machine) et les réseaux de neurones, entre autres (étape 3 de la Figure 7.41), le lecteur peut se référer au travail de Caruana and Niculescu-Mizil [2006], qui comparent la performance des différents algorithmes de la classification supervisée). Les algorithmes choisis sont ensuite appliqués sur le training set pour les entraîner (étape 4 de la Figure 7.41). Ensuite, ils sont évalués à l'aide d'un test set, où Y est inconnu pour chaque vecteur d'attributs X (étape 5 de la Figure 7.41). Le classifieur le plus précis, permettant de classer correctement les vecteurs X d'entrée en les affectant à la bonne variable Y, sera ensuite utilisé pour prédire la variable souhaitée lors de la nouvelle exécution de l'activité en question (étapes 6 et 7 de la Figure 7.41).

144. <http://www.cs.waikato.ac.nz/ml/weka/downloading.html>

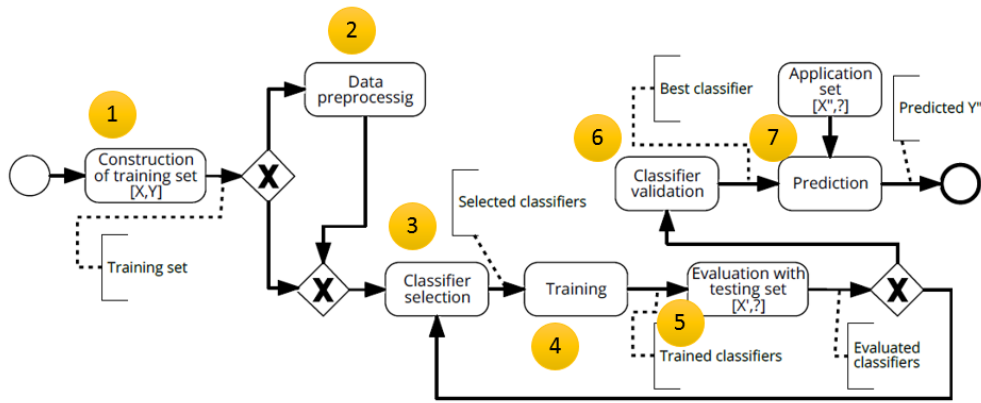


FIGURE 7.41 – Représentation BPMN du processus de la classification supervisée.

! Notons que l'évaluation des algorithmes de la classification supervisée n'est pas abordée dans ce travail de recherche puisque **nous faisons l'hypothèse que l'arbre de décision est l'algorithme le plus adapté à notre contexte.**

L'algorithme permettant de générer l'arbre de décision, appelé C4.5 et développé par [Quinlan \[1993\]](#), est implémenté dans l'outil Weka sous le nom "J48". Lors de la phase d'entraînement (étape 3 de la Figure 7.41), le J48 (en utilisant les paramètres par défaut) est capable de classer correctement 83.5468% des instances (vecteurs X) parmi les 3592¹⁴⁵ instances relatives à l'activité "External functional analysis" (Figure 7.40) du processus de conception de produit. Dans le présent travail de recherche nous supposons que 83.5468% des instances correctement classées est acceptable, **bien que l'idéal soit de chercher une méthode (modifier les paramètres de l'algorithme et/ou combiner plusieurs algorithmes, entre autres) permettant d'améliorer les résultats de prédiction afin d'avoir, idéalement, 100% d'instances correctement classées.**

Après avoir été entraîné et testé, l'algorithme J48 permet de prédire la valeur des paramètres d'activités (ressource logicielle dans la Figure 7.42) lors d'une nouvelle exécution de l'activité "External functional analysis". Dans cet exemple, la prévision prend en compte la ressource humaine qui exécute l'activité et l'heure du début de l'exécution de l'activité. Par conséquent, l'algorithme J48 est à 77,6% sûr que "Tom Stilinski" utilisera la ressource logicielle "TDC Need" pour exécuter l'activité "External functional analysis" à 9h25.

Instance	Human Resource	Start Time	Software
1	Tom Stilinski	09:25	?
2	Lydia Stanford	10:25	?
3	Rafael Dawson	14:15	?
4	Lydia Stanford	17:30	?
5	Rafael Dawson	07:45	?

➔

=== Predictions on test set ===				
inst#	actual	predicted	error	prediction
1	1:?	1:IDCNeed	0.776	
2	1:?	1:IDCNeed	0.742	
3	1:?	2:EnvisionAnalyst		0.997
4	1:?	1:IDCNeed	0.742	
5	1:?	2:EnvisionAnalyst		0.997

FIGURE 7.42 – Résultat de la prédiction de la ressource logicielle lors de l'exécution de l'activité "External functional analysis", par l'algorithme J48 avec les paramètres par défaut.

Le sous-module relatif à l'aide à la décision locale, du module d'aide à la décision, n'est pas encore entièrement implémenté (la transformation de la Figure 7.40 ainsi que les étapes de processus de la Figure 7.41 sont exécutés à la main). Le choix de l'algorithme permettant une meilleure prédiction ainsi que l'implémentation de la logique que nous venons de décrire dans cette partie, feront l'objet de nos travaux futurs.

145. Nous les avons construites à la main, aléatoirement sans suivre une logique définie pour ne pas impacter les résultats de la prédiction.

7.4.6 Synthèse et conclusion du module d'aide à la décision

Nous venons de décrire le module de l'aide à la décision dont la sortie est le pattern le plus adapté au contexte ainsi que les paramètres de ses activités. L'identification du bon pattern passe par une phase de filtrage où le contexte de la décision est comparé aux règles de décision contrôlant les différents patterns, et dont celui dont la règle est vérifiée par les données de contexte est proposé à l'utilisateur qui se trouve bloqué en devant l'exécuter du début et jusqu'à sa fin (par exemple, dans la Figure 7.43, l'utilisateur doit continuer à exécuter le premier pattern même si ($V_1 \geq 20$) au niveau de l'activité A_6). Ceci n'est vrai que si le contexte de décision ne change pas, parce que la vérification continue n'est pas prise en compte par l'IDSS qui est considéré comme statique à ce niveau (c-à-d il n'effectue le filtrage qu'au début de l'exécution du processus). Cependant, le contexte de décision peut être assujéti à plusieurs changements au niveaux des données de contexte (l'exemple le plus simple est l'état de disponibilité d'une ressource qui peut changer à tout moment et d'une façon imprévisible). Ceci nous amène à poser la question suivante :



Verrou scientifique 3

Comment permettre à l'IDSS de **naviguer** de manière **dynamique** entre les différents **patterns**, au cours du processus de l'aide à la décision, pour proposer les activités qui satisfont le contexte si un **changement** au niveau des variables de contexte est identifié ?

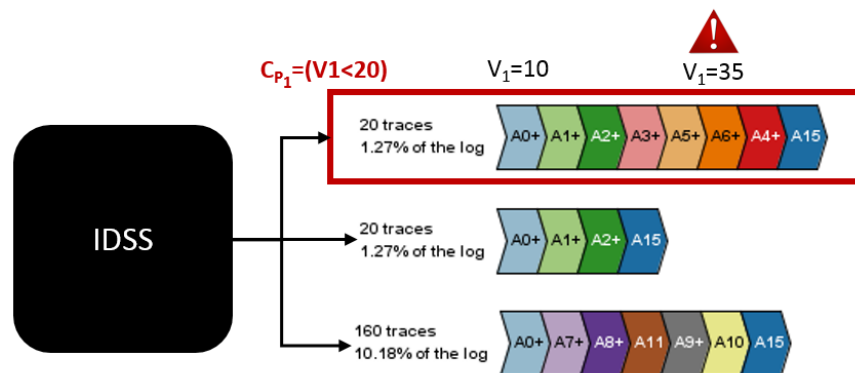


FIGURE 7.43 – Nécessité d'adapter la proposition des patterns au contexte de la décision.

7.5 Module de gestion des changements contextuels : vers un IDSS dynamique

7.5.1 Qu'est ce que la gestion des changements ?

La définition la plus basique de la gestion du changement est quel est le processus dans lequel un changement est identifié, analysé, évalué, planifié et propagé [Crnkovic et al., 2003]. Trois principaux types de changement sont identifiés dans la littérature :

- Premièrement, le changement qui précède une décision, tel qu'un changement d'organisation, un changement technologique, etc. [Helfat et al., 2009]. Ce type de changement aide les entreprises à rester pertinentes en se tenant au courant des avancées technologiques et des changements sur le marché, et est généralement planifié et évalué avant de décider de l'implémenter.
- Le deuxième type est la gestion des changements au niveau du produit, aussi appelé "engineering design change" [Ullah et al., 2016]. Ce type concerne les changements qui sont appliqués sur les produits pour répondre aux besoins des clients, entre autres, et est donc lié au premier type. La gestion

des changements, ici, consiste à propager les données modifiées tout au long du cycle de vie du produit.

- Le troisième type est la gestion du changement contextuel [Burstein et al., 2011]. Il consiste à prédire, identifier ou gérer les changements de contexte afin de permettre l'aide à la décision dynamique (également appelée aide à la décision en temps réel). En effet, dans les environnements dynamiques, seules les données contextuelles en temps réel peuvent fournir aux décideurs les connaissances exactes nécessaires à la prise de décision [Delic et al., 2001].

Dans le présent travail de recherche, nous nous intéressons à la **gestion des changements contextuels** car elle répond à notre objectif, qui est de **mettre à jour l'IDSS en informant des changements qui affectent le contexte décisionnel**. Par conséquent, seul le dernier type de changement est exploré et discuté par la suite.

7.5.2 État de l'art sur les techniques de gestion des changements contextuels

Une décision est définie par Resulaj et al. [2009] comme étant un engagement irréversible, vis-à-vis d'une proposition, fondé sur des preuves, les paramètres attendus et le **contexte** dans lequel la décision est prise. Le contexte est un concept essentiel et puissant pour une prise de décision efficace en temps réel. La définition du contexte qui est adoptée par la plupart des chercheurs est qu'il s'agit de "toute information pouvant être utilisée pour caractériser la situation d'une entité" [Dey, 2001]. Ces informations de contexte peuvent être importées à partir des capteurs, dérivées d'autres informations, raisonnées, calculées ou entrées explicitement par les utilisateurs [Burstein et al., 2011]. Il existe deux catégories de contexte selon Delir Haghghi et al. [2008] : le contexte dynamique (constitué des informations dynamiques telles que la fréquence cardiaque et les flux de données de pression entre autres) et le contexte statique dans lequel les informations sont statiques telles que les noms des décideurs, etc. [Burstein et al., 2011].

De nombreuses approches, dans la littérature, visent à comprendre le comportement du contexte de prise de décision. Certaines approches visent à modéliser le contexte et sont basées sur les techniques de raisonnement bayésien [Fox et al., 2003], la logique floue [Delir Haghghi et al., 2008], la modélisation graphique [Henricksen et al., 2002] et quelques autres approches de modélisation basées sur les ontologies [Truong et al., 2005]. Ziebart et al. [2008] proposent une présentation du contexte à l'aide de la modélisation du comportement de l'utilisateur. En plus des approches basées sur la modélisation, d'autres approches visent à prédire le contexte futur afin d'améliorer la gestion des changements contextuels, en utilisant des techniques de la fouille de données telles que la classification, les réseaux de neurones et les techniques basées sur des statistiques [Burstein et al., 2011]. D'autres approches utilisant les systèmes experts et les moteurs de règles pour prédire le contexte [Burstein et al., 2011].

7.5.3 Discussion des techniques de gestion des changements contextuels

La plupart des approches listées dans la sous-section précédente sont liées à la modélisation ou à la prédiction de contexte. Ces techniques sont confrontées à des défis qui concernent, entre autres, la difficulté de prédire les actions humaines qui ne peuvent être connues avant et qui, dans la plupart des situations de décision, font partie du contexte de décision en temps réel [Nurmi et al., 2005]. Ce dernier, tel que défini par Burstein et al. [2011], est une représentation de l'environnement en tant qu'ensemble de données contextuelles qui, dès qu'elles sont obtenues, fournissent aux décideurs les connaissances nécessaires pour décider. Dans notre contexte, ces données contextuelles sont à la fois statiques et dynamiques et ne sont pas toujours prédéfinies ou déduites car certaines d'entre elles changent constamment, ne seront connues qu'en temps réel ou interféreront avec d'autres contextes (par exemple, la pression d'une station de gaz dépend du comportement des autres stations voisines, de sa durée de vie et/ou le changement climatique). En outre, nous devons exprimer le contexte de manière déterministe, car il est utilisé pour filtrer les patterns en fonction des règles de décision sous la forme : si ($pression > 25$), alors "envoyer un technicien d'urgence"; ici, une valeur précise de la pression est nécessaire et non pas la distribution ou la prévision des valeurs de pression.

Pour répondre à notre objectif qui est d'informer notre IDSS des changements de contexte, nous avons besoin d'une compréhension continue des changements environnementaux et d'une connaissance explicite du contexte actuel. Par conséquent, nous supposons que les approches listées dans la Section 7.5.2 ne sont pas pertinentes dans notre contexte.

7.5.4 Proposition : application permettant la gestion dynamique des changements contextuels

Pour lever le cinquième verrou technologique, identifié dans la Section 7.4.6, nous introduisons le module de gestion des changements contextuels à deux niveaux, et dont le processus est présenté dans la Figure 7.44. La première étape consiste à exécuter l'IDSS statique (comme expliqué dans la Figure 7.39) pour générer les fichiers relatifs aux "règles de décision" et aux "patterns ordonnés" à partir de la base de trace de processus qui contient les exécutions de processus antérieures, générées par rapport au méta-modèle de trace identifié dans le travail de Es-Soufi et al. [2017a]. Le fichier "règles de décision", généré par l'IDSS statique, contient les règles qui contrôlent le déroulement du processus et le deuxième document, généré toujours par l'IDSS statique, concerne les différents patterns, qui ont été identifiés dans la base de traces, et qui sont triés par ordre de fréquence décroissant.

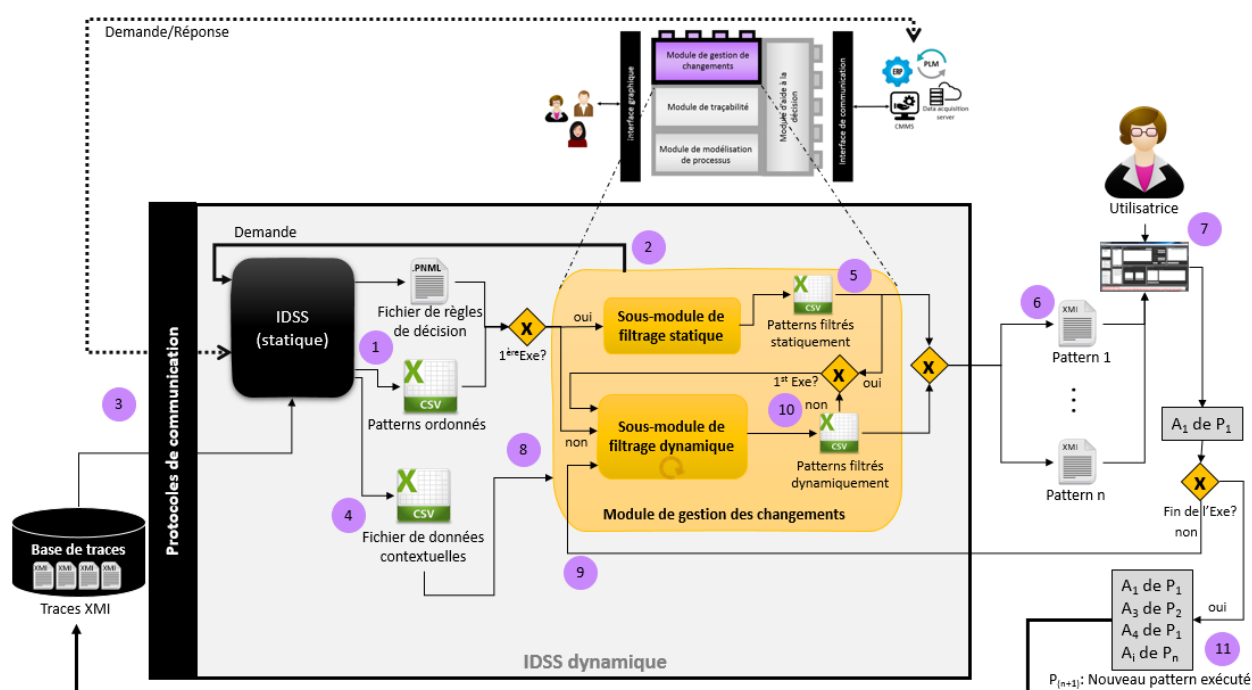


FIGURE 7.44 – Processus suivi par le module de gestion des changements.

Une fois que les deux fichiers sont générés par l'IDSS statique (étape 1 de la Figure 7.44), et s'il s'agit de sa première exécution pour un problème de décision donné, le sous-module de filtrage statique, du module de gestion des changements, est appelé. Ce dernier demande le contexte de décision actuel à l'IDSS (étape 2 de la Figure 7.44) qui, à son tour, transmet la demande aux systèmes connectés via la couche de communication (étape 3 de la Figure 7.44). Parmi les systèmes connectés à l'IDSS, nous citons le serveur d'acquisition de données qui collecte les données des différents capteurs (capteurs de pression, par exemple), du système de gestion informatisée de la maintenance (GMAO) qui fournit des données de maintenance telles que la date de la panne et/ou de la réparation, du progiciel de gestion intégré (ERP) fournissant des informations sur les processus opérationnels et le logiciel PLM (logiciel de gestion du cycle de vie du produit) contenant toutes les informations produit/processus tout au long du cycle de vie du produit (par exemple, les données de conception).

Une fois que l'IDSS reçoit les données contexte des systèmes connectés, il les copie dans un fichier de données contextuelles, qui est envoyé au module de filtrage statique (étape 8 de la Figure 7.44). L'exécution

de ce module génère le fichier de patterns classés et statiquement filtrés (étape 5 de la Figure 7.44), qui associe à chaque pattern, du fichier des patterns ordonnés (généré dans l'étape 1 de la Figure 7.44), un indicateur de règle de décision dont la valeur est 1 ou 0. 1 signifie que le pattern satisfait les règles de décision dans les conditions de contexte actuelles et est donc prêt à être proposé à l'utilisateur (étape 6 de la Figure 7.44) via l'interface utilisateur graphique IDSS présentée dans le travail de Roucoules et al. [2016]. L'utilisateur choisit un des patterns proposés et commence à exécuter sa première activité (étape 7 de la Figure 7.44). Si l'activité exécutée ne fait pas référence à la fin du processus (c'est-à-dire "activityEvent \neq end" comme expliqué dans le travail de Es-Soufi et al. [2017a]), alors l'activité et le pattern auquel elle appartient sont envoyés au module filtrage dynamique qui prend le relais (étape 9 de la Figure 7.44). Ce dernier demande le contexte à l'instant t (exécution des étapes 2, 3, 4, 8 de la Figure 7.44), et prend en entrée les fichiers de "données contextuelles" et les fichiers de "patterns classés et statiquement filtrés" ainsi que la référence de la dernière activité exécutée et du pattern auquel elle appartient, afin de mettre à jour les patterns classés et statiquement filtrés en générant un nouveau fichier appelé fichier de "patterns classés et dynamiquement filtrés", où l'indicateur de règle de décision est mis à jour en re-testant si les règles de décision sont satisfaites ou non par le contexte de décision actuel. Si ce dernier satisfait par exemple la règle de P_2 aux instants $t_{0 \rightarrow 3}$ (Figure 7.45), l'indicateur de règle de décision des activités $A_{8 \rightarrow 11}$ est mis à 1 tandis que celui des activités $A_{1 \rightarrow 7}$ et $A_{14 \rightarrow 16}$ est mis à 0. Si à t_4 le contexte ne satisfait plus la règle du pattern P_2 alors l'indicateur de règle de décision sur les activités postérieures de P_2 s'actualise et prend la valeur 0, alors que celui des activités postérieures du pattern P_1 , dont la règle est satisfaite, se met à 1 et ainsi de suite. Cette navigation entre les patterns permet, en plus de proposer le pattern le plus adapté au contexte, d'explorer d'autres processus conduisant ainsi vers l'innovation.

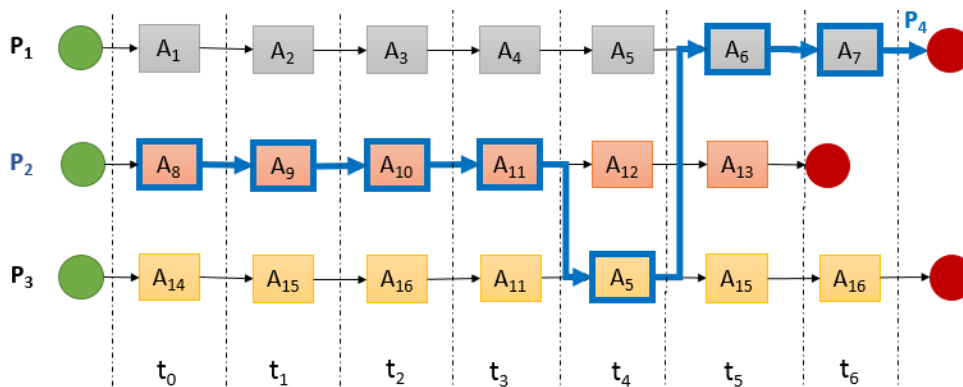


FIGURE 7.45 – Passage d'un pattern à un autre selon la règle et le contexte de la décision.

Les patterns dont l'indicateur de règle de décision est égal à 1 sont proposés à l'utilisateur en fonction de leur ordre de fréquence (étape 6 de la Figure 7.44). Ensuite, les étapes 7, 9, 2, 3, 4, 8, de la Figure 7.44, sont répétées jusqu'à ce que la dernière activité exécutée réfère à la fin du processus (c'est-à-dire "activityEvent == end" comme expliqué dans le travail de Es-Soufi et al. [2017a]). Dans ce cas, le nouveau modèle contenant toutes les activités exécutées est envoyé à la base de traces pour l'enrichir (étape 11 de la Figure 7.44). Notons que le nouveau pattern exécuté est envoyé à la base même s'il est équivalent à un des patterns existants, car l'activité qui génère les patterns classés (Figure 7.39) post-traite tous les patterns et n'en sélectionne que les patterns distincts en associant à chacun d'eux le numéro de son occurrence (fréquence d'apparition).

Le module de gestion des changements proposée permet à l'IDSS statique de la Figure 7.39, d'être informé -de manière dynamique- des changements des valeurs des variables de contexte. Il est entièrement implémenté en utilisant le langage de programmation Java.

7.6 Synthèse et conclusion

Dans le présent chapitre, nous avons découvert les deux modules qui constituent le cœur de notre proposition. Le module de l'aide à la décision permet de supporter les utilisateurs aux niveaux global en

leur proposant le pattern le plus adapté au contexte, et au niveau local, en leur proposant les paramètres d'activités les plus adaptés au contexte. Ce module est supporté par le module de gestion des changements qui permet de l'informer dynamiquement des changements au niveau des variables de contexte, permettant ainsi au module de l'aide à la décision d'être efficace et réactif aux changements. Les propositions ont été testées en utilisant un journal des événements construit à la main. Il nous reste, par conséquent, à tester la proposition sur un exemple du monde réel pour la vérifier et la valider. Ceci fera l'objet du chapitre suivant, où un processus réel de supervision des réseaux de gaz, en temps réel, sera considéré.

Troisième partie

VÉRIFICATION & VALIDATION, CONCLUSION & PERSPECTIVES

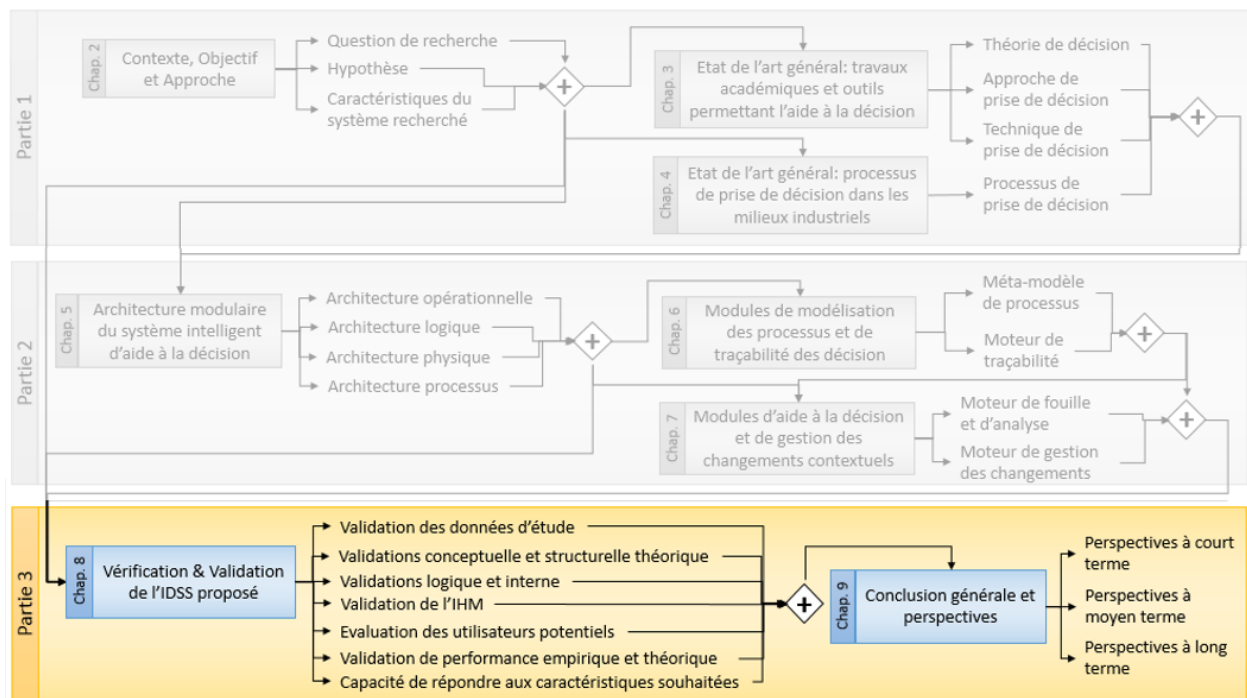


FIGURE 7.46 – Structure de la troisième partie du manuscrit (décrite en utilisant le formalisme BPMN [OMG, 2013]).

Cette dernière partie se compose de deux chapitres. Dans le chapitre 8, nous vérifions et validons le système proposé selon différents points de vue. Dans le chapitre 9, nous concluons le manuscrit en rappelant quels étaient la problématique, les objectifs scientifiques, l'hypothèse de recherche, l'approche adoptée et les résultats de la validation. Ensuite, nous présentons les perspectives à court, moyen et long termes ; qui permettent d'améliorer le système proposé et d'ouvrir de nouvelles voies de recherche.

Vérification & Validation de l'IDSS proposé

"No amount of experimentation can ever prove me right; a single experiment can prove me wrong."

Albert Einstein

8.1	Introduction	183
8.1.1	Besoin de vérifier et valider son système	183
8.1.2	Qu'est ce que la V&V et comment nous la définissons dans notre contexte ?	183
8.2	État de l'art sur les techniques de vérification et de validation des systèmes	184
8.3	Discussion de l'état de l'art sur les techniques de V&V : vers un cadre de V&V pour les IDSSs	189
8.4	Vérification et validation de l'IDSS proposé	190
8.4.1	Validité des données d'étude	190
8.4.2	Validation conceptuelle et validation structurelle théorique	191
8.4.3	Vérification et validité interne	191
8.4.4	Validité de l'IHM	194
8.4.5	Le projet industriel ayant créé le besoin de concevoir un système IDSS	194
8.4.6	Évaluation des utilisateurs potentiels	196
8.4.7	Validation de performance empirique	198
8.4.8	Validation de performance théorique	198
8.4.9	Capacité de répondre aux caractéristiques souhaitées	198
8.5	Synthèse et conclusion	199

Allez vous croire une personne si elle vous dit qu'elle dîne tous les soirs sur la lune avec "Shiawtzo", le bel extraterrestre à trois yeux ? Ou encore si elle a un appareil, offert par "Shiawtzo", qui permet de transformer la poubelle en argent ? Vous allez plutôt croire qu'elle souffre de troubles psychologiques, nous en sommes sûrs. Le cerveau humain ne peut, en effet, pas admettre les faits s'ils ne sont pas prouvés sur le plan réel. Ce concept de preuve, qui est devenu il y a quelques siècles déjà, l'un des fondements de la méthode scientifique, permet de vérifier et valider des hypothèses (en ayant recours à des arguments théoriques ou expérimentaux), des méthodes (en évaluant leurs performances) et des systèmes (en s'assurant qu'ils sont conformes aux exigences des parties-prenantes et qu'ils respectent les caractéristiques de conception attendues), etc.

Ce chapitre a pour but de vérifier et valider (ou pas !) le système IDSS que nous avons proposé. Nous commençons d'abord par définir les notions de vérification et de validation avant de découvrir les techniques, proposées dans la littérature, permettant de vérifier et valider un système. De ces dernières

nous définissons le cadre de V&V que nous adoptons pour étudier la validation du système IDSS proposé. Ensuite, nous présentons le projet "GONTRAND", étant à l'origine de la création du besoin de concevoir le système IDSS et ayant pour objectif la supervision en temps réel des réseaux de gaz favorisant l'injection du gaz vert. Le chapitre est clôturé après la prononciation du verdict (le système IDSS est-il validé ou pas ?).

8.1 Introduction

8.1.1 Besoin de vérifier et valider son système

Au début du cycle de vie d'un système quelconque, les clients finaux et les ingénieurs développeurs se réunissent pour identifier les besoins et les caractéristiques de conception attendues du système. Les ingénieurs procèdent ensuite à la traduction de ces besoins en un ensemble de spécifications, en identifiant l'ensemble d'exigences fonctionnelles et non fonctionnelles. Les premières spécifient les fonctions que le système doit exécuter, tandis que les deuxièmes définissent le comportement du système telles que sa performance et sa fiabilité, entre autres. Cette identification des exigences représente un processus hautement itératif dont la condition d'arrêt est l'atteinte d'un niveau de maturité suffisant pour commencer le développement du système.

Le contexte économique actuel ainsi que l'imperfection humaine¹⁴⁶, font que les systèmes développés doivent être vérifiés et validés avant d'être acquis par le client final. Les activités de vérification et de validation sont assurées au plus tôt et tout au long du processus de développement du système, pour :

- **Satisfaire le client**, en vérifiant que le système répond aux besoins des clients exprimés dans le cahier des charges [NASA, 2017],
- **Réduire les coûts**. En effet, la vérification des systèmes tôt pendant leur développement permet de réduire le risque de déviation du comportement du système par rapport à celui attendu, et donc évite tout retour pour correction ou re-développement [BKCASE, 2017],
- **Contrôler la qualité du système**, en assurant que le processus de conception est correctement suivi [Wingate, 2003],
- **Maîtriser le système**, en le documentant tout au long de son développement [Bellagamba, 2012],
- **Accélérer le processus de certification du système**, en prouvant qu'il répond aux exigences réglementaires appropriées [NASA, 2017]. Etc.

Les notions de validation et de vérification (ou encore V&V), étant deux éléments principaux de l'ISO 9000 [Poksinska et al., 2002], sont définies différemment d'une organisation à l'autre, selon le domaine d'application et/ou le produit développé (système, modèle, programme informatique, etc.), et sont parfois utilisées interchangeablement. Il est par conséquent nécessaire d'explorer quelques unes de ces définitions afin d'adopter celle qui nous paraît la plus pertinente.

8.1.2 Qu'est ce que la V&V et comment nous la définissons dans notre contexte ?

La vérification d'un système est définie par le PMI [2018] comme étant l'évaluation de sa conformité avec une exigence ou une spécification, tandis que la validation est vue comme étant le fait d'assurer qu'un système répond bien aux besoins du client et des autres parties prenantes identifiées. Bien qu'ici, le contraste entre les deux notions n'est pas si frappant, nous comprenons que le PMI [2018] qualifie la vérification comme un processus interne à l'organisation qui développe le système, tandis que la validation est vue comme un processus externe qui implique des clients externes. Pham [1999] définit la vérification d'un système logiciel comme étant *"the process of ensuring that the software is performing the process right"*, ceci implique l'utilisation des résultats d'une phase " P_1 " comme entrée de la phase suivante " P_2 " (Par exemple, les spécifications de conception d'un produit, résultat de la phase de sa conception, alimentera la phase de son manufacturing). Ainsi, le système est dit vérifié si à chaque fois, la sortie de l'activité implémente ses spécifications d'entrée. Le même auteur, Pham [1999], définit la validation comme étant *"the process of ensuring that the software is performing the right process"*; il s'agit ici de s'assurer que les exigences du système traduisent fidèlement les besoins et les objectifs des parties prenantes. Le standard "IEEE-STD-610" de l'ISO [1990] définit la vérification comme étant *"a test of a system to prove that it meets all its specified requirements at a particular stage of its development"*, et la validation comme

146. En effet, des études sur la fiabilité humaine ont montré que les personnes formées pour exécuter une activité donnée, commettent environ une à trois erreurs par heure dans les meilleurs scénarios [BKCASE, 2017].

étant "an activity that ensures that an end product stakeholder's true needs and expectations are met". La [BKCASE \[2017\]](#) adopte la même logique en définissant la vérification comme étant l'ensemble des actions utilisées pour s'assurer de l'exactitude de chaque élément du système; la validation est, par contre, définie comme étant la confirmation que les exigences ont bien été remplies. La [NASA \[2017\]](#) définit la vérification des modèles comme étant le "degree to which a model accurately mets its specifications. Answers "Is it what I intended?" et la validation des modèles comme étant "the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model".

De ces définitions et de la logique modulaire que nous avons adoptée pour construire notre système IDSS (discutée dans le Chapitre 5), nous considérons la définition suivante :

- La **vérification de l'IDSS** consiste à s'assurer que chacun des quatre modules du système est conforme avec l'exigence qui lui est appropriée. De plus, elle consiste à s'assurer que les programmes développés, pour supporter les activités d'un module donné, sont conformes à leurs spécifications. **Notons que cette notion concerne le milieu intérieur de l'IDSS**, comme le montre la Figure 8.1.
- La **validation de l'IDSS** consiste à s'assurer que le produit final, l'IDSS, résout le problème formulé dans la question de recherche, et qu'il répond bien aux besoins des clients lorsqu'il est utilisé dans des environnements théoriques et réels. **Notons que cette notion concerne le milieu extérieur de l'IDSS**, comme le montre la Figure 8.1.

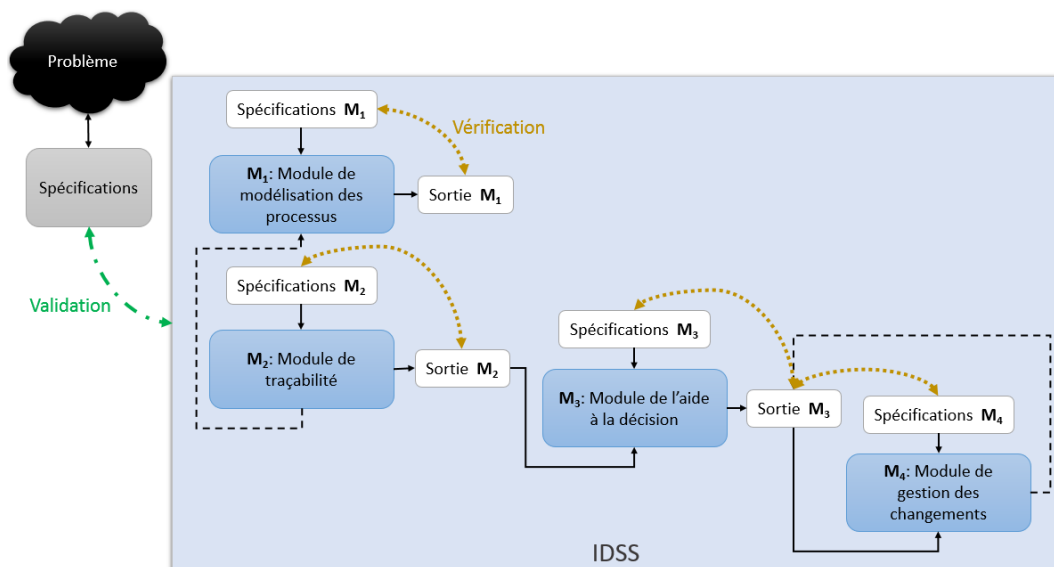


FIGURE 8.1 – Activités de vérification et validation dans notre contexte. Se référer à la Figure 1.1 pour plus d'informations sur le formalisme utilisé.

Plusieurs techniques permettant, la vérification et la validation des systèmes, sont proposées dans la littérature. Nous souhaitons explorer quelques unes pour choisir la technique la plus pertinente dans notre contexte. Ceci fera l'objet de la section suivante.

8.2 État de l'art sur les techniques de vérification et de validation des systèmes

[Balci \[1997\]](#) classifie les techniques de la V&V en quatre grandes catégories comme le montre la Figure 8.2 :

- **Techniques informelles** : elles font partie des techniques les plus utilisées. Elles sont dites informelles car les outils ou les approches utilisées ne sont constitués d'aucun formalisme mathématique, et reposent largement sur le raisonnement humain qui peut être, parfois, subjectif.

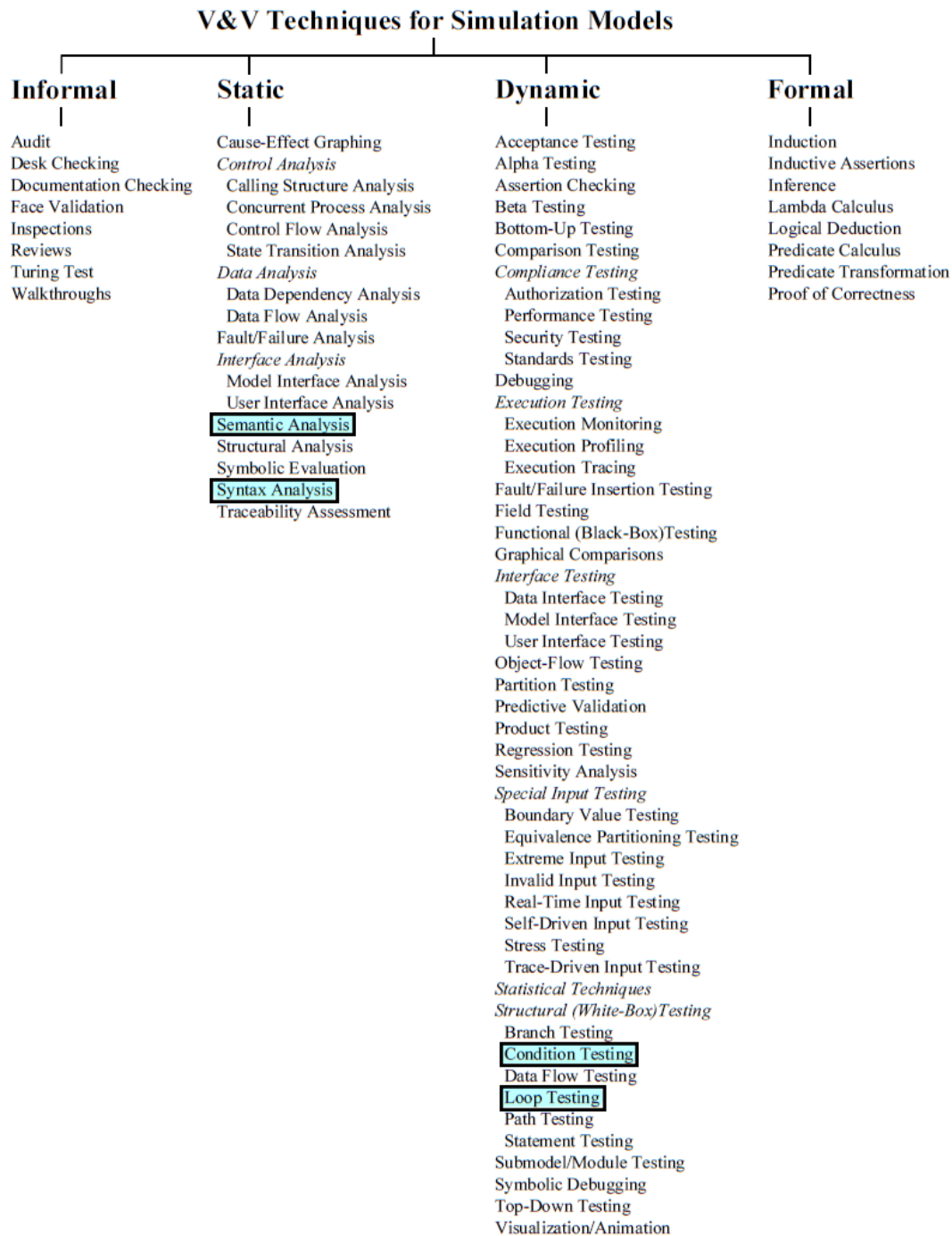


FIGURE 8.2 – Techniques de vérification et de validation des modèles de simulation selon Balci [1997]. Les techniques surlignées en bleu (et encadrées en noir) sont celles que nous avons utilisées pour vérifier et valider les programmes informatiques des modules de l'IDSS.

- **Techniques statiques** : elles sont utilisées pour évaluer la conception du modèle statique et le code source (l'implémentation), sans exécuter le modèle avec une machine. Ces techniques visent à vérifier la structure du modèle, les flux de données et de contrôle, la syntaxe, etc. Pour assurer une V&V complète, ces techniques doivent être appliquées en conjonction avec les techniques dynamiques.
- **Techniques dynamiques** : Contrairement à celles statiques, les techniques dynamiques sont basées sur l'exécution du modèle par la machine afin d'évaluer son comportement. Elles ne se contentent pas d'examiner le résultat de l'exécution, mais observent également le comportement du modèle en cours d'exécution. Le débogage, la simulation et les tests d'exécution sont des exemples de ces techniques.

- **Techniques formelles** : Ces techniques sont basées sur un raisonnement mathématique formel et sur des preuves.

Plusieurs travaux proposent des méthodologies pour valider le système, dans son ensemble, selon plusieurs perspectives. Selon Landry et al. [1983], le processus de validation parcourt les quatre phases de la modélisation comme le montre la Figure 8.3 : la première étant la "problématique", elle définit un problème réel qui crée un besoin de conception donné. La deuxième phase est le "modèle conceptuel" qui fait référence à l'image mentale qui représente la problématique comme perçue par les concepteurs et les utilisateurs. La troisième phase est le "modèle formel" qui se voit comme une traduction du modèle conceptuel en symboles mathématiques et/ou langages informatiques. La dernière phase concerne la "solution" qui est vue comme le résultat du processus de modélisation et obtenue à partir du modèle formel en utilisant des techniques et/ou procédures. Le système modélisé subit, selon Landry et al. [1983], une validation selon cinq perspectives :

- La **validation conceptuelle** : il s'agit ici de s'assurer que les concepts considérés dans le modèle conceptuel représentent bien la problématique, que le problème a été abordé de la bonne perspective et que cette dernière est capable d'amener à la bonne solution.
- La **validation logique** : il s'agit ici de **vérifier** que le modèle formel décrit correctement la problématique comme définie dans le modèle conceptuel en s'assurant que toutes les variables ou relations pertinentes sont présentes dans le modèle formel. Notons que ce type de validation inclue aussi la vérification (s'assurer que le modèle formel est implémenté comme décrit dans le modèle conceptuel).
- La **validation expérimentale** : il s'agit ici d'évaluer la qualité et l'efficacité du mécanisme et/ou la technique utilisés (algorithme, méthode, etc.) pour produire la solution.
- La **validation opérationnelle** : il s'agit ici de déterminer la qualité de la solution et son applicabilité par rapport à la problématique. Ce type de validation aide les développeurs pour accepter ou rejeter la solution, et leur apporte des recommandations concernant le modèle formel (variables à ajouter, etc.).
- La **validation des données** : ce type de validation s'oriente vers le test de la suffisance, la disponibilité, la justesse et la pertinence des données utilisées lors du développement et l'analyse du DSS.

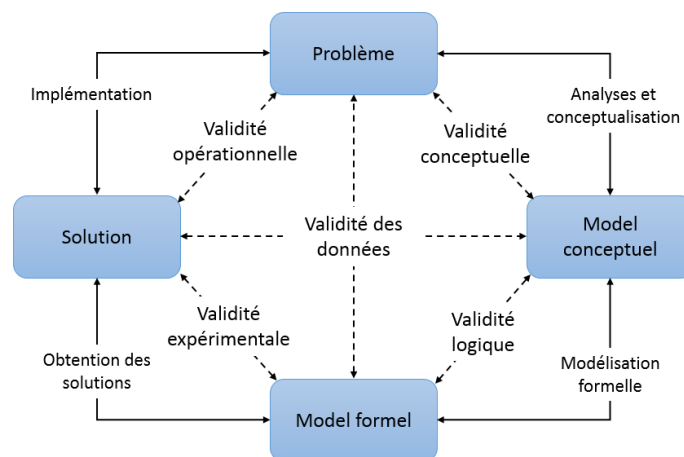


FIGURE 8.3 – La validation des systèmes vue par Landry et al. [1983].

Finlay and Wilson [1997] assurent que les systèmes d'aide à la décision (DSS) requièrent cinq types de validation pour couvrir tous leurs composants (Figure 8.4). Il s'agit de :

- La **validation du modèle logique** : il s'agit ici de valider le DSS analytiquement et théoriquement. La première consiste à contrôler séparément chaque relation du modèle logique tandis que la deuxième a pour but de s'assurer que les variables considérées sont théoriquement bonnes.
- La **validation du modèle de données** : il s'agit ici de vérifier la justesse et la précision des données.
- La **validation de l'interface** du DSS : il s'agit ici d'évaluer la fréquence de l'utilisation du DSS, et vérifier si ses caractéristiques aident vraiment les utilisateurs et les encouragent à l'utiliser.

- La **validation du concepteur** du DSS : il s'agit ici d'évaluer le développeur ayant conçu le DSS en termes de son expertise et est ce qu'il a conçu des systèmes similaires, entre autres.
- La **validité générale** du DSS : ce type fait référence à la validité du DSS dans son ensemble, il regroupe les validations conceptuelle, expérimentale, opérationnelle et la vérification (validation logique) comme expliqué ci-haut. De plus, il considère la **robustesse** qui se voit comme la capacité d'utiliser le DSS dans des contextes, autres que celui dans lequel il a été conçu, en ayant des résultats justes, et la **validité interne** qui consiste à évaluer la clarté et la fiabilité du DSS en testant si, dans les mêmes conditions, il produit les mêmes résultats.

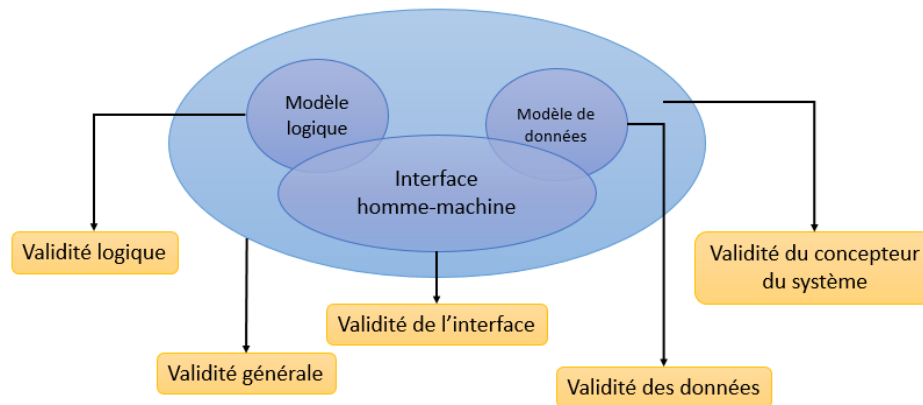


FIGURE 8.4 – La validation des systèmes vue par Finlay and Wilson [1997].

Borenstein [1998] qualifie la validation d'un DSS comme un processus évolutionnaire à deux étapes principales comme le montre la Figure 8.5 :

- La **validation en laboratoire** : c'est un ensemble d'expériences qui doivent se faire au cours au développement du DSS, et qui ont lieu dans des environnements "artificiels" construits par l'équipe chargée d'évaluer le DSS. Certains de ces tests peuvent faire appel à des clients finaux ou des utilisateurs potentiels qui contribuent à la validation à travers des questionnaires, des entretiens et des ateliers de présentation du prototype développé. Ce type de validation englobe :
 - La **validité apparente** (face validity) du DSS : elle est vue comme un mécanisme de retour d'information permettant de raffiner, de reformuler et/ou de réviser les prototypes du DSS. Son objectif principal est d'arriver à une cohérence entre l'opinion du concepteur et celui de l'utilisateur final. Elle garantit, en effet, que le problème formulé est conforme avec le problème réel énoncé et qu'il est suffisamment structuré pour qu'une solution crédible puisse être trouvée avant que le développement approfondi et détaillé du DSS ne se poursuive.
 - La **vérification et la validation des sous-systèmes** du DSS : elle consiste à tester, vérifier et valider chacun des modules du DSS au fur et à mesure de leur développement. L'objectif principal est d'identifier les zones, dans chacun des sous-systèmes dans chaque module du DSS, dans lesquelles un effort de développement ou de révision doit être assuré.
 - La **validation prescriptive** : elle consiste à valider le DSS à l'aide de tests en laboratoire dans lesquels les résultats sont connus. Ceci dit que le DSS est testé en utilisant des données d'entrée dont les résultats correspondants sont connus, les résultats du DSS sont ensuite comparés avec ces résultats connus.
 - L'**évaluation des utilisateurs** : c'est le processus dans lequel les utilisateurs, qui n'étaient pas impliqués dans le développement du DSS, peuvent déterminer, avec un certain degré de confiance, si les résultats du DSS peuvent ou non être considérés.
- La **validation de terrain** : ici, le DSS est utilisé dans un contexte réel (d'où le nom *field tests*), où les erreurs de performance sont identifiées. Miser and Quade [1985] voient que la validation sur le

terrain, bien qu'elle soit parfois assez coûteuse, **est la plus efficace entre tous les autres types**. C'est, en effet, une étape très souhaitable avant une implémentation complète parce que ce n'est que sur le terrain où le comportement réel d'un DSS est observé, car les données de test sont réelles plutôt que d'être construites en laboratoire (et donc peuvent être subjectives et impactent négativement le processus de validation).

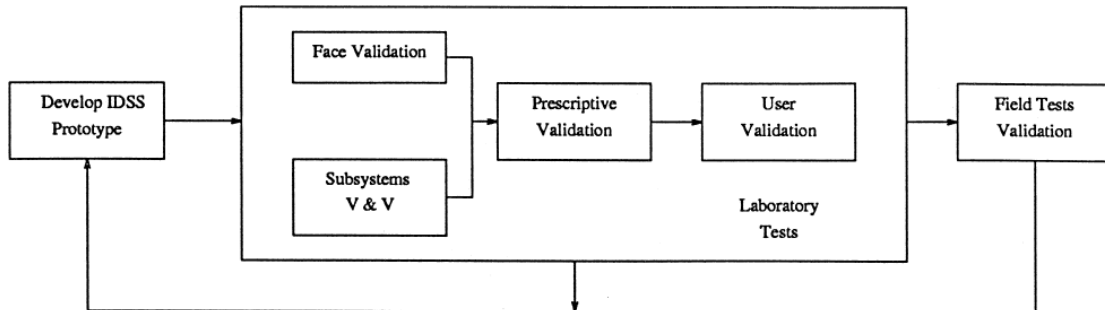


FIGURE 8.5 – Processus de validation des systèmes à deux niveaux [Borenstein, 1998].

Seepersad et al. [2006] essaient de répondre à la question la plus posée dans le domaine de l'ingénierie : "Comment valider une méthode de conception?" en proposant le concept du "validation square" permettant de valider les méthodes de conception (Figure 8.6), où quatre types de validations sont considérées :

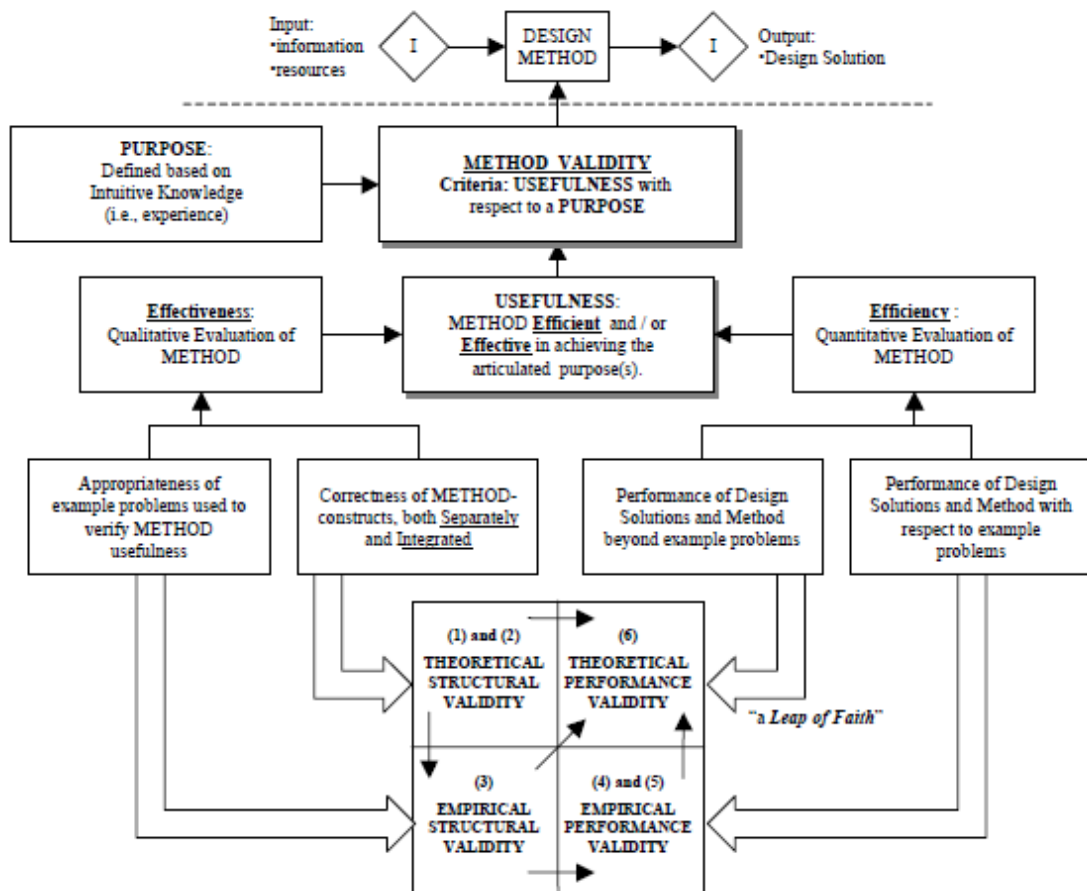


FIGURE 8.6 – La validation des systèmes vue par Seepersad et al. [2006].

- La **validation structurelle théorique** : il s'agit ici de tester deux choses pour vérifier la cohérence de la méthode et ses concepts. La première concerne l'acceptation de la validité des concepts individuels

constituant la méthode, et la deuxième concerne l'acceptation de la façon comment ces concepts sont assemblés dans la méthode.

- La **validation structurelle empirique** : il s'agit ici de tester la pertinence des exemples utilisés pour vérifier la méthode. Ceci revient à s'assurer que les problèmes réels, dans lesquels la méthode sera utilisée, sont suffisamment similaires aux problèmes exemples pour lesquels la méthode est jugée validée.
- La **validation de performance empirique** : ici, deux choses sont faites. Premièrement, vérifier l'usage et la performance de la méthode pour certains problèmes exemples, en termes de la pertinence du résultat de la méthode et sa capacité de réduire les coûts et/ou le temps et/ou d'améliorer la qualité, entre plusieurs autres (le lecteur peut se référer au travail de [Phillips-Wren et al. \[2006\]](#) qui présentent un nombre important d'indicateurs de performance et de critères d'évaluation des i-DMSSs¹⁴⁷). La deuxième concerne la vérification que la solution du problème est un résultat de l'application de la méthode, ceci revient à quantitativement évaluer l'impact de la méthode en comparant les solutions avec et sans elle.
- La **validation de performance théorique** : il s'agit ici de prouver que la méthode est utile au-delà des exemples de problème testés.

8.3 Discussion de l'état de l'art sur les techniques de V&V : vers un cadre de V&V pour les IDSSs

Dans la section précédente, nous avons découvert quatre des méthodes de validation proposées dans la littérature, celles de [Landry et al. \[1983\]](#), [Finlay and Wilson \[1997\]](#), [Borenstein \[1998\]](#) et de [Seepersad et al. \[2006\]](#), en plus de la classification proposée par [Balci \[1997\]](#). **Toutes ces méthodes nous paraissent intéressantes pour une validation couvrant tous les aspects d'un IDSS**, bien que quelques types de validation soient sémantiquement redondants (comme par exemple la validation de performance empirique de [Seepersad et al. \[2006\]](#) et la validation prescriptive de [Borenstein \[1998\]](#) qui veulent dire la même chose). Par conséquent, nous définissons, de ces méthodes étudiées, le cadre de validation de la Figure 8.7, dont les activités de vérification et de validation, utilisées pour évaluer le IDSS proposé, sont brièvement décrites comme suit :

- **Validité des données d'étude** (validation structurelle empirique) : en vérifiant la pertinence des données utilisées lors de l'analyse de l'IDSS.
- **Validation conceptuelle** (validité apparente) : elle assure que les concepts présents dans le modèle conceptuel décrivent bien la problématique, où les spécifications sont représentées par la question de recherche. Sinon, des itérations de révision doivent avoir lieu jusqu'à ce que le modèle conceptuel et la problématique soient conformes.
- **Validation structurelle théorique** : pour s'assurer que l'assemblage des composants de l'IDSS proposé est accepté, et permet de répondre à la problématique.
- **Vérification** (Validation logique et V&V statique et dynamique des modules) : pour vérifier que le modèle logique est implémenté exactement comme décrit dans le modèle conceptuel. Ceci revient à vérifier et valider chacun des quatre module de l'IDSS proposé au fur et à mesure de leur développement. Nous considérons que la validation expérimentale est la même que la validation logique parce que l'interface de l'IDSS fait appel aux modèles formels (étant des traductions en langage informatique du modèle conceptuel) dont la validation est d'ordre logique.
- **Validité interne** : en testant si dans les mêmes conditions, le IDSS fournit la même solution. Ce type de validation permet de s'assurer que le système proposé est fiable, comme défini dans la Section 2.3.4.
- **Validité de l'IHM** : en vérifiant si elle est simple et user-friendly,

147. Ou Intelligent Decision-Making Support Systems, c'est une appellation équivalente à l'IDSS que nous utilisons.

- **Évaluation des utilisateurs potentiels** : ici, les utilisateurs finaux ou potentiels acceptent ou refusent la solution proposée par l'IDSS,
- **Validation de performance empirique** (Validations prescriptive et opérationnelle) : en vérifiant l'utilité du système pour le problème exemple et ses performances en termes de temps, coûts, etc.
- **Validation de performance théorique** : en étudiant les validations structurelle théorique, structurelle empirique et la validation de performance empirique (étapes 2, 4 et 5 de la Figure 8.7) lorsqu'il s'agit d'un nouveau problème non testé auparavant.
- **Capacité de répondre aux caractéristiques souhaitées** : dans ce type de validation, nous vérifions si l'IDSS a la capacité de se doter des caractéristiques définies dans la Section 2.3.4.

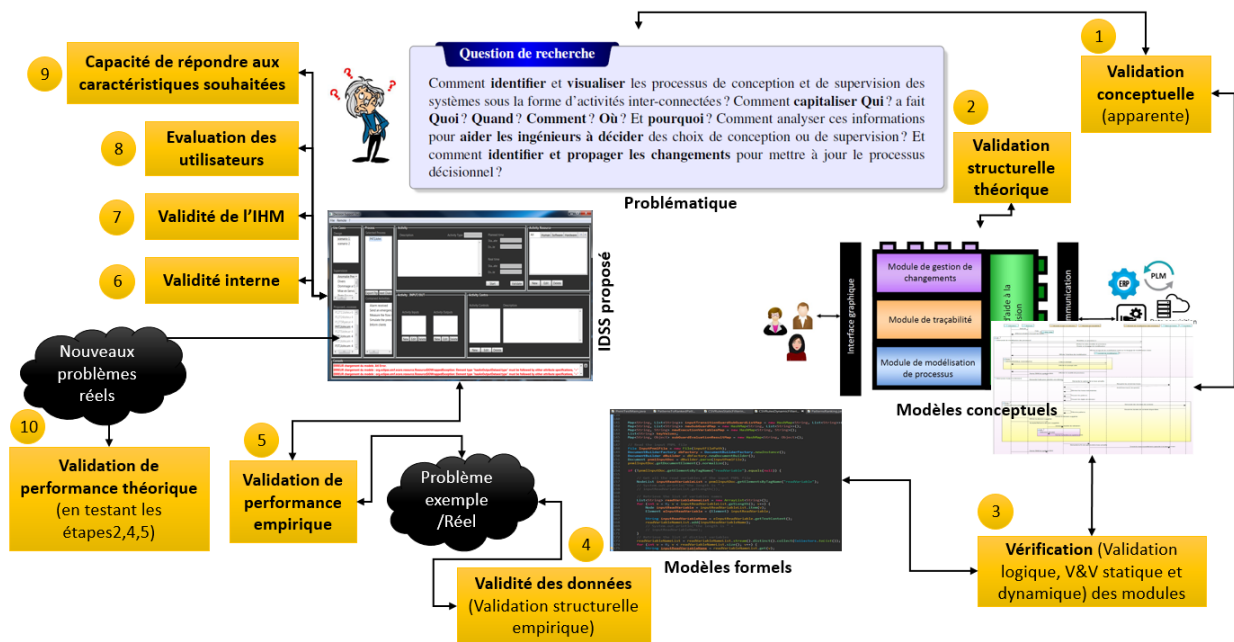


FIGURE 8.7 – Cadre de V&V que nous adoptons pour valider le IDSS proposé. Les activités de validation sont en jaune.

Maintenant que nous avons introduit le cadre de validation qui, à notre opinion, couvrira tous les aspects de l'IDSS, nous passons à la section suivante où chacun des types de validation considérés sera détaillé et où "GONTRAND", le projet à l'origine du besoin de concevoir un IDSS ayant les caractéristiques présentées dans la Section 2.3.4, est présenté.

8.4 Vérification et validation de l'IDSS proposé

8.4.1 Validité des données d'étude

Pour concevoir et implémenter notre IDSS, nous avons construit 1572 traces d'exécution du processus exemple de la Figure 7.22, en traçant les concepts du méta-modèle de trace de la Figure 6.15, et en suivant la logique présentée dans le Tableau 7.9 lors de la construction des traces. Le méta-modèle de trace identifie tous les concepts que nous jugeons nécessaires pour spécifier, d'une façon juste et pertinente, un processus quelconque dans le contexte de la conception ou la supervision des systèmes. De plus, le processus exemple que nous avons considéré contient toutes les jonctions possibles que nous pouvons rencontrer lors de l'exécution d'un processus industriel complexe; et nous considérons, en outre, la possibilité de présence des mêmes activités dans des patterns différents (prise en compte des chevauchements qui peuvent avoir lieu entre les patterns). Ceci est une raison pertinente pour valider les données que nous avons considérées pour développer notre IDSS.

Notons que nous venons de valider la **justesse**^a et la **pertinence**^b de la nature des données (traces) considérées et non pas leur **suffisance**. En effet, il est toujours conseillé de considérer un nombre beaucoup plus important de traces lorsqu'il s'agit d'utiliser les techniques de fouille (que ce soit de données ou de processus), ceci aidera l'IDSS à proposer la meilleure solution - tenant compte du contexte.

^a. En Anglais Data accuracy, il s'agit ici de vérifier que les données sont correctes (conformes à la réalité) et présentées d'une façon non-ambiguë : <http://etutorials.org/Misc/data+quality/>.

^b. En Anglais : Data relevance. Il s'agit ici de vérifier que la donnée aidera à prendre la bonne décision : <http://leadsift.com/how-to-determine-what-data-is-relevant-and-what-is-distracting/>. Ceci est vérifié parce que les données (traces) que nous considérons sont issues du méta-modèle de trace qui met en évidence les données les plus intéressantes à capturer lors de l'exécution d'un processus.

8.4.2 Validation conceptuelle et validation structurelle théorique

Pour répondre aux exigences exprimées dans la question de recherche de la Section 2.3.2, nous avons étudié l'architecture du système IDSS selon plusieurs points de vue. En effet :

- Le point de vue opérationnel nous a permis d'identifier l'interaction de l'IDSS avec son environnement, en termes de services.
- Le point de vue logique nous a permis d'identifier les fonctions que le système IDSS doit exécuter pour fournir les services, identifiés dans le point de vue opérationnel, à ses utilisateurs.
- Le point de vue physique nous a permis d'identifier les solutions de conception permettant d'implémenter les fonctions identifiées,
- Le point de vue processus et scénarios nous a permis de définir l'ordre d'exécution des fonctions pour répondre aux exigences énoncées dans la question de recherche.

Les points de vue logique, physique et processus nous ont permis de **clairement** identifier les composants du système IDSS - que nous avons appelés modules par la suite ; et chacun de ces modules répond à une des exigences, listées dans la question de recherche, comme le montre la Figure 8.8. De plus, ces modules sont assemblés comme nous l'avons présenté dans la Figure 1.1 et cet assemblage est le seul (**en ayant les mêmes composants**) qui a la capacité de répondre aux exigences de la question de recherche. En effet :

- Le méta-modèle de trace, qui surligne les concepts à tracer, est un **sous-ensemble** du méta-modèle de processus. Il est donc primordial d'identifier un méta-modèle de processus avant de commencer l'activité de la traçabilité,
- Les traces issues de l'activité de traçabilité sont ensuite vues comme une **ressource** par l'activité de l'aide à la décision, qui fouille dans ces traces, et propose à l'utilisateur la bonne décision compte tenu du contexte,
- L'aspect dynamique du contexte nous oblige à ajouter une autre activité dont le but est d'identifier et propager les changements contextuels en permettant ainsi de **mettre à jour** l'activité de l'aide à la décision.

Par conséquent, nous validons notre système IDSS au niveau conceptuel ainsi qu'au niveau structurel théorique.

8.4.3 Vérification et validité interne

Il s'agit ici de vérifier que chaque module implémente correctement ses spécifications :

Module de modélisation des processus et de traçabilité : Ici, a été développé le méta-modèle de processus qui doit être vérifié pour s'assurer qu'il ne viole pas la syntaxe du méta-méta-modèle appelé ECORE. Rappelons que la vérification syntaxique des méta-modèles (resp. Modèles) consiste à s'assurer que ces derniers sont conformes à leurs méta méta-modèles (resp. Méta-modèles). De plus, la vérification syntaxique des programmes informatiques implémentés en java, par exemple, consiste à s'assurer qu'ils respectent la syntaxe du langage Java. Ces deux dernières vérifications sont automatisées

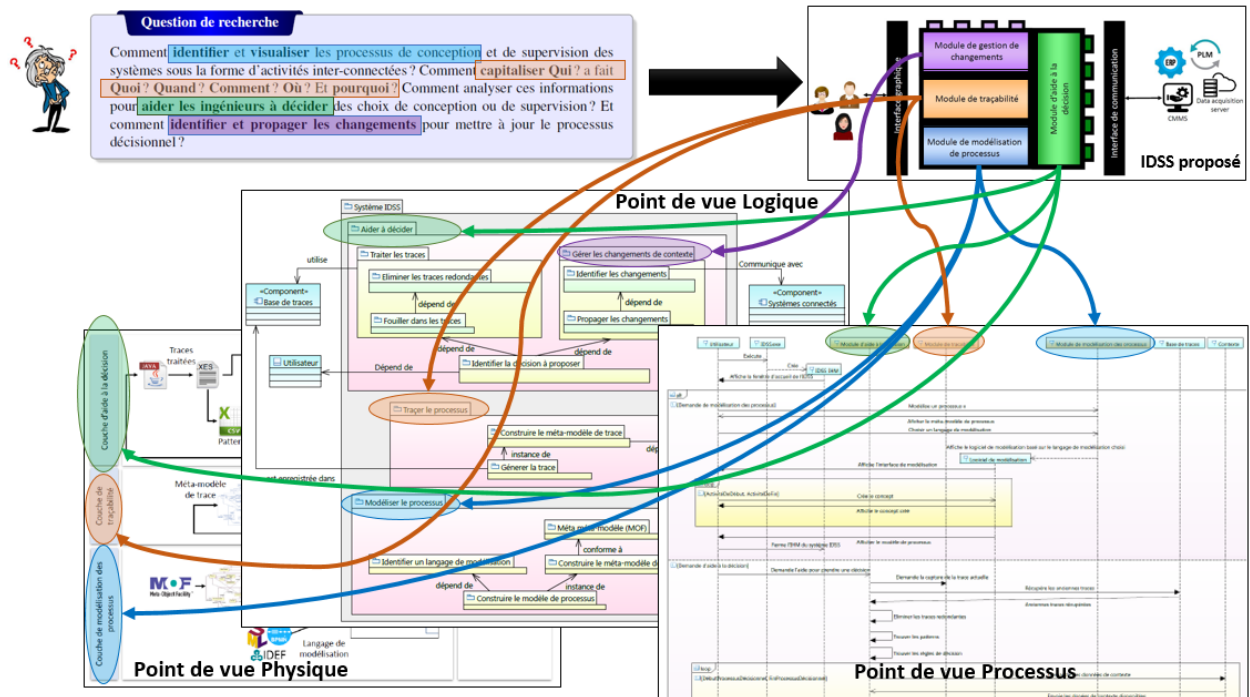
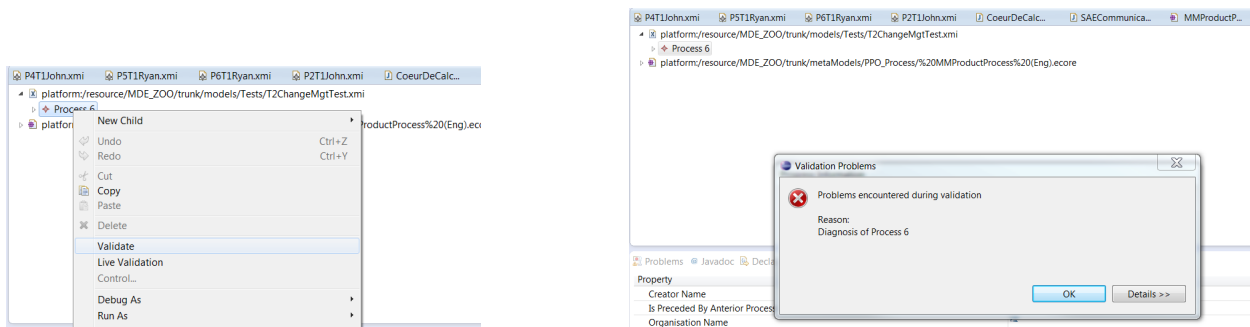


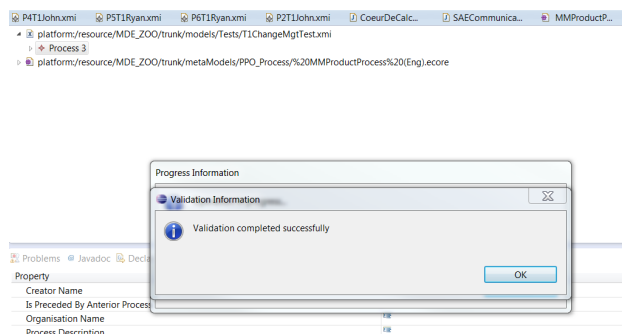
FIGURE 8.8 – Validation conceptuelle de l'IDSS proposé.

dans l'environnement Eclipse qui nous a permis de valider notre méta-modèle de processus (et donc de trace) comme le montre la Figure 8.9.



8.9.a. Possibilité de valider automatiquement les méta-modèles (resp. Modèles) dans l'environnement Eclipse

8.9.b. Exemple d'une validation échouée



8.9.c. Exemple d'une validation réussie

FIGURE 8.9 – La validation automatique des méta-modèles (resp. Modèles) dans l'environnement Eclipse.

Notons que la sémantique ne peut être validée par Eclipse, nous avons donc modélisé le processus exemple

de la Figure 7.22 comme le montre la Figure 8.10.a. Ce modèle textuel, qui est vu aussi comme une trace dans notre contexte dont le fichier XML est présenté dans la Figure 8.10.b, a été validé après l'avoir analysé. Lors de cette analyse, nous avons vérifié si les concepts que nous avons souhaité tracer (Qui a fait quoi, quand, comment, où et pourquoi) sont bien présents dans la trace.

Notons que l'utilisateur qui ne souhaite pas avoir un modèle textuel, peut choisir n'importe quel langage de modélisation et procéder à la modélisation graphique de son processus en respectant la sémantique de notre méta-modèle de processus.

The image shows two parts of a software development environment. On the left (8.10.a), a tree view displays a trace structure under 'Process RunningExample'. It includes a 'Planned Activity A0' with sub-activities like 'Human Resource Ryan', 'Software Resource Logiciel X', and 'Product Solution Space V1=12'. Below this, a table lists properties for 'Activity A0'.

Property	Value
Activity Description	C'est la première activité du processus exemple
Activity Event	start
Activity Name	A0
Activity State	available
Activity Type	decision
Is Followed By Next Activity	Planned Activity A1
Is Preceded By Anterior Activity	
Is Preceded By Sub Process	
Planned End Time	2016-11-08T11:00:00.000+0100
Planned Start Time	2016-11-08T10:00:00.000+0100
Real End Time	2016-11-08T11:07:00.000+0100
Real Start Time	2016-11-08T10:00:00.000+0100
Successor Gateway	xor

On the right (8.10.b), the XML code for the trace is shown, including namespace declarations, process metadata, and detailed activity information such as description, start/end times, and associated resources.

8.10.a. Trace générée en instanciant le méta-modèle de trace. Notons qu'il s'agit du pattern P_6 du Tableau 7.8

8.10.b. Fichier XML relatif à la trace de la Figure 8.10.a

FIGURE 8.10 – La trace générée en instanciant le méta-modèle de trace et le fichier XML correspondant.

Le moteur de traçabilité de l'activité de traçabilité est implémenté sous la forme d'un programme développé en JAVA. Pour vérifier ce dernier, nous avons considéré les techniques de V&V statiques, notamment l'analyse de la sémantique et de la syntaxe du programme, ainsi que les techniques dynamiques en testant les conditions et les boucles du programme (comme le montre la Figure 8.2). Après un nombre de révisions, le programme a été validé. Notons que l'idéal est de vérifier son code en adoptant les différentes techniques présentées dans la Figure 8.2. Nous nous sommes contentés de l'analyse de la sémantique du code et de son syntaxe ainsi que le test de ses conditions et ses boucles parce que cela nous paraît suffisant vu que les résultats des programmes informatiques sont conformes à leurs spécifications.

Modules de l'aide à la décision et de la gestion des changements : Dans ce module ont été développés les programmes "XMI2XES", "Pnmls2Pnml", "PNMLRulesPropagation", "Patterns2RankedPatterns", "StaticRulesFiltering" et "DynamicRulesFiltering", ainsi que le processus de l'aide à la décision de la Figure 7.39 et de gestion des changements de la Figure 7.44. Les programmes ont été développés en JAVA, et ont été vérifiés en utilisant les mêmes techniques statiques et dynamiques que celles utilisées lors de la vérification du moteur de traçabilité. Par conséquent, les programmes sont vérifiés syntaxiquement et sémantiquement ; et ont été, de plus, testés sur le processus de référence de la Figure 7.22 et ont produit le résultat que nous avons souhaité avoir (le lecteur peut se référer au Chapitre 7 pour plus de détails). Par conséquent, ces deux modules sont logiquement validés. De plus, leur validité interne est prouvée parce qu'ils ont produit les mêmes résultats après les avoir testés plusieurs fois dans les mêmes conditions.

8.4.4 Validité de l'IHM

L'IDSS proposé est doté d'une interface graphique interactive comme le montre la Figure 8.11. Cette interface est implémentée à partir des concepts du méta-modèle de processus proposé. En effet, plusieurs blocs sont présents dans l'interface, notamment les blocs :

- **Use Cases** : ce bloc concerne le contexte qui peut être soit la conception ou la supervision d'un système dont les traces de chacun des deux sont placées dans les sous-blocs "Design" (1 de la Figure 8.11) ou "Supervision" (2 de la Figure 8.11) selon la nature du scénario. Nous avons introduit la notion de scénario pour classer les processus afin de les organiser dans l'interface et permettre ainsi à l'utilisateur d'être plus à l'aise lors de la manipulation de l'IDSS. Les traces de chacun des scénarios de processus sont listées dans le sous-bloc "Proposed processes" (3 de la Figure 8.11).
- **Process** : ce bloc contient deux sous-blocs. Le "Selected Process" (4 de la Figure 8.11) a pour intérêt de dérouler les sous-processus du processus sélectionné, s'ils existent. Notons qu'il n'est pas possible d'intégrer les sous-processus parmi les activités parce que eux aussi contiennent des activités. Le sous-bloc "Contained Activities" (5 de la Figure 8.11) permet de lister les activités contenues dans le processus (ou sous-processus) sélectionné.
- **Activity** : quatre sous-blocs sont présents ici. Le premier étant le type de l'activité qui peut être soit une modélisation ou une décision, comme discuté dans la Section 6.1.5.1 (6 de la Figure 8.11). Le deuxième concerne la description de l'activité (7 de la Figure 8.11), le troisième concerne les temps planifiés et réels du début et de la fin de l'exécution de l'activité (8 de la Figure 8.11). Le quatrième bloc concerne les boutons qui commandent le début effectif et la validation (fin) de la nouvelle exécution de l'activité (9 de la Figure 8.11).
- **Activity Resources** : il s'agit ici des ressources matérielles, logicielles et humaines, entre autres, utilisées lors de l'exécution de l'activité (10 de de la Figure 8.11). Lors d'une nouvelle exécution, il est possibles d'ajouter, de modifier ou de supprimer une ressource, ceci justifie la présence des boutons "New", "Edit" et "Delete".
- **Activity Input/Output** : Ce sont les données d'entrée et de sortie de l'activité (11 de la Figure 8.11). Il est également possible d'ajouter, de modifier ou de supprimer une donnée d'entrée ou de sortie d'une activité.
- **Activity Control** : ce bloc concerne les contraintes qui contrôlent l'activité et leur description (12 de la Figure 8.11).
- **Console** : c'est la fenêtre à l'aide de laquelle le IDSS communique avec l'utilisateur, en signalant des erreurs et des avertissements, entre autres (13 de la Figure 8.11).

L'interface de l'IDSS proposée a été présentée au comité de pilotage et discutée lors de plusieurs réunions. Elle est supposée être simple d'utilisation parce qu'elle est conforme au méta-modèle de processus proposé où la liaison entre les différents blocs est explicitement définie. Cependant :

! L'évaluation de la **validité de l'interface**, qui consiste à calculer la fréquence de son utilisation, **ne sera réellement et objectivement assurée que lorsque le IDSS sera implémenté dans un environnement organisationnel réel.**

Avant de passer à la validation des utilisateurs et la validation de performance empirique, nous souhaitons introduire le projet industriel, où l'IDSS proposé sera intégré et exploité.

8.4.5 Le projet industriel ayant créé le besoin de concevoir un système IDSS

Contexte, objectifs et problématique : Pour mettre en œuvre les objectifs de la politique européenne de l'énergie¹⁴⁸, les systèmes énergétiques devront être capables d'assurer une production d'énergie renouvelable et décentralisée. Le développement d'une production décentralisée conduit à une multiplication considérable des points d'injection de l'énergie au niveau des réseaux de distribution,

148. https://europa.eu/european-union/topics/energy_fr

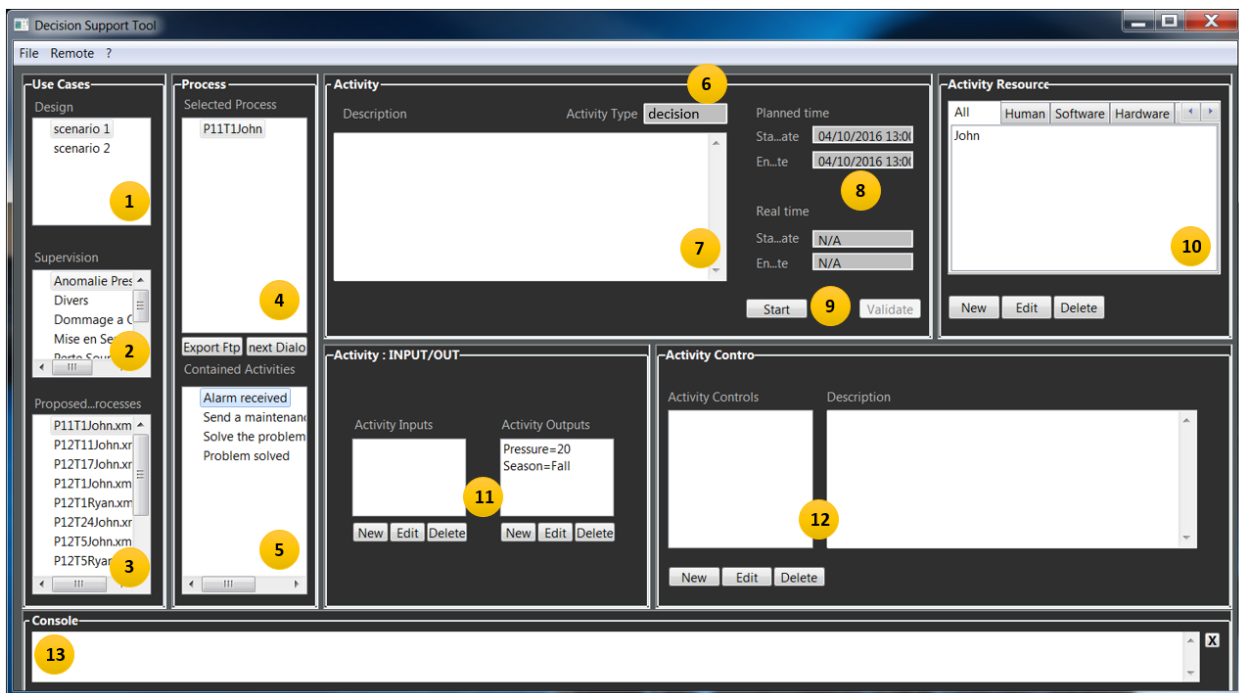


FIGURE 8.11 – Interface de l'IDSS proposé.

permettant ainsi de passer des réseaux où le gaz provenait uniquement des distributeurs nationaux aux réseaux où le gaz est injecté par de nombreux producteurs locaux de biométhane, évoluant ainsi vers un réseau de distribution de gaz moderne, connu sous le nom de "Smart Gas Grid". Ce concept peut être défini comme un ensemble de réseaux de distribution de gaz associés avec les technologies de l'information et de la communication récentes, dans lesquels est maximisée l'utilisation des énergies renouvelables en transformant le gaz en gaz vert (biométhane) à 100% renouvelable. En 2011, la commission européenne a tenté de définir ce concept émergent à travers trois fonctionnalités principales :

- L'intégration et augmentation de la part du gaz vert dans la consommation de gaz,
- L'augmentation de l'efficacité du réseau de distribution de gaz en renforçant davantage la sécurité industrielle et la qualité d'approvisionnement,
- L'intégration de technologies efficaces pour les clients.

L'intégration du biométhane dans de nombreux points de distribution fait que la qualité et la quantité du biométhane injecté peut ne pas être la même. Ceci relève le défi d'assurer l'équilibre entre les variations de consommation de gaz et la production de gaz vert pour maximiser leur intégration dans le réseau. Pour cela :

! Les **distributeurs de gaz** doivent disposer des **outils et moyens** nécessaires pour **connaître et piloter** (superviser) en **temps réel** la **quantité** et la **qualité** des gaz injectés en chaque point de distribution.

Le projet GONTRAND¹⁴⁹ (GestiON Temps réel d'un RéseAU National de Distribution) : Ce projet vise à développer les moyens permettant de connaître et piloter en temps réel la quantité et la qualité des gaz injectés en chaque point de distribution et de permettre d'augmenter la sécurité des réseaux en renforçant l'aide à la décision lors de la supervision du réseau et la gestion des incidents grâce à une meilleure connaissance de l'état du réseau et des expériences réussies dans le passé.

Cinq briques technologiques, inexistantes à ce jour et indispensables pour une conduite en temps réel efficace, doivent être développées dans le cadre du projet GONTRAND :

149. <https://www.gontrand.net/projet>

- Un **système d'aide à la décision** permettant d'aider au quotidien le superviseur dans les décisions relatives à la gestion des incidents, entre autres,
- Un **cœur de calcul** permettant de simuler l'état du réseau à travers les modèles de comportement dynamiques,
- Une **plateforme interopérable** dédiée au pilotage d'un réseau de distribution de gaz en permettant de visualiser et d'analyser son état,
- Une **solution Machine To Machine (M2M)** permettant de gérer les connectivités hétérogènes,
- Un **système d'analyse de gaz en temps réel** et in situ permettant d'analyser continuellement la composition du biométhane.

La conception du système IDSS, qui fait l'objet du présent travail de recherche, vient donc pour répondre aux besoins de la première brique technologique.

8.4.6 Évaluation des utilisateurs potentiels

Le projet GONTRAND est doté d'un comité de pilotage, composé d'un représentant de chacun des entreprises et laboratoires impliqués. Ce comité a pour mission d'imaginer un réseau d'exploitation de gaz fictif et de vérifier que le système IDSS proposé fournit la bonne décision au superviseur lorsqu'il est exploité lors de la supervision du réseau fictif conçu. Deux scénarios ont été identifiés, il s'agit de l'anomalie de pression où la pression chute ou augmente d'une façon anormale, et le dommage à ouvrage où un composant du réseau fictif (vanne, poste, etc.) est impacté.

Après plusieurs réunions avec le comité de pilotage, nous avons construit le modèle de processus, conformément à notre méta-modèle de processus, rassemblant les différents patterns possibles du premier scénario, c-à-d, qui sont exécutés dans le passé, lors de la gestion de l'alarme relative à une anomalie de pression dans le réseau fictif (Figure 8.12). À partir de ce modèle et conformément au méta-modèle de trace, nous avons créé les différents patterns (ayant le format XMI comme expliqué dans la Section 7.4.1) et nous les avons sauvegardés dans la base de traces, qui sera ensuite utilisée par le système IDSS lorsqu'il est exploité.

Au cours d'une nouvelle supervision du réseau de gaz fictif, et lorsqu'une alarme est reçue, le superviseur fait appel au système IDSS qui exécute toutes les étapes du processus de la Figure 7.39, et compte tenu du contexte, il propose la décision qu'il juge bonne sachant les règles de décision qu'il a identifiées (Figure 8.13), en exécutant tout le processus de la Figure 7.39.

Pour expliquer ceci, nous considérons le Tableau 8.1 qui présente les données de contexte aux instants t_1 , t_2 et t_3 relatifs aux trois jonctions Xor dans le processus de la Figure 8.12. Ces données sont acquises par le système via son interface de communication dont le protocole que nous avons choisi ¹⁵⁰ de considérer dans le cadre du projet GONTRAND, repose sur la communication via les sockets ¹⁰⁹. Par exemple, la commande "S:QVA@V-PDR-PL0-DEB-Qva1" veut dire que le système IDSS demande à l'hyperviseur la valeur du débit (Qva1) sur le poste "PL0". Notons que l'hyperviseur est un système où toutes les données venant des systèmes connectés dans le réseau, sont fédérées. De la même façon, le système IDSS communique avec le superviseur à travers sa couche de communication en lui suggérant des actions (les activités du pattern le plus significatif). Par exemple, la commande "S:DIN0!@AP001="compétence2 ", "positionPIS1" veut dire que le système IDSS propose au superviseur d'envoyer la "compétence 2" à la "positionPIS1".

À t_1 , le IDSS propose au superviseur prénommé "John" d'exécuter les activités du pattern "P4TIJohn" (Figure 8.14.a), parce que c'est le pattern qui satisfait le contexte à t_1 . Le superviseur exécute donc les activités "Envoyer un technicien d'urgence" et "Mesurer le débit" comme le montre le processus de la Figure 8.12. À t_2 , le IDSS se met à jour à l'aide du module de gestion dynamique des changements de variables de contexte, et voit que le pattern "P4TIJohn" satisfait toujours le contexte, donc propose au superviseur d'exécuter l'activité "Simuler la baisse de la pression de consigne" (Figure 8.14.a). À t_3 , et après la mise à jour du système IDSS, ce dernier voit que le pattern "P4TIJohn" ne satisfait plus le contexte, il propose donc au superviseur de continuer l'exécution de l'activité "baisser la pression de

150. Le choix d'utiliser ce protocole est fait par les entreprises et laboratoires impliqués dans le projet.

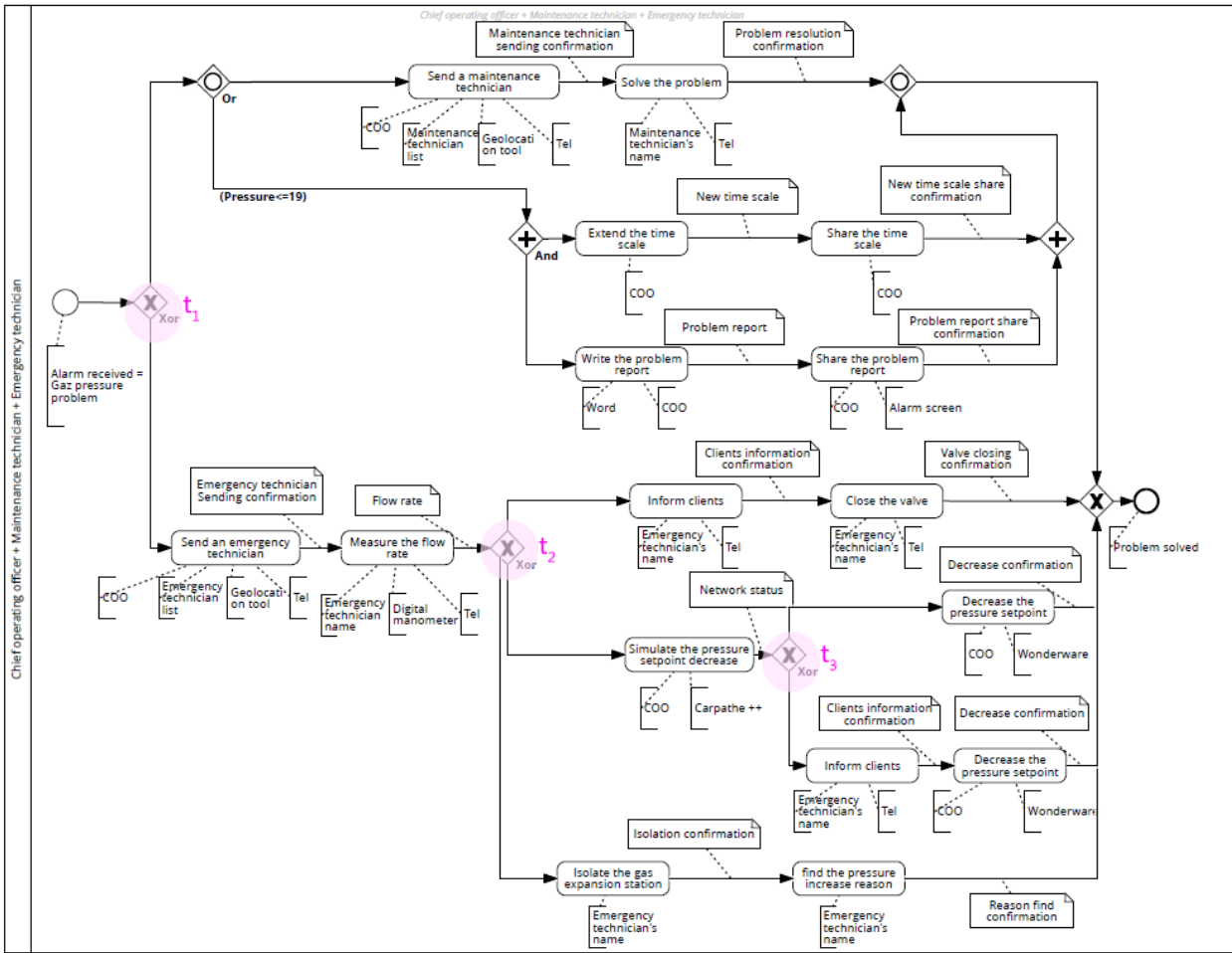


FIGURE 8.12 – Représentation BPMN du processus rassemblant tous les patterns exécutés lors de la gestion de l’anomalie de pression dans le réseau fictif. Notons qu’aucune règle de décision n’est connue a priori.

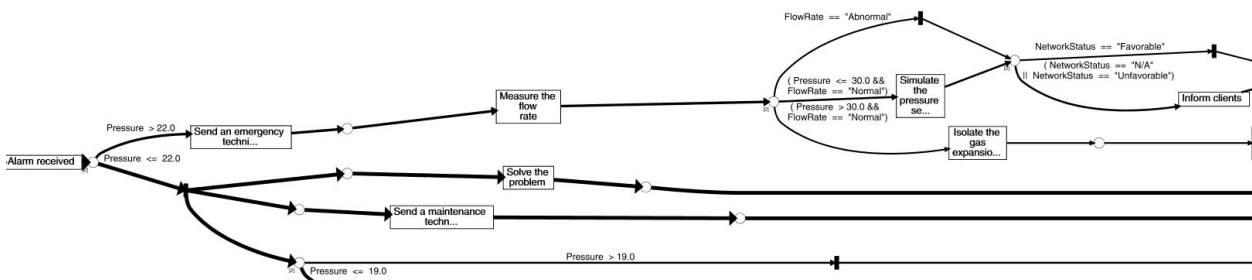


FIGURE 8.13 – Partie des règles de décision identifiées, contrôlant l’exécution des patterns de la Figure 8.12.

consigne” du pattern ”P3T1John” - qui suit la dernière activité qui a été exécutée dans le pattern ”P4T1John” (Figure 8.14.b).

	t_1	t_2	t_3
Superviseur	John	John	John
Saison	Hiver	Hiver	Hiver
Débit	Anormal	Normal	Normal
État du réseau	Défavorable	Défavorable	Favorable

TABLE 8.1 – Données contextuelles aux instants t_1 , t_2 et t_3 lors de la supervision du réseau de gaz virtuel.

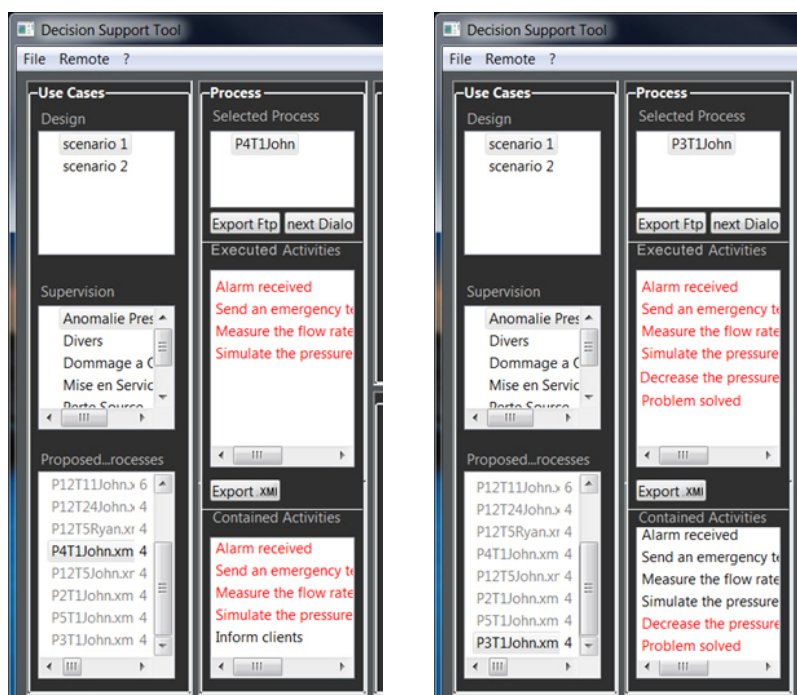
8.14.a. Décisions de t_1 à t_2 8.14.b. Décisions à t_3

FIGURE 8.14 – Le système IDSS dynamique : passage en temps réel d'un pattern à un autre lors de l'aide à la décision.

Le comité de pilotage du projet GONTRAND a accepté les décisions proposées par l'IDSS dans les contextes décisionnels étudiés lors de la supervision du réseau de gaz fictif.

8.4.7 Validation de performance empirique

Il s'agit ici de quantifier les performances de l'IDSS, lors de son exécution au cours de la supervision d'un réseau de gaz, en termes de sa capacité à réduire le temps pris pour décider, d'augmenter la qualité des décisions lors du traitement des alarmes, d'accroître la sécurité des réseaux de gaz supervisés, lever les problèmes industriels identifiés dans la Section 2.3.1, etc. Pour réaliser cette étude de performance empirique et évaluer l'impact du système proposé, il faudrait faire une comparaison entre de la décision prise sans et avec l'utilisation de l'IDSS. Pour que l'évaluation soit objective, **il est nécessaire que ce dernier soit entièrement intégré dans un environnement réel.**

8.4.8 Validation de performance théorique

Ce type de validation concerne l'aspect quantitatif des performances de l'IDSS proposé en évaluant la capacité de ce dernier à réduire le temps, les coûts et augmenter la qualité et accroître la sécurité, entre autres, au cours de l'exécution des processus exemples, autres que ceux que nous avons testés. Ceci **nécessite que l'IDSS soit appliqué sur des processus de supervision, autres que celui que nous avons étudié, et sur des processus de conception des systèmes.**

8.4.9 Capacité de répondre aux caractéristiques souhaitées

Il s'agit ici de tester si le système proposé est doté des caractéristiques que nous avons souhaitées avoir (et qui sont listées dans la Section 2.3.4). Le Tableau 8.2 présente le résultat de cette étude.

En effet, nous avons montré dans le Chapitre 7 que l'IDSS proposé, à travers ses modules, est capable de modéliser et tracer les processus en plus de l'aide à la décision, aux niveaux global et local (et donc orienté activité et processus). De plus, il est capable de s'adapter dynamiquement aux changements du contexte

	IDSS proposé
Capacité de modéliser les processus	✓
Capacité de tracer	✓
Capacité d'aider à décider	✓
Capacité à identifier et propager les changements contextuels	✓
Modulaire d'architecture	✓
Open source	✓
Générique	✓
Indépendant	✓
Semi-autonome/Intelligent	✓
Capacité de support	✓
Facile d'utilisation	✓
Extensible	✓
Maintenable	✓
Portable	✓
Efficient	non testé
Efficace	non testé
Exact	~
Fiable	✓
Orienté activité et processus	✓

TABLE 8.2 – Étude des caractéristiques du système IDSS proposé.

à l'aide du module de la gestion des changements contextuels. La décision proposée par le système IDSS (intelligent et semi-autonome) n'est pas automatiquement validée, c'est en effet à l'utilisateur de l'accepter (et donc de la valider) ou de la refuser; ceci évite d'imposer une décision, qui peut être incorrecte. De plus l'utilisateur est supporté à tous les niveaux et à n'importe quelle phase, parce que l'IDSS a la capacité de filtrer ses propositions selon la personne faisant partie du contexte de la décision.

De plus, l'IDSS proposé est open source et générique parce qu'il s'applique à tous les processus qui relèvent du contexte de la conception ou la supervision des systèmes. L'IDSS est aussi indépendant de toute infrastructure matérielle parce que son architecture est à 100% logicielle; le IDSS proposé est aussi indépendant de tout système d'information parce qu'il exécute ses fonctions sans s'appuyer sur d'autres systèmes, tout en permettant la communication avec eux via sa couche de communication où aucun standard de communication n'est imposé.

L'IDSS est simple d'utilisation et extensible. En effet, l'architecture modulaire permet de supprimer ou d'ajouter un module sans impacter le reste des modules (il est par exemple possible d'enlever le module de traçabilité des processus et connecter le système IDSS directement à une base de traces ayant le format XMI, comme le montre la Figure 8.15). Le système proposé est maintenable grâce à son architecture modulaire qui permet d'identifier les différents modules et aux programmes dont le code est visible et peut être amélioré. L'IDSS est portable parce qu'il n'est pas lié à une architecture matérielle spécifique, et il est fiable puisqu'il produit la même sortie en l'appliquant plusieurs fois dans le même contexte. Le système IDSS n'est par contre pas testé dans un environnement réel pour déterminer s'il est efficient et efficace. De plus, il n'est pas tout à fait exact parce qu'il n'adhère pas encore à **toutes** les caractéristiques spécifiées, notamment l'efficacité et l'efficience qui ne sont pas encore étudiées.

8.5 Synthèse et conclusion

Dans ce chapitre, nous avons essayé de vérifier et valider le système IDSS proposé. Pour ce, nous avons suivi le processus de la Figure 8.7. Il s'est avéré qu'il n'est pas possible de quantifier les performances de notre système parce que ceci nécessite que ce dernier soit entièrement implémenté dans un environnement

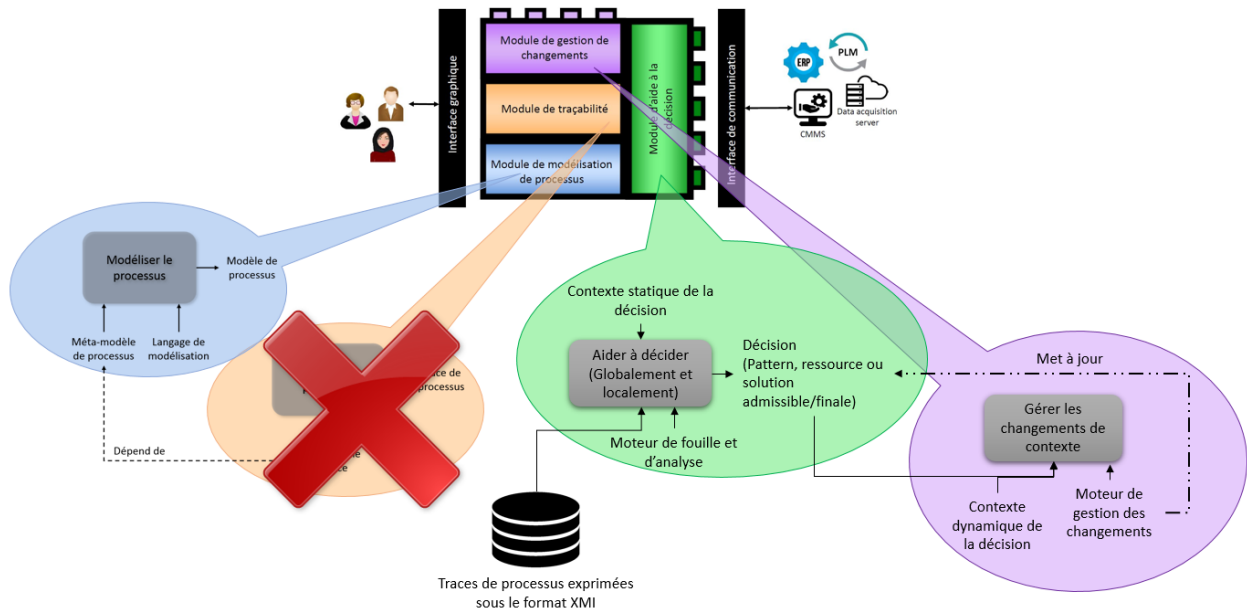


FIGURE 8.15 – Architecture modulaire du système IDSS : possibilité de remplacer les modules à condition que le module remplaçant ait les mêmes entrées/sorties que le module remplacé.

réel et exploité lors de l'exécution d'un processus industriel réel. notre système est donc entièrement vérifié mais à moitié validé, comme le montre le Tableau 8.3

	IDSS proposé
Validité des données d'étude	✓
Validation conceptuelle	✓
Validation structurelle théorique	✓
Vérification	✓
Validité interne	✓
Validité de l'IHM	~
Évaluation des utilisateurs potentiels	✓
Validation de performance empirique	non testé
Validation de performance théorique	non testé
Capacité de répondre aux caractéristiques souhaitées	~

TABLE 8.3 – Récapitulatif du résultat de la V&V du système IDSS proposé.

Conclusion générale et perspectives

"The true function of philosophy is to educate us in the principles of reasoning and not to put an end to further reasoning by the introduction of fixed conclusions."

George Henry Lewes

9.1 Synthèse générale	202
9.1.1 Attendez ! Mais pourquoi devons nous proposer un système IDSS ?	202
9.1.2 D'accord, et comment nous avons procédé à la proposition du système IDSS ?	202
9.1.3 Parfait ! Mais a-t-il été vérifié cet IDSS ? Quelles sont ses limites ?	203
9.2 Vers de nouvelles perspectives	204
9.2.1 Perspectives à court terme	204
9.2.2 Perspectives à moyen terme	205
9.2.3 Perspectives à long terme	205
9.3 Conclusion de la conclusion	207

Vous croyez que la conclusion d'un travail de recherche est une conclusion au vrai sens du terme ? Si c'était le cas, on se serait, au XXI^{ème} siècle, limité aux "locomotives à vapeur", aux ordinateurs "ABC" (Atanasoff-Berry Computer), à la théorie de la "terre plate". On aurait communiqué à distance via les "pigeons voyageurs", on se serait opéré sans "anesthésie générale" et on aurait prié pour que nos enfants ne meurent pas de la "tuberculose" ou de la "peste". La nature curieuse et persévérante de l'être humain l'a toujours poussé à progresser continuellement, en considérant la conclusion d'un travail de recherche donné, comme une introduction ouvrant les portes vers de nouveaux défis.

Cette conclusion, qui n'est certes qu'une synthèse générale du présent travail de recherche, vise à présenter un bilan sur ce qui a été fait. Notamment, pourquoi il a été fait et avec quelles hypothèses ? Quels sont les résultats obtenus et comment ils le sont ? Quelles sont les limites de la proposition ? Et quelles sont les perspectives à court, long et moyen termes, visant à lever les limites identifiées ?

9.1 Synthèse générale

9.1.1 Attendez! Mais pourquoi devons nous proposer un système IDSS ?

La problématique : Le contexte économique actuel, ainsi que l'évolution technologique qui marque cette décennie, obligent les entreprises à être férocement compétitives pour continuer d'exister dans le marché. Atteindre la compétitivité repose essentiellement sur la maîtrise des processus industriels, notamment la conception et la supervision des systèmes étant les plus critiques parmi les autres processus de développement des systèmes. En effet, les retours industriels montrent que plusieurs problèmes sont rencontrés lors de la conception ou la supervision d'un système. Ces problèmes peuvent être classés en trois grandes catégories, la première concerne les difficultés rencontrées lors de la recherche et l'échange de l'information. La deuxième concerne les difficultés lors de la prise de décision et la troisième concerne les difficultés lors de l'identification et la propagation des changements contextuels au cours de la prise de décision.

Les objectifs scientifiques : Pour lever les difficultés identifiées, nous nous sommes fixés comme objectifs, la réponse à la question qui constitue le barycentre de notre recherche : *"Comment identifier et visualiser les processus de conception et de supervision des systèmes sous la forme d'activités inter-connectées? Comment capitaliser Qui? a fait Quoi? Quand? Comment? Où? Et Pourquoi? Comment analyser ces informations pour aider les ingénieurs à décider des choix de conception ou de supervision? Et comment identifier et propager les changements pour mettre à jour le processus décisionnel?"*

9.1.2 D'accord, et comment nous avons procédé à la proposition du système IDSS ?

L'hypothèse : Nous avons considéré qu'en mettant la casquette d'un analyste de données et de processus, nous arriverons à résoudre la problématique industrielle identifiée. En effet, nous avons supposé que le couplage des techniques de modélisation de processus et de la traçabilité a le pouvoir de lever la première catégorie de problèmes. En outre, nous avons supposé que le couplage des techniques de fouille de données et de fouille de processus permettent de résoudre les problèmes de la deuxième et troisième catégories.

L'approche : Pour aider les ingénieurs à décider, nous avons proposé un système d'aide à la décision intelligent (IDSS) et modulaire, dont chacun des quatre modules exécute une des fonctions suivantes :

- **Modéliser le processus :** il s'agit ici d'assister l'ingénieur lors de la modélisation d'un processus industriel. Cet ingénieur peut ne pas savoir exactement quoi modéliser parce qu'il est non-expérimenté ou parce qu'il est perdu à cause de la complexité des processus. Nous proposons donc un méta-modèle de processus qui regroupe tous les concepts les plus intéressants à modéliser dans le contexte de la conception ou la supervision d'un système, ainsi que les relations entre eux. L'ingénieur peut donc choisir n'importe quel langage de modélisation comme IDEF0 ou BPMN, entre autres, et procède à la modélisation graphique de son processus en ayant notre méta-modèle sous les yeux.
- **Tracer le processus :** De même, l'ingénieur peut ne pas savoir quoi tracer au juste lors de l'exécution d'un processus. Nous avons donc proposer un méta-modèle de trace, qui est le méta-modèle de processus lui même, permettant de guider l'ingénieur ou le système de traçabilité. Le choix de considérer le méta-modèle de processus comme méta-modèle de trace vient du fait qu'il soit intéressant de tout tracer parce que tous les concepts sont intéressants pour avoir une idée claire sur comment le processus est réellement exécuté. L'ingénieur peut ensuite adapter la trace selon ses besoins.
- **Aider à décider :** En ayant les anciennes traces issues du module de traçabilité, ou de n'importe quel système de traçabilité - mais à conditions que les traces aient le format XMI, le module de l'aide à la décision les utilise comme matière pour la fouille. En effet, ce module est composé de deux autres sous-modules, le premier concerne l'aide à la décision globale et permet, en utilisant les

techniques de la fouille des processus, de proposer à l'utilisateur le pattern (une exécution possible du processus de son début à sa fin) le plus significatif compte tenu du contexte de la décision, ceci passe par trois étapes globales : l'identification des différents patterns, l'identification des règles de décision en analysant les anciennes traces, et le filtrage des patterns par rapport aux règles identifiées. Le deuxième sous-module est dédié à l'aide à la décision locale et permet de prédire les paramètres (ressources et donnée de sortie) de chaque activité du pattern choisi. Ainsi, parmi les processus $P_1 = \{A_1, A_2, A_3\}$ et $P_2 = \{A_4, A_5, A_6\}$, et les ressources R_1, R_2 et R_3 , il est possible - en tenant compte du contexte C - de proposer à l'utilisateur d'exécuter le pattern P_1 et d'utiliser la ressource R_1 pour A_1 et A_2 et R_3 pour A_3 .

- **Gérer les changements de contexte** : Le contexte, étant la contrainte principale du module d'aide à la décision, peut changer dans le temps. En effet, la valeur de la donnée d , mesurée ou calculée à t_1 peut être différente de la valeur de la même donnée, mesurée à t_2 , ceci amène le IDSS à proposer un pattern qui ne sera plus valide à un certain moment lors de son exécution, en d'autres mots, le pattern proposé ne satisfera plus les règles de décision à un instant t_i . Pour pallier ce problème, a été introduit le module de la gestion des changements, qui a pour but de mettre à jour le module d'aide à la décision en identifiant les changements au niveau des données contextuelles, et en filtrant les patterns à l'instant correspondant à l'identification de ces changements. Le système IDSS peut ainsi passer dynamiquement d'un pattern, qui ne satisfait plus les règles, à un autre qui les satisfait.

9.1.3 Parfait ! Mais a-t-il été vérifié cet IDSS ? Quelles sont ses limites ?

Les résultats de la V&V : Pour s'assurer que le système proposé adhère à toutes les exigences qu'on a listées dans la question de recherche ainsi qu'à toutes les caractéristiques que nous avons souhaitées avoir dans un système IDSS, nous avons procédé à la vérification et la validation de notre système selon plusieurs perspectives :

- La **validation des données d'études** : pour développer le système IDSS et valider son comportement, nous l'avons appliqué sur 1572 traces construites à la main en instanciant le méta-modèle de trace que nous avons proposé. Les données d'études (traces et processus considérés) sont validées parce qu'elles sont justes et pertinentes. En effet, elles sont jugées être justes parce qu'elles sont conformes à la réalité en décrivant un processus réel contenant les différentes jonctions possibles. De plus, elles sont jugées être pertinentes parce que les traces viennent d'un méta-modèle où les concepts les plus intéressants sont surlignés. Il est, en revanche, toujours intéressant d'avoir une base de données très riche pour que le système IDSS (et n'importe quel système basé sur les techniques de la fouille) soit précis.
- La **validation conceptuelle** avait pour but d'assurer que les concepts présents dans les modèles conceptuels permettent une description exhaustive de la problématique. Ceci a bien été le cas, et le système a donc été conceptuellement validé.
- La **validation structurelle théorique** avait pour objectif d'assurer que les modules du système IDSS sont architecturés de façon à ce que le comportement du système soit conforme au comportement attendu. Ceci a également été le cas, et le système est donc validé dans cette perspective.
- La **vérification** avait pour but de s'assurer que les modèles logiques sont implémentés exactement comme préconisé par les modèles conceptuels. Le système a été **vérifié** et **validé** dans cette perspective.
- La **validation interne** a pour objectif de s'assurer que le système proposé se comporte de la même façon lorsqu'il est exécuté plusieurs fois dans les mêmes conditions. Ceci a été le cas, parce que l'identification des patterns et le filtrage statique et dynamiques sont basés sur des techniques déterministes. Le système est par conséquent validé dans cette perspective.
- La **validation de l'interface graphique du système IDSS** : L'interface homme machine du système IDSS est simple d'utilisation parce que tous ses blocs, ainsi que les relations entre leurs éléments, sont explicitement décrits dans le méta-modèle de processus que nous proposons, mais ne peut être

objectivement validée avant que le système soit réellement implémenté et utilisé lors de l'exécution d'un processus réel.

- La **validation des utilisateurs potentiels** : il s'agissait ici de tester le système IDSS en l'utilisant lors de la supervision d'un réseau de gaz fictif, conçu par le comité de pilotage du projet GONTRAND, qui vise la supervision intelligente et en temps réel des réseaux de gaz. Le comité a validé les résultats proposés par le IDSS qui a permis d'assister les superviseurs dans leurs processus décisionnels dans un environnement marqué par la fréquence de changements des variables de contexte. L'IDSS est donc validé dans cette perspective.
- Les **validations de performance empirique et de performance théorique** : Le système proposé ne peut pas être validé dans ces deux perspectives. Il faut, en effet, l'implémenter et l'intégrer dans un environnement organisationnel réel pour assurer la validation de performance empirique. De plus, il faut qu'il soit appliqué lors de l'exécution d'un exemple de processus, autre que celui que nous avons testé, pour assurer la validation de performance théorique.
- La **capacité de répondre aux caractéristiques souhaitées** : Nous avons montré que l'IDSS n'a pas toutes les caractéristiques que nous avons souhaitées avoir. En effet, L'efficacité et l'efficacité du système n'ont pas pu être testées parce que le système doit être implémenté dans un milieu industriel réel. De plus, il n'est pas tout à fait exact parce qu'il n'adhère pas encore à toutes les caractéristiques spécifiées, notamment l'efficacité et l'efficacité.

Les limites de l'IDSS : Aucun travail de recherche n'est parfait. Le système IDSS proposé a, en effet, trois limites principales :

- La première concerne les traces d'entrée de l'IDSS qui **doivent** être exprimées sous le format XMI (XML Metadata Interchange). Cette contrainte limite l'utilisation de l'IDSS, parce le format des traces peut varier d'un système de traçabilité à l'autre, si l'utilisateur choisit de ne pas utiliser le moteur de traçabilité proposé.
- La deuxième limite concerne l'automatisation de l'activité de génération des règles de décision de la Figure 7.39, qui n'est pas encore assurée. En effet, il n'est pas encore possible d'appeler le programme JAVA, relatif à l'algorithme MPE qui permet la génération des règles de décision, en dehors de l'environnement ProM. En d'autres mots, il faut passer par l'interface graphique de ProM pour pouvoir exécuter l'algorithme MPE. La génération des règles de décision est par conséquent manuelle à l'état présent.
- Nous avons choisi de concevoir un système IDSS modulaire, parce que cette architecture permet l'utilisation des modules pour d'autres fins, comme par exemple lors du développement d'un autre système d'aide à la décision. La troisième limite concerne le champ d'utilisation du module de la gestion des changements, qui ne peut mettre à jour que les systèmes d'aide à la décision basés sur les techniques de fouille de processus parce qu'il se base sur les fichiers de règles de décision exprimés sous le format PNML.

9.2 Vers de nouvelles perspectives

Tout système proposé ouvre la porte vers de nouveaux défis. Ces défis peuvent être relevés à court, moyen ou long terme, selon leur nature.

9.2.1 Perspectives à court terme

Il s'agit ici des perspectives relatives au développement du système IDSS et qui doivent immédiatement et impérativement être abordées pour que le système soit entièrement opérationnel dans un milieu industriel réel.

- La première perspective concerne l'automatisation de la génération des règles de décision en cherchant une méthode permettant l'appel du programme JAVA de l'algorithme MPE, indépendamment de l'environnement ProM, c-à-d, sans passer par son interface graphique.

- La deuxième perspective à court terme concerne la comparaison de performance des algorithmes de la classification supervisée afin de choisir celui qui permet une meilleure prédiction des paramètres des activités.
- La troisième perspective concerne l'implémentation de l'algorithme relatif au processus de l'aide à la décision locale.

9.2.2 Perspectives à moyen terme

Il s'agit ici des perspectives qui doivent être abordées dans le futur proche pour permettre à l'IDSS d'être opérationnel pour tout type de processus métiers relatifs à la conception ou la supervision d'un système :

- Lors de l'exécution d'un processus donné, il est possible d'exécuter plusieurs fois la même activité dans le même pattern; un constructeur automobile peut par exemple répéter l'activité de polissage jusqu'à l'obtention d'une finition jugée parfaite. La première perspective concerne, par conséquent, la prise en compte des itérations (loops) au niveau de l'exécution des activités. En d'autres termes, il faut développer une méthode permettant d'identifier la règle contrôlant l'exécution multiple d'une activité. Pour expliquer davantage cette perspective, nous considérons le processus exemple de la Figure 9.1.a, qui est une extension du processus de la ??, en ajoutant 20 traces différentes à notre journal des événements initial, où l'exécution de l'activité A_{13} se répète deux fois si ($V_1 > 54$).

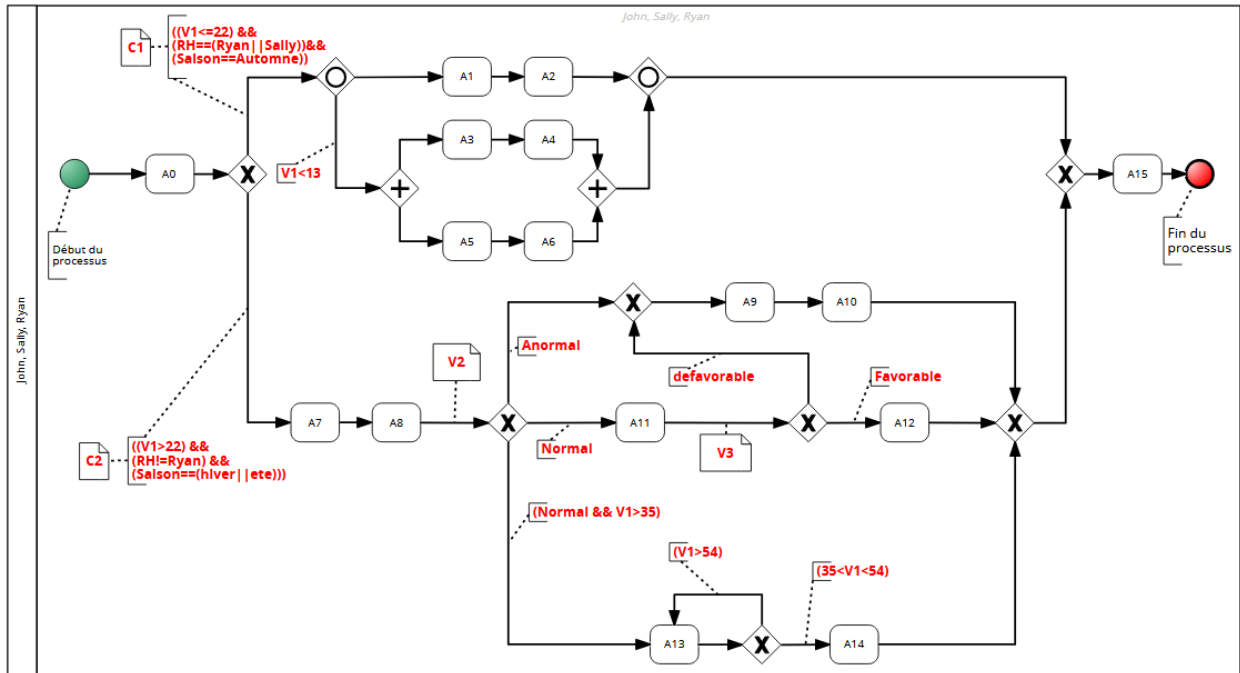
Nous avons déjà identifié la condition relative à la variable V_1 qui contrôle l'exécution du pattern $P_4 = \{A_0, A_7, A_8, A_{13}, A_{14}, A_{15}\}$, comme le montre le Tableau 7.8. En effet, le pattern P_4 s'exécute si ($V_1 > 35$); si en plus ($V_1 > 54$) le pattern P_4 s'exécute en exécutant deux fois l'activité A_{13} , c-à-d, $P_4 = \{A_0, A_7, A_8, A_{13}, A_{13}, A_{14}, A_{15}\}$. La règle générée, en tenant compte des 20 nouvelles traces, ne tient pas compte de cette exécution multiple de A_{13} comme le montre la Figure 9.1.b. Ici, les algorithmes IM et MPE ont réussi à identifier l'itération au niveau de A_{13} mais n'ont pas été capables d'identifier la règle contrôlant cette itération. En effet, le chemin et la règle en bleu sont relatifs à l'exécution de P_4 , comme a déjà été discuté, et la jonction en rouge montre qu'à ce stade, il est possible d'exécuter A_{14} (chemin vert sur la Figure 9.1.b) ou de revenir pour exécuter A_{13} (chemin rouge sur la Figure 9.1.b), mais aucune règle n'est identifiée.

- La deuxième perspective est d'intégrer l'IDSS dans un environnement industriel réel et de le valider après avoir évalué ses performances (Phillips-Wren et al. [2006] proposent un nombre important d'indicateurs de performance).
- La troisième perspective est de classer les patterns selon leur performance et non pas juste selon leur fréquence. En d'autres mots, proposer à l'utilisateur le pattern le plus performant, le plus fréquent et qui satisfait les règles de décision et le contexte. De plus, il serait très intéressant de calculer le risque de la décision avant qu'elle soit prise ainsi que l'impact de l'incertitude sur la décision, et de les lier à la performance du pattern pour augmenter la précision du système IDSS.
- La quatrième perspective consiste à développer une méthode permettant de tenir compte des justifications des utilisateurs, comme présenté dans le méta-modèle de processus, lors de la fouille des processus et des données, de façon à ce que ces dernières contrôlent le filtrage des patterns. Par exemple, entre les deux patterns P_1 et P_2 qui satisfont tous les deux la condition C_1 et qui sont tous les deux fréquents (les deux sont exécutés 1000 fois par exemple), l'IDSS propose le pattern P_1 parce qu'il est **plus agréable à exécuter**.

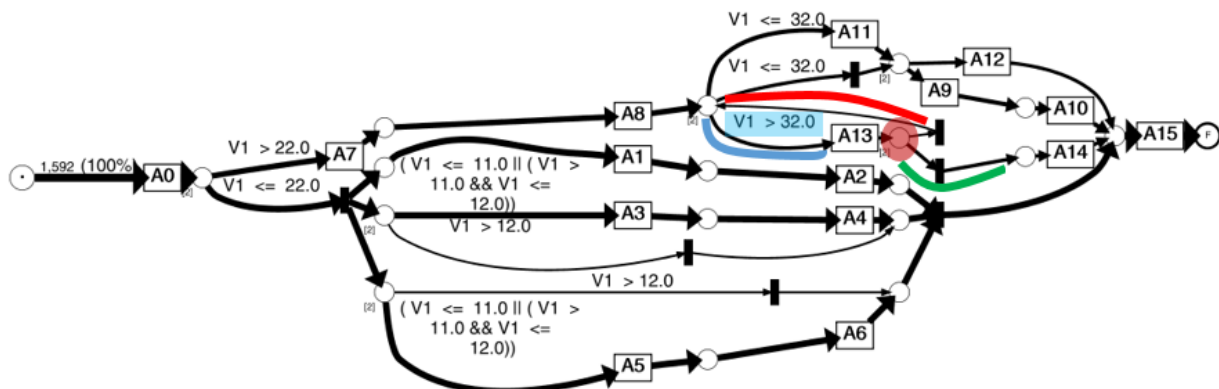
9.2.3 Perspectives à long terme

Ce sont des perspectives qui doivent préférablement être abordées pour adhérer à la logique de la méthode scientifique. Quatre perspectives à long termes sont identifiées :

- La première concerne l'architecture du système IDSS. Il serait intéressant de développer plusieurs architectures candidates du système IDSS et de calculer la performance de chacune d'elles pour choisir la plus performante.



9.1.a. Processus exemple avec le présence de l'exécution multiple des activités (ici, c'est l'activité A_{13} qui s'exécute plusieurs fois si $(V_1 > 54)$.)



9.1.b. Règle de décision relative à la variable de contexte V_1 identifiée à partir des patterns du processus de la Figure 9.1.a

FIGURE 9.1 – Incapacité d'identifier les règles relatives à l'exécution multiple des activités.

- La deuxième perspective concerne le développement d'un nouveau langage de modélisation qui sera conforme à notre méta-modèle de processus. Ceci aidera les utilisateurs à ne pas se limiter à un modèle textuel. En effet, ce dernier est le seul moyen pour modéliser tous les concepts comme nous les avons définis. Le langage BPMN, par exemple, ne permet pas de modéliser tous nos concepts parce que son méta-modèle ne les comprend pas, l'utilisateur se trouve donc obligé à modéliser nos concepts comme des éléments de données ou des commentaires BPMN.
- La troisième perspective consiste à tester si l'utilisation des autres techniques de fouille de données, notamment la fouille des séquences et des règles d'association, donnera lieu à un IDSS plus précis et plus performant que le notre (après avoir étudié la performance de l'IDSS proposé, bien sûr).
- La quatrième perspective concerne l'utilisation des techniques de l'intelligence augmentée, qui pourront être plus performantes que les techniques de l'intelligence artificielle, ces dernières qui dépendent fortement de la base de données à partir de laquelle les algorithmes de la fouille sont entraînés.

9.3 Conclusion de la conclusion

Dans ce travail de recherche, nous avons levé trois verrous scientifiques et trois verrous technologiques :

- **Verrous scientifiques :**
 - Comment répondre à la question de recherche de la Section 2.3.2 ?
 - Comment obtenir des règles de décision précises en utilisant l’algorithme MPE (Multi-Perspective Explorer), sans modifier le journal des événements ?
 - Comment mettre à jour le système IDSS pour qu’il tienne compte de l’aspect dynamique du contexte de décision ?
- **Verrous technologiques :**
 - Comment passer d’un ensemble de traces XMI à un seul fichier XES ¹²⁷ ?
 - Comment passer d’un ensemble de fichiers CSV où chacun contient la variante du pattern et ses doublons à un seul fichier CSV ne contenant que les variantes des patterns et leur fréquence ?
 - Comment propager les règles de décision à travers les activités invisibles ¹⁴¹ ?

Nous avons ensuite vérifié et étudié la validité de notre système proposé avant de conclure le manuscrit en listant quelques perspectives pouvant faire l’objet d’un travail de thèse futur. En arrivant à la fin, nous espérons que vous avez passé un moment agréable en lisant le présent manuscrit de thèse.

*Widad Es-Soufi
Esma Yahia
Lionel Roucoules*

Quatrième partie

BIBLIOGRAPHIE, ANNEXE, LISTE DES PUBLICATIONS ET DES NOTES

Liste des publications

Journaux internationaux

1. Lionel Roucoules, Esma Yahia, Widad Es-Soufi, and Serge Tichkiewitch. Engineering design memory for design rationale and change management toward innovation. *CIRP Annals - Manufacturing Technology*, 65(1) :193 – 196, 2016.
2. Widad Es-Soufi, Esma Yahia and Lionel Roucoules. An Intelligent Decision-Making Tool to Empower Systems Design and Supervision. *In submission*, X(X) :XX – XX, 2019.

Conférences internationales

1. Widad Es-Soufi, Esma Yahia, and Lionel Roucoules. On the use of process mining and machine learning to support decision making in systems design. In *Product Lifecycle Management for Digital Transformation of Industries - 13th IFIP WG 5.1 International Conference, PLM 2016, Columbia, SC, USA, July 11-13, 2016, Revised Selected Papers*, pages 56–66, 2016.
2. Widad Es-Soufi, Esma Yahia, and Lionel Roucoules. *Collaborative Design and Supervision Processes Meta-Model for Rationale Capitalization*, pages 1123–1130. Springer International Publishing, Cham, 2017.
3. Widad Es-Soufi, Esma Yahia, and Lionel Roucoules. *A Process Mining Based Approach to Support Decision Making*, pages 264–274. Springer International Publishing, Cham, 2017.
4. Widad Es-Soufi, Esma Yahia, and Lionel Roucoules. A dynamic contextual change management application for real time decision-making support. In *Product Lifecycle Management (article soumis)*, 2018.

Bibliographie

- Acosta, G., González, C. A., and Pulido, B. (2001). Basic tasks for knowledge-based supervision in process control. *Engineering Applications of Artificial Intelligence*, 14(4) :441 – 455. (page 20).
- Adams, R. J., Advanced Aviation Concepts, I., and United States. (1993). *How expert pilots think : cognitive processes in expert decision making*. Number xiii, 69 p. Federal Aviation Administration, Research and Development Service; Available through the National Technical Information Service, Washington, D.C. : Springfield, Va. (page 69).
- Adomavicius, G. and Zhang, J. (2012). Impact of data characteristics on recommender systems performance. *ACM Trans. Manage. Inf. Syst.*, 3(1) :3 :1–3 :17. (page 50).
- AFIS (2012). *Découvrir et Comprendre l'Ingénierie Système*. Collection AFIS. Edition Cépaduès. (pages 91, 92, 93).
- Aggarwal, C. C. et al. (2016). *Recommender systems*. Springer. (pages 47, 48, 49).
- Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2) :207–216. (pages 135, 147).
- Agrawal, R. and Srikant, R. (1995). Mining sequential patterns. In *Data Engineering, 1995. Proceedings of the Eleventh International Conference on*, pages 3–14. IEEE. (pages xxi, 135, 136).
- Ahmed, S. and Hansen, C. T. (2002). A decision-making model for engineering designers. In *Proceedings Engineering Design Conference (EDC'02), King's College, London, UK*, pages 217–228. (page 83).
- Albalade, A. and Minker, W. (2013). *Semi-Supervised and Unsupervised Machine Learning : Novel Strategies*. ISTE. Wiley. (pages 138, 140).
- Alberts, C., Dorofee, A., Higuera, R., Murphy, R., Walker, J., and Williams, R. (1996). *Continuous Risk Management Guidebook*. Software Engineering Institute, Carnegie Mellon University. (page 63).
- Alique, A., Haber, R. E., Haber, R. H., Ros, S., and Gonzalez, C. (2000). A neural network-based model for the prediction of cutting force in milling process. a progress study on a real case. In *Intelligent Control, 2000. Proceedings of the 2000 IEEE International Symposium on*, pages 121–125. IEEE. (page 42).
- Alter, S. (1980). *Decision support systems : current practice and continuing challenges*. Addison-Wesley series on decision support. Addison-Wesley Pub. (pages xxi, 50, 51).
- Alting, L. (1995). Life cycle engineering and design. *CIRP Annals*, 44(2) :569 – 580. (page 38).
- Altshuller, G. S. (1999). *The innovation algorithm : TRIZ, systematic innovation and technical creativity*. Technical Innovation Center, Inc. (pages 38, 39).
- Anand, S. S. and Mobasher, B. (2003). Intelligent techniques for web personalization. In *Proceedings of the 2003 international conference on Intelligent Techniques for Web Personalization*, pages 1–36. Springer-Verlag. (page 47).

- Anderson, D. (2001). *Design for manufacturability : optimizing cost, quality, and time to market*. CIM Press. (page 20).
- Andreasen, M. and Hein, L. (1987). *Integrated product development*. IFS (Publications). (page 15).
- Anquetil, N., Kulesza, U., Mitschke, R., Moreira, A., Royer, J.-C., Rummler, A., and Sousa, A. (2010). A model-driven traceability framework for software product lines. *Software & Systems Modeling*, 9(4) :427–451. (pages xix, 124, 254).
- Anthony, R. (1965). *Planning and control systems : a framework for analysis*. Studies in management control. Division of Research, Graduate School of Business Administration, Harvard University. (pages 29, 50).
- Appelbaum, E. (2000). *Manufacturing Advantage : Why High-performance Work Systems Pay Off*. IRL Cornell paperbacks. Cornell University Press. (page 11).
- Aruldoss, M., Lakshmi, T. M., and Venkatesan, V. P. (2013). A survey on multi criteria decision making methods and its applications. *American Journal of Information Systems*, 1(1) :31–43. (page 37).
- Asbury, S. and Jacobs, E. (2014). *Dynamic Risk Assessment : The Practical Guide to Making Risk-Based Decisions with the 3-Level Risk Management Model*. Taylor & Francis. (page 71).
- Ashton, K. et al. (2009). That "internet of things" thing. *RFID journal*, 22(7) :97–114. (pages 88, 259).
- Åström, K. J. and Hägglund, T. (1995). *PID controllers : theory, design, and tuning*, volume 2. Instrument society of America Research Triangle Park, NC. (page 42).
- Aviation-Knowledge (2012). *Decision Making Model*. AviationKnowledge. Available at : <http://aviationknowledge.wikidot.com/sop:decision-making-model>. (page 71).
- Azevedo, G. and Bertrand, H. (2000). From multinational to global companies : identifying the dimensions of the change. In *12th Sase Annual Meeting, Conference On-Line Proceedings*. (pages 104, 259).
- Badke-Schaub, P. and Gehrlicher, A. (2003). Patterns of decisions in design : leaps, loops, cycles, sequences and meta-processes. In *DS 31 : Proceedings of ICED 03, the 14th International Conference on Engineering Design, Stockholm*. (pages 19, 22).
- Baeza-Yates, R., Ribeiro-Neto, B., et al. (1999). *Modern information retrieval*, volume 463. ACM press New York. (page 124).
- Bailey, I. (2008). Brief introduction to modaf with v1. 2 updates. In *IET Seminar on Enterprise Architecture Framework, London*, pages 1–18. (page 92).
- Balci, O. (1997). Verification validation and accreditation of simulation models. In *Proceedings of the 29th conference on Winter simulation*, pages 135–141. IEEE Computer Society. (pages xviii, 184, 185, 189).
- Balci, O. (1998). *Software Engineering Lecture Notes*. Department of Computer Science, Virginia Tech, Blacksburg, VA. (pages 25, 26).
- Balling, R. J. and Sobieszczanski-Sobieski, J. (1996). Optimization of coupled systems-a critical overview of approaches. *AIAA journal*, 34(1) :6–17. (page 35).
- Barbera, S., Hammond, P., and Seidl, C. (1998). *Handbook of Utility Theory : Volume 1 : Principles*. Handbook of Utility Theory. Springer US. (page 34).
- Bazerman, M. and Watkins, M. (2004). *Predictable Surprises : The Disasters You Should Have Seen Coming, and how to Prevent Them*. Leadership for the common good. Harvard Business School Press. (page 33).

- Bazhenova, E. and Weske, M. (2016). Deriving decision models from process models by enhanced decision mining. In *International conference on business process management*, pages 444–457. Springer. (page 165).
- Becker, J., Rosemann, M., and Uthmann, C. v. (2000). Guidelines of business process modeling. In *Business Process Management, Models, Techniques, and Empirical Studies*, pages 30–49, London, UK, UK. Springer-Verlag. (page 103).
- Beemer, B. A. and Gregg, D. G. (2008). Advisory systems to support decision making. In *Handbook on Decision Support Systems I*, pages 511–527. Springer. (page 45).
- Bekke, J. (1992). *Semantic Data Modeling*. Prentice Hall. (page 104).
- Bell, D. E., Raiffa, H., and Tversky, A. (1988). *Decision making : Descriptive, normative, and prescriptive interactions*. cambridge university Press. (page 34).
- Bellagamba, L. (2012). *Systems Engineering and Architecting : Creating Formal Requirements*. CRC Press. (pages 16, 183, 257).
- Bellatreche, L., Dung, N. X., Pierra, G., and Hondjack, D. (2006). Contribution of ontology-based data modeling to automatic integration of electronic catalogues within engineering databases. *Computers in Industry*, 57(8) :711–724. (page 104).
- Bellinger, G., Castro, D., and Mills, A. (2004). Data, information, knowledge, and wisdom. (page 123).
- Belton, V. and Stewart, T. (2002). *Multiple Criteria Decision Analysis : An Integrated Approach*. Springer US. (pages 34, 38).
- Ben-Haim, Y. (2001). *Information-gap Decision Theory : Decisions Under Severe Uncertainty*. Academic Press Series on Decision and Risk. Academic Press. (page 35).
- Benner, L. (1975). D.e.c.i.d.e. in hazardous materials emergencies. *Fire journal*, 69(4) :21–26. (page 71).
- Berger, J. (2013). *Statistical Decision Theory : Foundations, Concepts, and Methods*. Springer Series in Statistics. Springer New York. (page 34).
- Berliner, C. and Brimson, J. (1988). *Cost Management for Today's Advanced Manufacturing : The CAM-I Conceptual Design*. Harvard Business School Press. (pages xiii, 18, 20, 23).
- Beverland, M. B., Ewing, M. T., and Matanda, M. J. (2006). Driving-market or market-driven? a case study analysis of the new product development practices of chinese business-to-business firms. *Industrial Marketing Management*, 35(3) :383 – 393. New Product Development in Asia. (page 13).
- Bézivin, J. (2005). Model driven engineering : An emerging technical space. In *International Summer School on Generative and Transformational Techniques in Software Engineering*, pages 36–64. Springer. (pages 156, 261).
- Bhargava, H. K., Power, D. J., and Sun, D. (2007). Progress in web-based decision support technologies. *Decision Support Systems*, 43(4) :1083 – 1095. Special Issue Clusters. (page 51).
- Birge, J. and Louveaux, F. (2011). *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering. Springer New York. (page 38).
- BKCASE (2017). *The Guide to the Systems Engineering Body of Knowledge (SEBoK)*. v. 1.9. R.D. Adcock (EIC). Hoboken, NJ : The Trustees of the Stevens Institute of Technology. Accessed DATE. www.sebokwiki.org. BKCASE is managed and maintained by the Stevens Institute of Technology Systems Engineering Research Center, the International Council on Systems Engineering, and the Institute of Electrical and Electronics Engineers Computer Society. (pages xiv, 15, 76, 77, 183, 184, 261).

- Blessing, L. and Chakrabarti, A. (2009). *DRM, a Design Research Methodology*. Springer London. (page 20).
- Bobadilla, J., Ortega, F., Hernando, A., and Gutiérrez, A. (2013). Recommender systems survey. *Knowledge-Based Systems*, 46 :109 – 132. (pages xiv, 49).
- Boeing (2005). *Systems Architecting : An introduction*. Version 1.0. (pages xiv, 89, 90).
- Bonini, C. (1963). *Simulation of information and decision systems in the firm*. Ford Foundation doctoral dissertation series. Prentice-Hall. (page 50).
- Booch, G. (2004). *Object-Oriented Analysis and Design with Applications (3rd Edition)*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA. (page 96).
- Borenstein, D. (1998). Towards a practical method to validate decision support systems. *Decision Support Systems*, 23(3) :227 – 239. (pages xviii, 187, 188, 189).
- Bose, R. J. C. and van der Aalst, W. M. (2009). Abstractions in process mining : A taxonomy of patterns. In *Business Process Management*, pages 159–175. Springer. (page 158).
- Bouyssou, D. (2000). *Evaluation and Decision Models : A Critical Perspective*. International Series in Operations Research & Management Science. Springer US. (page 34).
- Brim, O. (1962). *Personality and Decision Processes : Studies in the Social Psychology of Thinking*. Stanford studies in sociology. Stanford University Press. (page 43).
- Brown, S. L. and Eisenhardt, K. M. (1995). Product development : Past research, present findings, and future directions. *The Academy of Management Review*, 20(2) :343–378. (page 15).
- Bubenko, J. and Kirikova, M. (1995). "worlds" in requirements acquisition and modelling. (page 103).
- Buckley, P. J., Pass, C. L., and Prescott, K. (1988). Measures of international competitiveness : A critical survey. *Journal of Marketing Management*, 4(2) :175–200. (page 13).
- Buede, D. (2011). *The Engineering Design of Systems : Models and Methods*. Wiley Series in Systems Engineering and Management. Wiley. (pages xxi, 15, 29, 76).
- Burke, R. (2007). *Hybrid Web Recommender Systems*, pages 377–408. Springer Berlin Heidelberg, Berlin, Heidelberg. (page 47).
- Burstein, F., Brézillon, P., and Zaslavsky, A. (2011). *Supporting real-time decision making : The role of context in decision support on the move*, volume 13 of *Annals of Information Systems*. Springer Verlag. (pages 23, 175).
- Burstein, F. and Holsapple, C. (2008). *Handbook on Decision Support Systems 1 : Basic Themes*. International Handbooks on Information Systems. Springer Berlin Heidelberg. (page 51).
- Carbone, E. and Hey, J. D. (2000). Which error story is best? *Journal of Risk and Uncertainty*, 20(2) :161–176. (page 35).
- Carlsson, C., Jelassi, T., and Walden, P. (1998). Intelligent systems and active dss. In *hicss*, page 4. IEEE. (page 50).
- Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168. ACM. (page 172).

- Carvalho, D., Campbell, R., Belford, G., and Mickunas, D. (2003). Definition of a user environment in a ubiquitous system. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2888 :1151–1169. (pages 109, 259).
- Castle, N. (2018). *What is Semi-Supervised Learning?* DataScience. Available at : <https://www.datascience.com/blog/what-is-semi-supervised-learning>. (page 140).
- Catalkaya, S., Knuplesch, D., Chiao, C., and Reichert, M. (2014). Enriching business process models with decision rules. In Lohmann, N., Song, M., and Wohed, P., editors, *Business Process Management Workshops*, pages 198–211, Cham. Springer International Publishing. (page 165).
- Cavaillès, J., Délégation Générale pour l'Armement., and Equipe qualité. (1995). *Méthodes de management de Programme : Pour l'utilisation interne à la DGA, cet ouvrage est repéré sous le code DGA/AQ902*. Teknea, Toulouse. (page 15).
- Chai, J., Liu, J. N., and Ngai, E. W. (2013). Application of decision-making techniques in supplier selection : A systematic review of literature. *Expert Systems with Applications*, 40(10) :3872 – 3885. (pages 30, 31, 38).
- Chapelle, O., Schlkopf, B., and Zien, A. (2010). *Semi-Supervised Learning*. The MIT Press, 1st edition. (page 140).
- Chase, R., Aquilano, N., and Jacobs, F. (2004). *Operations Management for Competitive Advantage*. The McGraw-Hill / Irwin Series : Operations and decision sciences. McGraw-Hill. (pages 11, 18).
- Chau, K. and Albermani, F. (2004). An expert system on design of liquid-retaining structures with blackboard architecture. *Expert Systems*, 21(4) :183–191. (page 45).
- Clark, A., Grünig, R., O'Dea, C., and Kühn, R. (2009). *Successful Decision-making : A Systematic Approach to Complex Problems*. Business and Economics. Springer Berlin Heidelberg. (pages xiii, xxi, 32, 33).
- Clark, K. and Fujimoto, T. (1991). *Product Development Performance : Strategy, Organization, and Management in the World Auto Industry*. Harvard Business School Press. (page 16).
- Clark, K. B., Chew, W. B., and Fujimoto, T. (1987). Product development in the world auto industry. *Brookings Papers on Economic Activity*, 18(3) :729–782. (page 16).
- Clark, K. B. and Fujimoto, T. (1992). Product development and competitiveness. *Journal of the Japanese and International Economies*, 6(2) :101 – 143. (page 12).
- Cleland-Huang, J., Gotel, O., and Zisman, A. (2012). *Software and Systems Traceability*. Springer Publishing Company, Incorporated. (pages xv, 121, 122, 123, 124).
- Clerc, M. (2010). *Particle Swarm Optimization*. ISTE. Wiley. (page 39).
- Cohon, J. (2004). *Multiobjective Programming and Planning*. Dover Books on Computer Science Series. Dover Publications. (page 38).
- Collins, J. (2011). *Good to Great : Why Some Companies Make the Leap...And Others Don't*. HarperCollins. (page 11).
- Comba, L., Belforte, G., Dabbene, F., and Gay, P. (2013). Methods for traceability in food production processes involving bulk products. *Biosystems Engineering*, 116(1) :51 – 63. (page 124).
- Combacau, M., Berruet, P., Zamai, E., Charbonnaud, P., and Khatab, A. (2000). Supervision and monitoring of production systems. *IFAC Proceedings Volumes*, 33(17) :849 – 854. 2nd IFAC Conference on Management and Control of Production and Logistics (MCPL 2000), Grenoble, France, 5-8 July 2000. (page 23).

- Conklin, J. and Begeman, M. L. (1987). gibis : A hypertext tool for team design deliberation. In *Proceedings of the ACM conference on Hypertext*, pages 247–251. ACM. (page 124).
- Cooper, W. W., Seiford, L. M., and Zhu, J. (2004). *Data Envelopment Analysis*, pages 1–39. Springer US, Boston, MA. (page 38).
- Corner, J. L. and Corner, P. D. (1995). Characteristics of decisions in decision analysis practice. *The Journal of the Operational Research Society*, 46(3) :304–314. (page 30).
- Council, N., Education, D., Society, B., Characterization, C., Fineberg, H., and Stern, P. (1996). *Understanding Risk : Informing Decisions in a Democratic Society*. National Academies Press. (page 72).
- Couturier, P., LÃ´, M., Imoussaten, A., Chapurlat, V., and Montmain, J. (2014). Tracking the consequences of design decisions in mechatronic systems engineering. *Mechatronics*, 24(7) :763 – 774. 1. Model-Based Mechatronic System Design 2. Model Based Engineering. (pages xv, xix, 107, 112, 113, 124, 244).
- Crerie, R., Baião, F. A., and Santoro, F. M. (2009). Discovering business rules through process mining. In *Enterprise, Business-Process and Information Systems Modeling*, pages 136–148. Springer. (page 150).
- Crnkovic, I., Asklund, U., and Dahlqvist, A. (2003). *Implementing and Integrating Product Data Management and Software Configuration Management*. Artech House computing library. Artech House. (page 174).
- Cutkosky, M. R., Engelmores, R. S., Fikes, R. E., Genesereth, M. R., Gruber, T. R., Mark, W. S., Tenenbaum, J. M., and Weber, J. C. (1993). Pact : an experiment in integrating concurrent engineering systems. *Computer*, 26(1) :28–37. (page 38).
- Cuyvers, R., Lauwereins, R., and Peperstraete, J. (1990). Fault-tolerance in process control possibilities, limitations and trends. *Journal A : vol*, 31(4). (page 40).
- Dale, E. (1973). *Management : theory and practice*. McGraw-Hill series in management. McGraw-Hill. (page 29).
- Daoudi, F. and Nurcan, S. (2007). A benchmarking framework for methods to design flexible business processes. *Software Process : Improvement and Practice*, 12(1) :51–63. (pages 11, 19, 111).
- Dash, M. and Koot, P. W. (2009). Feature selection for clustering. In *Encyclopedia of database systems*, pages 1119–1125. Springer. (page 139).
- Davenport, T. H. (1993). *Process Innovation : Reengineering Work Through Information Technology*. Harvard Business School Press, Boston, MA, USA. (page 18).
- Davenport, T. H. and Short, J. E. (2003). Information technology and business process redesign. *Operations management : critical perspectives on business and management*, 1 :97. (page 19).
- David, B. (1987). Multi-expert systems for cad. In *Intelligent CAD Systems I*, pages 57–67. Springer. (page 45).
- De Leoni, M., Dumas, M., and García-Bañuelos, L. (2013). Discovering branching conditions from business process execution logs. In Cortellessa, V. and Varró, D., editors, *Fundamental Approaches to Software Engineering*, pages 114–129, Berlin, Heidelberg. Springer Berlin Heidelberg. (pages 165, 166).
- De Leoni, M. and van der Aalst, W. M. (2013). Data-aware process mining : discovering decisions in processes using alignments. In *Proceedings of the 28th annual ACM symposium on applied computing*, pages 1454–1461. ACM. (pages 165, 166).

- Delic, K. A., Douillet, L., and Dayal, U. (2001). Towards an architecture for real-time decision support systems : challenges and solutions. In *Proceedings 2001 International Database Engineering and Applications Symposium*, pages 303–311. (page 175).
- Delir Haghighi, P., Krishnaswamy, S., Zaslavsky, A., and Gaber, M. M. (2008). Reasoning about context in uncertain pervasive computing environments. In Roggen, D., Lombriser, C., Tröster, G., Kortuem, G., and Havinga, P., editors, *Smart Sensing and Context*, pages 112–125, Berlin, Heidelberg. Springer Berlin Heidelberg. (page 175).
- Dertouzos, M., Lester, R., and Solow, R. (1989). *Made in America*. Cambridge, MA (USA); The MIT Press. (page 10).
- Dewey, J. (1910). *How We Think*. Heath’s pedagogical library. D.C. Heath & Company. (page 42).
- Dey, A. K. (2001). Understanding and using context. *Personal Ubiquitous Comput.*, 5(1) :4–7. (page 175).
- Dezfuli, H., Maggio, G., and Everett, C. (2010). Risk-Informed Decision Making; Application to the Technology Development Alternative Selection. In *Making Safety Matter*, volume 680 of *ESA Special Publication*, page 32. (page 30).
- Dezfuli, H., Stamatelatos, M., Maggio, G., Everett, C., and Youngblood, R. (2010). *Nasa Risk-Informed Decision Making Handbook. Version 1. 0 - Nasa/Sp-2010-576*. Books Express Publishing. (pages xiv, 63, 64, 65).
- Ding, S. X. (2008). *Model-based fault diagnosis techniques : design schemes, algorithms, and tools*. Springer Science & Business Media. (page 41).
- DMS (2016a). *Decision Modeling with DMN, How to Build a Decision Requirements Model using the new Decision Model and Notation (DMN) standard*. Decision Management Solutions. Available at : <http://www.decisionmanagementsolutions.com/whitepaper/decision-modeling-with-dmn/>. (pages xiv, 78, 80).
- DMS (2016b). *An Introduction to Decision Modeling with DMN*. Decision Management Solutions. Available at : <http://www.omg.org/news/whitepapers/An-Introduction-to-Decision-Modeling-with-DMN.pdf>. (pages xiv, 79, 80).
- DoD (2015). *Department of Defense Architectural Framework (DoDAF) V2.02*. Volume II : Architectural Data and Models, U.S. Department of Defense (DoD). (pages xiv, 90, 91).
- Dong, X., DeVries, W., and Wozny, M. (1991). Feature-based reasoning in fixture design. *CIRP Annals-Manufacturing Technology*, 40(1) :111–114. (page 38).
- Donovan, J. J. and Madnick, S. E. (1977). Institutional and ad hoc dss and their effective use. *SIGMIS Database*, 8(3) :79–88. (page 50).
- Doyle, J. (1999). Rational decision making. *MIT encyclopedia of the cognitive sciences*, pages 701–703. (page 34).
- Dumas, M., Van der Aalst, W. M., and Ter Hofstede, A. H. (2005). *Process-aware information systems : bridging people and software through process technology*. John Wiley & Sons. (page 143).
- Duncan, S. J., Bras, B., and Paredis, C. J. (2008). An approach to robust decision making under severe uncertainty in life cycle design. *International Journal of Sustainable Design*, 1(1) :45–59. (page 35).
- Dunkl, R., Rinderle-Ma, S., Grossmann, W., and Anton Fröschl, K. (2015). A method for analyzing time series data in process mining : Application and extension of decision point analysis. In Nurcan, S. and Pimenidis, E., editors, *Information Systems Engineering in Complex Environments*, pages 68–84, Cham. Springer International Publishing. (pages 165, 166).

- Durkin, J. (1996). Expert systems : a view of the field. *IEEE Expert : Intelligent Systems and Their Applications*, 11(2) :56–63. (page 44).
- Edwards, W. (1954). The theory of decision making. *Psychological Bulletin*, 51(4) :380–417. (page 32).
- Edwards, W. (1961). Behavioral decision theory. *Annual Review of Psychology*, 12(1) :473–498. PMID : 13725822. (page 34).
- Edwards, W. (1962). Dynamic decision theory and probabilistic information processings. *Human Factors*, 4(2) :59–74. (page 35).
- Edwards, W. and Fasolo, B. (2001). Decision technology. *Annual Review of Psychology*, 52(1) :581–606. PMID : 11148318. (page 34).
- Edwards, W., Miles, R., and von Winterfeldt, D. (2007). *Advances in Decision Analysis : From Foundations to Applications*. Cambridge University Press. (page 76).
- EHEST (2012). *Leaflet HE 4 - Decision Making for Single Pilot Helicopter Operations*. European Aviation Safety Agency. (pages 68, 71).
- El Gamoussi, S. (2016). *Proposition of an improvement methodology of the Product Development Process based on a Lean approach*. Theses, Université Paris-Saclay. (pages 15, 17).
- Eppinger, S. D. and Browning, T. R. (2012). *Design structure matrix methods and applications*. MIT press. (page 39).
- Es-Soufi, W., Yahia, E., and Roucoules, L. (2016). On the use of process mining and machine learning to support decision making in systems design. In *IFIP International Conference on Product Lifecycle Management*, pages 56–66. Springer. (pages 150, 158).
- Es-Soufi, W., Yahia, E., and Roucoules, L. (2017a). *Collaborative Design and Supervision Processes Meta-Model for Rationale Capitalization*, pages 1123–1130. Springer International Publishing, Cham. (pages 110, 176, 177).
- Es-Soufi, W., Yahia, E., and Roucoules, L. (2017b). *A Process Mining Based Approach to Support Decision Making*, pages 264–274. Springer International Publishing, Cham. (pages xviii, 171).
- Espiñeira, M. and Santaclara, F. J. (2016). *Advances in food traceability techniques and technologies : improving quality throughout the food chain*. Woodhead Publishing. (page 124).
- Evers, J. H., Oehler, G. M., and Tucker, M. G. (1998). Improving engineering productivity : a time study of an engineer’s typical work day. In *Engineering and Technology Management, 1998. Pioneering New Technologies : Management Issues and Challenges in the Third Millennium. IEMC '98 Proceedings. International Conference on*, pages 377–383. (page 22).
- FAA (2013). *Instrument Flying Handbook : FAA Handbook : Faa-h-8083-15b*. Createspace Independent Pub. (pages 68, 69, 71).
- Fanchon, J. (2010). *Guide des sciences et technologies industrielles*. AFNOR. (page 15).
- Farhang Mehr, A. and Tumer, I. Y. (2006). Risk-Based Decision-Making for Managing Resources During the Design of Complex Space Exploration Systems. *Journal of Mechanical Design*, 128(4) :1014–1022. (page 63).
- Fasbinder, M. (2007). *Why model business processes ?* DeveloperWorks, IBM Corporation. (page 103).
- Ferris, C. and Farrell, J. (2003). What are web services ? *Communications of the ACM*, 46(6) :31. (pages 132, 260).

- Finger, S. and Dixon, J. R. (1989a). A review of research in mechanical engineering design. part i : Descriptive, prescriptive, and computer-based models of design processes. *Research in Engineering Design*, 1(1) :51–67. (page 37).
- Finger, S. and Dixon, J. R. (1989b). A review of research in mechanical engineering design. part ii : Representations, analysis, and design for the life cycle. *Research in Engineering Design*, 1(2) :121–137. (page 37).
- Finlay, P. N. and Wilson, J. M. (1997). Validity of decision support systems : towards a validation methodology. *Systems Research and Behavioral Science : The Official Journal of the International Federation for Systems Research*, 14(3) :169–182. (pages xviii, 186, 187, 189).
- Fischer, G. (1991). The importance of models in making complex systems comprehensible. In Tauber, M. and Ackermann, D., editors, *Mental Models and Human-Computer Interaction 2*, volume 2 of *Human Factors in Information Technology*, pages 3 – 36. North-Holland. (page 103).
- Fishburn, P. (1970). *Utility theory for decision making*. Publications in operations research. Wiley. (page 34).
- Fishburn, P. C. (1981). Subjective expected utility : A review of normative theories. *Theory and Decision*, 13(2) :139–199. (page 34).
- FOCA (2016). *Operations and Training Manual Certification Leaflet*. FOCA GM/INFO. (page 71).
- Ford (2005). Bill ford : Innovation key to ford’s future; commitment to hybrids to grow. <http://media.ford.com>. (page 13).
- Ford, F. N. (1985a). Decision support systems and expert systems : A comparison. *Information & Management*, 8(1) :21 – 26. (page 44).
- Ford, F. N. (1985b). Decision support systems and expert systems : a comparison. *Information & Management*, 8(1) :21–26. (page 47).
- Fowler, M. (1997). *Analysis Patterns : Reusable Objects Models*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. (page 106).
- Fox, V., Hightower, J., Liao, L., Schulz, D., and Borriello, G. (2003). Bayesian filtering for location estimation. *IEEE Pervasive Computing*, 2(3) :24–33. (page 175).
- French, M. (1971). *Engineering Design : The Conceptual Stage*. Heinemann Educational. (page 37).
- FREY, E., GOMES, S., and SAGOT, J.-C. (2007). Application de la méthode qfd comme outil d’extraction des connaissances métier en conception intégrée. pages –. (page 20).
- Garetti, M., Terzi, S., Bertacci, N., and Brianza, M. (2005). Organisational change and knowledge management in plm implementation. *International Journal of Product Lifecycle Management*, 1(1) :43–51. (page 17).
- Gautier, F. and Glard, V. (2000). *Vers une meilleure maîtrise des coûts engagés sur le cycle de vie, lors de la conception de produits nouveaux*, volume 6. Association francophone de comptabilité, Paris. bibtex : gautier_vers_2000. (pages xiii, 20).
- Genard, J.-L. and Pirlot, M. (2002). *Multi-Criteria Decision-Aid in a Philosophical Perspective*, pages 89–117. Springer US, Boston, MA. (page 34).

- Gero, J. (1985). *Knowledge engineering in computer-aided design : proceedings of the IFIP WG 5.2 Working Conference on Knowledge Engineering in Computer-Aided Design, Budapest, Hungary, 17-19 September, 1984*. Number v. 1984 in *Knowledge Engineering in Computer-aided Design : Proceedings of the IFIP WG 5.2 Working Conference on Knowledge Engineering in Computer-Aided Design, Budapest, Hungary, 17-19 September, 1984*. North-Holland. (page 38).
- Ghattas, J., Soffer, P., and Peleg, M. (2014). Improving business process decision making based on past experience. *Decision Support Systems*, 59 :93–107. (page 165).
- Gigerenzer, G. and Gaissmaier, W. (2011). Heuristic decision making. *Annual review of psychology*, 62 :451–482. (page 36).
- Gingele, J., Childe, S., and Miles, M. (2002). A modelling technique for re-engineering business processes controlled by {ISO} 9001. *Computers in Industry*, 49(3) :235 – 251. (page 120).
- Ginn, D. and Zairi, M. (2005). Best practice qfd application : an internal/external benchmarking approach based on ford motors' experience. *International Journal of Quality & Reliability Management*, 22(1) :38–58. (pages 38, 39).
- Godart, C. and Perrin, O. (2009). *Les processus métiers : Concepts, modèles et systèmes*. IC2, Informatique et systèmes d'information. Hermès science publications. (page 19).
- Goethals, B. and Zaki, M. J. (2004). Advances in frequent itemset mining implementations : report on fimi'03. *Acm Sigkdd Explorations Newsletter*, 6(1) :109–117. (page 147).
- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition. (page 39).
- González-Moreno, Á. and Sáez-Martínez, F. J. (2008). Rivalry and strategic groups : what makes a company a rival? *Journal of Management & Governance*, 12(3) :261–285. (page 10).
- Gorry, G. A. and Scott Morton, M. S. (1971). A framework for management information systems. (page 50).
- Griira, N., Crucianu, M., and Boujemaa, N. (2004). Unsupervised and semi-supervised clustering : a brief survey. *A review of machine learning techniques for processing multimedia content, Report of the MUSCLE European Network of Excellence (FP6)*, pages 1001–1030. (pages 140, 141).
- Grover, J. (2012). *Strategic Economic Decision-Making : Using Bayesian Belief Networks to Solve Complex Problems*. SpringerBriefs in Statistics. Springer New York. (page 46).
- Gunther, C. and Verbeek, H. (2014). *XES - standard definition*. BPM reports. BPMcenter.org. (pages xvii, 153, 155).
- Guo, Z., Ngai, E., Yang, C., and Liang, X. (2015). An rfid-based intelligent decision support system architecture for production monitoring and scheduling in a distributed manufacturing environment. *International Journal of Production Economics*, 159 :16 – 28. (pages xiv, 52, 54).
- Haber, R. E., Alique, A., Ros, S., and Haber, R. (2002). Application of knowledge-based systems for supervision and control of machining processes. In *Handbook of Software Engineering and Knowledge Engineering : Volume II : Emerging Technologies*, pages 673–709. World Scientific. (pages 41, 42).
- Hackathorn, R. D. and Keen, P. G. W. (1981). Organizational strategies for personal computing in decision support systems. *MIS Q.*, 5(3) :21–27. (page 50).
- Hales, C. (1987). *Analysis of the engineering design process in an industrial context*. PhD thesis, University of Cambridge. (page 37).

- Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2001). On clustering validation techniques. *Journal of intelligent information systems*, 17(2-3) :107–145. (pages [xvi](#), [139](#), [140](#)).
- Hammond, J., Keeney, R., and Raiffa, H. (1999). *Smart Choices : A Practical Guide to Making Better Decisions*. Number v. 226 in *Smart Choices : A Practical Guide to Making Better Decisions*. Harvard Business School Press. (pages [31](#), [43](#)).
- Han, J., Cheng, H., Xin, D., and Yan, X. (2007). Frequent pattern mining : current status and future directions. *Data Mining and Knowledge Discovery*, 15(1) :55–86. (page [137](#)).
- Han, J., Kamber, M., and Pei, J. (2011). *Data Mining : Concepts and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science. (page [138](#)).
- Hansen, C. T. and Andreasen, M. M. (2004). A mapping of design decision-making. In *DS 32 : Proceedings of DESIGN 2004, the 8th International Design Conference, Dubrovnik, Croatia*. (pages [21](#), [22](#)).
- Hansson, S. O. (2005). *Decision Theory : A Brief Introduction*. Department of Philosophy and the History of Technology, Royal Institute of Technology (KTH), Stockholm. (pages [33](#), [42](#)).
- Harman, D. (1993). Overview of the first trec conference. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '93*, pages 36–47, New York, NY, USA. ACM. (page [124](#)).
- Harris, D. and Li, W.-C. (2017). *Decision Making in Aviation*. Taylor & Francis. (pages [xiv](#), [69](#)).
- Harrison, R. (2018). *Togaf (r) 9 Foundation Study Guide*. Van Haren. (page [92](#)).
- Hartley, J. R. (1990). *Concurrent Engineering : Shortening Lead Times, Raising Quality, and Lowering Costs*. Product development and design series. Productivity Press. (pages [17](#), [39](#)).
- Hastie, R. (2001). Problems for judgment and decision making. *Annual Review of Psychology*, 52(1) :653–683. PMID : 11148321. (page [34](#)).
- Hatchuel, A. and Weil, B. (2003). A new approach of innovative design : an introduction to ck theory. In *DS 31 : Proceedings of ICED 03, the 14th International Conference on Engineering Design, Stockholm*. (page [124](#)).
- Havey, M. (2005). *Essential Business Process Modeling*. O'Reilly Media. (page [103](#)).
- Heidari, F., Loucopoulos, P., and Kedad, Z. (2011). A quality-oriented business process meta-model. In *Workshop on Enterprise and Organizational Modeling and Simulation*, pages 85–99. Springer. (pages [xv](#), [xix](#), [107](#), [112](#), [113](#), [246](#)).
- Helfat, C., Finkelstein, S., Mitchell, W., Peteraf, M., Singh, H., Teece, D., and Winter, S. (2009). *Dynamic Capabilities : Understanding Strategic Change in Organizations*. Wiley. (page [174](#)).
- Henricksen, K., Indulska, J., and Rakotonirainy, A. (2002). Modeling context information in pervasive computing systems. In Mattern, F. and Naghshineh, M., editors, *Pervasive Computing*, pages 167–180, Berlin, Heidelberg. Springer Berlin Heidelberg. (page [175](#)).
- Hill, D. R. (1996). *Object-Oriented Analysis and Simulation*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA. (page [90](#)).
- Hill, T., Marquez, L., O'Connor, M., and Remus, W. (1994). Artificial neural network models for forecasting and decision making. *International Journal of Forecasting*, 10(1) :5 – 15. (page [39](#)).
- Hipp, J., Guntzer, U., and Nakhaeizadeh, G. (2000). Algorithms for association rule mining - a general survey and comparison. *SIGKDD Explor. Newsl.*, 2(1) :58–64. (pages [39](#), [135](#)).

- Hodge, V. and Austin, J. (2004). A survey of outlier detection methodologies. *Artificial intelligence review*, 22(2) :85–126. (page 138).
- Hooley, B. L. and Foyle, D. C. (2006). Pilot navigation errors on the airport surface : Identifying contributing factors and mitigating solutions. *The International Journal of Aviation Psychology*, 16(1) :51–76. (pages 68, 69).
- Hopgood, A. A., editor (2001). *Intelligent Systems for Engineers and Scientists (2Nd Ed.)*. CRC Press, Inc., Boca Raton, FL, USA. (pages xiii, 45, 46).
- Howard, T., Culley, S., and Dekoninck, E. (2007). Creativity in the engineering design process. In *16th International Design Conference on Engineering Design, Paris*. (page 19).
- Huang, G. (1996). *Design for X : Concurrent engineering imperatives*. Springer Berlin. (pages 17, 38).
- Hug, C., Deneckère, R., and Salinesi, C. (2012). Map-tbs : Map process enactment traces and analysis. In *Research Challenges in Information Science (RCIS), 2012 Sixth International Conference on*, pages 1–6. IEEE. (pages xix, 124, 255).
- IIBA (2015). *A Guide to the Business Analysis Body of Knowledge (Babok Guide)*. Number v. 3. International Institute of Business Analysis. (pages xiv, 80, 81).
- Import.io (2017). *Data Mining vs. Machine Learning : What's The Difference ?* Available at : <https://www.import.io/post/data-mining-machine-learning-difference/>. (pages 132, 260).
- INCOSE (2015). *Systems Engineering Handbook : A Guide for System Life Cycle Processes and Activities*. Wiley. (pages xiii, xiv, xxi, 15, 17, 21, 29, 75, 88, 89).
- Isaksson, O., Jeppsson, P., Fuxin, F., Johansson, H., Johansson, P., Katchaounov, T., Lindeblad, M., Haoxue, M., Malmqvist, J., Mesihovic, S., et al. (2000). Trends in product modelling : an endrea perspective. In *Produktmodeller 2000 : 07/10/2000-08/10/2000*. Linköping universitet. (page 104).
- Isermann, R. (1984). Process fault detection based on modeling and estimation methods - a survey. *automatica*, 20(4) :387–404. (page 41).
- Isermann, R. and Ballé, P. (1996). Trends in the application of model based fault detection and diagnosis of technical processes. *IFAC Proceedings Volumes*, 29(1) :6325 – 6336. 13th World Congress of IFAC, 1996, San Francisco USA, 30 June - 5 July. (page 40).
- Isinkaye, F., Folajimi, Y., and Ojokoh, B. (2015). Recommendation systems : Principles, methods and evaluation. *Egyptian Informatics Journal*, 16(3) :261 – 273. (page 48).
- ISO (1990). Ieee standard glossary of software engineering terminology. *IEEE Std 610.12-1990*, pages 1–84. (page 183).
- ISO (1994). *ISO 8402 : 1994 : Quality Management and Quality Assurance - Vocabulary*. International Organization for Standardization. (page 121).
- ISO (2011). Systems and software engineering – architecture description. *ISO/IEC/IEEE 42010 :2011(E) (Revision of ISO/IEC 42010 :2007 and IEEE Std 1471-2000)*, pages 1 – 46. (pages 88, 259).
- ISO (2015a). *ISO 9001 :2015 How to use it*. [ebook] Geneva, Switzerland : iso.org, pp.1-12. Available at : https://www.iso.org/files/live/sites/isoorg/files/archive/pdf/en/iso_9001-2015_-_how_to_use_it.pdf [Accessed 15 Oct. 2017]. (page 82).
- ISO (2015b). *Quality management principles*. 1st ed. [ebook] Geneva , Switzerland : ISO, pp.1-12. Available at : http://www.iso.org/iso/qmp_2012.pdf [Accessed 25 May 2015]. (page 82).

- ISO/IEC (2001). *ISO/IEC 9126. Software engineering – Product quality*. ISO. (page 26).
- Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering : A review. *ACM Comput. Surv.*, 31(3) :264–323. (page 140).
- James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An Introduction to Statistical Learning : with Applications in R*. Springer Texts in Statistics. Springer New York. (page 137).
- Jensen, F. (1996). *An Introduction To Bayesian Networks*. Taylor & Francis. (page 39).
- John S. Gero and Coyne, R. (1987). Knowledge-based Planning as a Design Paradigm. (page 37).
- Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning : A survey. *CoRR*, cs.AI/9605103. (pages xvi, 142).
- Kahneman, D. (2011). *Thinking, Fast and Slow*. Farrar, Straus and Giroux. (page 33).
- Kahneman, D. and Tversky, A. (2013). *Prospect Theory : An Analysis of Decision Under Risk*, chapter Chapter 6, pages 99–127. WORLD SCIENTIFIC. (page 34).
- Kahraman, C., Onar, S. C., and Oztaysi, B. (2015). Fuzzy multicriteria decision-making : A literature review. *International Journal of Computational Intelligence Systems*, 8(4) :637–666. (page 37).
- Kaplan, S. and Garrick, B. J. (1981). On the quantitative definition of risk. *Risk Analysis*, 1(1) :11–27. (page 30).
- Karray, M. H. (2012). *Contribution à la spécification et à l'élaboration d'une plateforme de maintenance orientée connaissances*. PhD thesis, Université de Franche-Comté. (pages xix, 124, 253).
- Katona, G. (1946). Psychological analysis of business decisions and expectations. *The American Economic Review*, 36(1) :44–62. (page 29).
- Keeney, R. L. (1993). Creativity in ms/or : Value-focused thinking-creativity directed toward decision making. *Interfaces*, 23(3) :62–67. (page 77).
- Keeney, R. L. (2002). Common mistakes in making value trade-offs. *Operations Research*, 50(6) :935–945. (page 76).
- Ketels, C. (2016). Review of competitiveness frameworks. *National Competitiveness Council of Dublin*. (page 10).
- Khatri, N. (2000). Managing human resource for competitive advantage : a study of companies in singapore. *The International Journal of Human Resource Management*, 11(2) :336–365. (page 11).
- Kodratoff, Y. and Michalski, R. (2014). *Machine Learning : An Artificial Intelligence Approach*. Number v. 3. Elsevier Science. (page 137).
- Kolodner, J. (2014). *Case-Based Reasoning*. Elsevier Science. (page 39).
- Kotsiantis, S. B. (2007). Supervised machine learning : A review of classification techniques. In *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering : Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*, pages 3–24, Amsterdam, The Netherlands, The Netherlands. IOS Press. (pages xvi, 139).
- Krause, F.-L., Kimura, F., Kjellberg, T., Lu, S.-Y., van der Wolf, Alting, L., ElMaraghy, H., Eversheim, W., Iwata, K., Suh, N., Tipnis, V., Week, M., and A.C.H. (1993). Product modelling. *CIRP Annals*, 42(2) :695 – 706. (page 38).

- Krishnamurty, S. (2006). Normative decision analysis in engineering design. *Decision Making in Engineering Design*, 4(4) :21–33. (page 37).
- Krishnapuram, B., Williams, D., Xue, Y., Carin, L., Figueiredo, M., and Hartemink, A. J. (2005). On semi-supervised classification. In *Advances in neural information processing systems*, pages 721–728. (page 141).
- Kroll, E., Shihmanter, A., et al. (2011). Capturing the conceptual design process with concept-configuration-evaluation triplets. In *DS 68-6 : Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol. 6 : Design Information and Knowledge, Lyngby/Copenhagen, Denmark, 15.-19.08. 2011*. (page 126).
- Kruchten, P. (1995). Architectural blueprints - the "4 + 1" view model of software architecture. *Tutorial Proceedings of Tri-Ada*, 95 :540–555. (pages xiv, 91, 92, 96).
- Kudyba, S. (2014). *Big Data, Mining, and Analytics : Components of Strategic Decision Making*. CRC Press. (page 137).
- Kulkarni, P. (2012). *Reinforcement and Systemic Machine Learning for Decision Making*. IEEE Press Series on Systems Science and Engineering. Wiley. (page 142).
- Kumar, P. and Tandon, P. (2015). Uncertainty and decision making in product design : a fuzzy approach. In *DS79 : Proceedings of The Third International Conference on Design Creativity, Indian Institute of Science, Bangalore*. (page 39).
- Kumar, V. (1992). Algorithms for constraint-satisfaction problems : A survey. *AI magazine*, 13(1) :32. (pages 114, 259).
- Kurtev, I., Bézivin, J., and Aksit, M. (2002). Technological spaces : An initial appraisal. *CoopIS, DOA*, 2002. (pages xvii, 157).
- Kvarnström, B. (2008). *Traceability methods for continuous processes*. PhD thesis, Luleå tekniska universitet. (page 121).
- Kwasitsu, L. (2003). Information-seeking behavior of design, process, and manufacturing engineers. *Library & Information Science Research*, 25(4) :459 – 476. (pages 20, 21).
- Labrousse, M. and Bernard, A. (2008). Fbs-ppre, an enterprise knowledge lifecycle model. In *Methods and tools for effective knowledge life-cycle-management*, pages 285–305. Springer. (pages xv, xix, 107, 109, 112, 113, 245).
- Laghouaouta, Y., Anwar, A., Nassar, M., and Coulette, B. (2017). A dedicated approach for model composition traceability. *Information and Software Technology*, 91 :142 – 159. (pages xix, 124, 255).
- Lai, S. H.-Y. (1993). KbdA - a knowledge based design system for assembly. *Computers & Industrial Engineering*, 25(1) :585 – 588. (page 47).
- Landry, M., Malouin, J.-L., Oral, M., et al. (1983). Model validation in operations research. *European Journal of Operational Research*, 14(3) :207–220. (pages xviii, 186, 189).
- Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., and Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, 6(4) :239–242. (pages 88, 259).
- Lau, R. (2002). Competitive factors and their relative importance in the us electronics and computer industries. *International Journal of Operations & Production Management*, 22(1) :125–135. (page 10).
- Lave, C. and March, J. (1975). *An introduction to models in the social sciences*. Harper & Row. (page 103).

- Lawson, B. (2006). *How Designers Think : The Design Process Demystified*. How Designers Think : The Design Process Demystified. Elsevier/Architectural. (pages 19, 21).
- Lee, B.-E. and Suh, S.-H. (2009). An architecture for ubiquitous product life cycle support system and its extension to machine tools with product data model. *The International Journal of Advanced Manufacturing Technology*, 42(5-6) :606–620. (pages xix, 109, 113, 114, 252).
- Lee, D.-H. (2013). Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2. (page 140).
- Lee, J. and Lai, K.-Y. (1991). What's in design rationale? *Human-Computer Interaction*, 6(3-4) :251–280. (pages 123, 124).
- Lee, J., Sun, M., and Lebanon, G. (2012). A comparative study of collaborative filtering algorithms. *arXiv preprint arXiv :1205.3193*. (page 49).
- Leemans, M. and van der Aalst, W. M. P. (2015). Discovery of frequent episodes in event logs. In Ceravolo, P., Russo, B., and Accorsi, R., editors, *Data-Driven Process Discovery and Analysis*, pages 1–31, Cham. Springer International Publishing. (page 147).
- Leemans, S. J., Fahland, D., and van der Aalst, W. M. (2014). Exploring processes and deviations. In *International Conference on Business Process Management*, pages 304–316. Springer. (page 161).
- Leemans, S. J. J., Fahland, D., and van der Aalst, W. M. P. (2013). Discovering block-structured process models from event logs - a constructive approach. In Colom, J.-M. and Desel, J., editors, *Application and Theory of Petri Nets and Concurrency*, pages 311–329, Berlin, Heidelberg. Springer Berlin Heidelberg. (page 161).
- Lengnick-Hall, C. A. (1992). Innovation and competitive advantage : What we know and what we need to learn. *Journal of Management*, 18(2) :399–429. (page 13).
- Lewis, K. and Mistree, F. (1998). Collaborative, sequential, and isolated decisions in design. *Journal of Mechanical Design*, 120(4) :643–652. (page 35).
- Leyval, L., Montmain, J., and Gentil, S. (1994). Qualitative analysis for decision making in supervision of industrial continuous processes. *Mathematics and Computers in Simulation*, 36(2) :149 – 163. (page 20).
- Li, D.-F. (2011). Closeness coefficient based nonlinear programming method for interval-valued intuitionistic fuzzy multiattribute decision making with incomplete preference information. *Applied Soft Computing*, 11(4) :3402 – 3418. (page 38).
- Li, D.-F., Chen, G.-H., and Huang, Z.-G. (2010). Linear programming method for multiattribute group decision making using if sets. *Information Sciences*, 180(9) :1591 – 1609. (page 38).
- Li, L. X. (2000). An analysis of sources of competitiveness and performance of chinese manufacturers. *International Journal of Operations & Production Management*, 20(3) :299–315. (pages 10, 13).
- Li, Y., Wan, L., and Xiong, T. (2011). Product data model for plm system. *The International Journal of Advanced Manufacturing Technology*, 55(9-12) :1149–1158. (pages xix, 109, 114, 250).
- Liew, A. (2007). Understanding data, information, knowledge and their inter-relationships. *Journal of Knowledge Management Practice*, 8(2) :1–16. (page 52).
- Linden, G., Smith, B., and York, J. (2003). Amazon.com recommendations : item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1) :76–80. (page 47).

- Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., and Lederberg, J. (1993). Dendral : A case study of the first expert system for scientific hypothesis formation. *Artificial Intelligence*, 61(2) :209 – 261. (page 44).
- Lipschutz, S. (1998). *Schaum's Outline of Theory and Problems of Set Theory and Related Topics*. Schaum's Outlines. McGraw-Hill, New York, second edition. (pages 30, 36).
- Liu, A., Lu, S., Zhang, Z., Li, T., and Xie, Y. (2017). Function recommender system for product planning and design. *CIRP Annals*, 66(1) :181 – 184. (page 50).
- Liu, J., Dolan, P., and Pedersen, E. R. (2010a). Personalized news recommendation based on click behavior. In *Proceedings of the 15th International Conference on Intelligent User Interfaces, IUI '10*, pages 31–40, New York, NY, USA. ACM. (page 47).
- Liu, S., Duffy, A. H. B., Whitfield, R. I., and Boyle, I. M. (2010b). Integration of decision support systems to improve decision support performance. *Knowledge and Information Systems*, 22(3) :261–286. (pages xiv, 52, 53).
- Liu, S., Forrest, J., and Yang, Y. (2012). A brief introduction to grey systems theory. *Grey Systems : Theory and Application*, 2(2) :89–104. (page 39).
- Liu, S.-Y., Chi, C.-F., and Li, W.-C. (2013). The application of human factors analysis and classification system (hfacs) to investigate human errors in helicopter accidents. In *Proceedings of the 10th International Conference on Engineering Psychology and Cognitive Ergonomics : Applications and Services - Volume Part II, EPCE'13*, pages 85–94, Berlin, Heidelberg. Springer-Verlag. (page 68).
- London, P., Hankins, B., Sapossnek, M., and Luby, S. (1992). Chapter 8 - {THE} {EXPERT} {COST} {AND} {MANUFACTURABILITY} guide : A {CUSTOMIZABLE} {EXPERT} {SYSTEM}. In TONG, C. and SRIRAM, D., editors, *Artificial Intelligence in Engineering Design*, pages 223 – 234. Academic Press, San Diego. (page 46).
- Loomes, G. and Sugden, R. (1982). Regret theory : An alternative theory of rational choice under uncertainty. *The Economic Journal*, 92(368) :805–824. (page 34).
- Lutters, E., van Houten, F. J., Bernard, A., Mermoz, E., and Schutte, C. S. (2014). Tools and techniques for product design. *CIRP Annals*, 63(2) :607 – 630. (pages 37, 39).
- Mabroukeh, N. R. and Ezeife, C. I. (2010). A taxonomy of sequential pattern mining algorithms. *ACM Computing Surveys (CSUR)*, 43(1) :3. (page 137).
- Maher, J. (1989). *Beyond CRM to decisional heuristics : An airline generated model to examine accidents and incidents caused by crew errors in deciding*. Fifth International Symposium on Aviation Psychology, Ohio State University. (page 71).
- Mallak, L. (1998). Putting organizational resilience to work. *Industrial Management (Norcross, Georgia)*, 40(6 NOV./DEC.) :8–13. (page 21).
- Mannhardt, F., de Leoni, M., and Reijers, H. A. (2015). The multi-perspective process explorer. In *BPM*, pages 130–134. (pages 165, 166).
- Mannhardt, F., De Leoni, M., Reijers, H. A., and Van Der Aalst, W. M. (2016). Decision mining revisited - discovering overlapping rules. In *International Conference on Advanced Information Systems Engineering*, pages 377–392. Springer. (page 165).
- Mannila, H., Toivonen, H., and Verkamo, A. I. (1997). Discovery of frequent episodes in event sequences. *Data mining and knowledge discovery*, 1(3) :259–289. (pages 147, 260).

- Mano, A. P., Simões, J. M. S., Lima, L. S., de Toledo, J. C., and da Silva, S. L. (2007). *The Main Problems in the Product Development Process by Large-sized Companies of the Brazilian Agricultural Machines and Implements Sector*, pages 749–755. Springer London, London. (page 23).
- Marca, D. A. and McGowan, C. L. (1987). *SADT : Structured Analysis and Design Technique*. McGraw-Hill, Inc., New York, NY, USA. (page 90).
- Mardani, A., Jusoh, A., Nor, K. M., Khalifah, Z., Zakwan, N., and Valipour, A. (2015a). Multiple criteria decision-making techniques and their applications - a review of the literature from 2000 to 2014. *Economic Research-Ekonomska Istraživanja*, 28(1) :516–571. (page 36).
- Mardani, A., Jusoh, A., and Zavadskas, E. K. (2015b). Fuzzy multiple criteria decision-making techniques and applications - two decades review from 1994 to 2014. *Expert Systems with Applications*, 42(8) :4126 – 4148. (page 37).
- Mardani, A., Jusoh, A., and Zavadskas, E. K. (2015c). Fuzzy multiple criteria decision-making techniques and applications—two decades review from 1994 to 2014. *Expert Systems with Applications*, 42(8) :4126–4148. (page 37).
- Mardani, A., Zavadskas, E. K., Govindan, K., Amat Senin, A., and Jusoh, A. (2016). Vikor technique : A systematic review of the state of the art literature on methodologies and applications. *Sustainability*, 8(1) :37. (page 37).
- Martins, P. V. and Zacarias, M. (2015). Business process and practice alignment meta-model. *Procedia Computer Science*, 64 :314 – 323. Conference on ENTERprise Information Systems/International Conference on Project MANagement/Conference on Health and Social Care Information Systems and Technologies, CENTERIS/ProjMAN / HCist 2015 October 7-9, 2015. (pages xv, xix, 107, 112, 113, 248).
- Maté, A. and Trujillo, J. (2012). A trace metamodel proposal based on the model driven architecture framework for the traceability of user requirements in data warehouses. *Information Systems*, 37(8) :753 – 766. Special Issue : Advanced Information Systems Engineering (CAiSE'11). (pages xix, 124, 255).
- Matlin, M. and Farmer, T. (2017). *Cognition*. John Wiley & Sons, Limited. (page 34).
- Matta, N., Atifi, H., and Ducellier, G. (2016). *Daily Knowledge Valuation in Organizations : Traceability and Capitalization*. Wiley. (pages 121, 123, 124).
- Mayo, M. (2017). *The Data Science Puzzle, Revisited*. KDnuggets. (pages xvi, 133).
- McKechnie, L., Fisher, Karen E., .. Erdelez, Sanda, .. for Information Science, A. S., and Technology (2005). *Theories of information behavior*. Medford, N.J. published for the American Society for Information Science and Technology by Information Today. Includes bibliographical references and index. (page 30).
- Merkert, J., Mueller, M., and Hubl, M. (2015). A survey of the application of machine learning in decision support systems. In *ECIS*. (page 50).
- Meyer, B. (1992). *Eiffel : the language*. Prentice Hall object-oriented series. Prentice Hall, New York, London, Toronto. Bibliogr. p. 509-512. Index. (page 111).
- Milanović, D. D., Misita, M., Tadić, D., and Milanović, D. L. (2010). The design of hybrid system for servicing process support in small businesses. *FME Transactions*, 38(3) :143–149. (page 47).
- Miljković, D. (2011). Fault detection methods : A literature survey. In *MIPRO, 2011 proceedings of the 34th international convention*, pages 750–755. IEEE. (page 41).
- Mingers, J. and Rosenhead, J. (2001). *Rational analysis for a problematic world revisited*, volume 1. John Wiley and Sons Ltd. (page 38).

- Mintzberg, H., Ahlstrand, B., and Lampel, J. (2005). *Strategy Safari : A Guided Tour Through The Wilds of Strategic Management*. Free Press. (page 30).
- Mintzberg, H., Raisinghani, D., and Théorêt, A. (1976). The structure of "unstructured" decision processes. *Administrative Science Quarterly*, 21(2) :246–275. (pages xiii, 39, 43).
- Miser, H. and Quade, E. (1985). *Handbook of Systems Analysis : Overview of Uses, Procedures, Applications, and Practice*. Number v. 1 in Handbook of Systems Analysis. North-Holland. (page 187).
- Mohan, K. and Ramesh, B. (2007). Traceability-based knowledge integration in group decision and negotiation activities. *Decision Support Systems*, 43(3) :968 – 989. Integrated Decision Support. (page 124).
- Monedero, J. (2000). Parametric design : a review and some experiences. *Automation in Construction*, 9(4) :369–377. (page 38).
- Montagna, F. (2011). Decision-aiding tools in innovative product development contexts. *Research in Engineering Design*, 22(2) :63–86. (pages xiii, 39, 40).
- Montgomery, C. (2012). *The Strategist : Be the Leader Your Business Needs*. HarperCollins Publishers Limited. (page 11).
- Moon, S. K., Simpson, T. W., and Kumara, S. R. T. (2008). An agent-based recommender system for developing customized families of products. *Journal of Intelligent Manufacturing*, 20(6) :649. (page 50).
- Moones, E., Yahia, E., and Roucoules, L. (2015). Design process and trace modelling for design rationale capture. In *Joint Conference on Mechanical, Design Engineering & Advanced Manufacturing*. (pages xv, xix, 107, 109, 112, 113, 114, 245).
- Morel, C. (2014). *Les décisions absurdes*. Number v. 1. Editions Gallimard. (page 63).
- Morgan, J. and Liker, J. (2006). *The Toyota Product Development System : Integrating People, Process, and Technology*. Taylor & Francis. (page 16).
- Morris, W. (1969). *The American Heritage dictionary of the English language*. Houghton Mifflin. (page 103).
- Mortensen, N. and Tichem, M. (1994). Observations about decision making in design - a case study. In *Proceedings of WDK workshop evaluation and decision making in design (EVAD)*, Technical University of Denmark, pages 1–6. (page 22).
- Mostafa, S. A., Ahmad, M. S., Mohammed, M. A., and Obaid, O. I. (2012). Implementing an expert diagnostic assistance system for car failure and malfunction. *IJCSI International Journal of Computer Science Issues*, 9(2) :1694–0814. (pages xiii, 46).
- Mouzas, S. (2006). Efficiency versus effectiveness in business networks. *Journal of Business Research*, 59(10) :1124 – 1132. (page 12).
- Murata, T. (1989). Petri nets : Properties, analysis and applications. *Proceedings of the IEEE*, 77(4) :541–580. (page 144).
- Murray, S. R. (1997). Deliberate decision making by aircraft pilots : A simple reminder to avoid decision making under panic. *The International Journal of Aviation Psychology*, 7(1) :83–100. (page 71).
- Nabende, P. and Wanyama, T. (2008). An expert system for diagnosing heavy-duty diesel engine faults. In Sobh, T., editor, *Advances in Computer and Information Sciences and Engineering*, pages 384–389, Dordrecht. Springer Netherlands. (page 45).

- NASA (2010a). *NASA at 50 : Interviews with NASA's Senior Leadership*. NASA SP. Wright, R. and Johnson, S. and Dick, S.J. (page 63).
- NASA (2010b). *NASA Systems Engineering Handbook*. DIANE Publishing Company. (pages xiv, 15, 64, 66, 67).
- NASA (2017). *Nasa systems engineering handbook revision 2*. (pages 183, 184).
- Naur, P. and Randell, B., editors (1969). *Software Engineering : Report of a Conference Sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968, Brussels, Scientific Affairs Division, NATO*. (page 121).
- Nikolopoulos, C. (1997). *Expert Systems : Introduction to First and Second Generation and Hybrid Knowledge Based Systems*. Taylor & Francis. (page 45).
- Noël, F. and Roucoules, L. (2008). The ppo design model with respect to digital enterprise technologies among product life cycle. *International Journal of Computer Integrated Manufacturing*, 21(2) :139–145. (pages 108, 113, 114).
- Noran, O. (2007). A decision support framework for collaborative networks. In Camarinha-Matos, L. M., Afsarmanesh, H., Novais, P., and Analide, C., editors, *Establishing the Foundation of Collaborative Networks*, pages 83–90, Boston, MA. Springer US. (pages xix, 250).
- Nowak, P., Rose, B., Saint-Marc, L., Callot, M., Eynard, B., Gzara-Yesilbas, L., and Lombard, M. (2004). Towards a design process model enabling the integration of product, process and organization. In *5th International Conference on Integrated Design and Manufacturing in Mechanical Engineering, IDMME*, pages 5–7. (pages xv, xix, 107, 112, 113, 124, 243).
- NTSB (2000). *Aircraft Accident Report : Controlled flight into terrain, Korean Air flight 801, Boeing 747300, HL7468, Nimitz Hill, Guam, August 6, 1997*. DIANE Publishing. (page 69).
- Nunes, I. and Jannach, D. (2017). A systematic review and taxonomy of explanations in decision support and recommender systems. *User Modeling and User-Adapted Interaction*, 27(3) :393–444. (page 50).
- Nurmi, P., Martin, M., Flanagan, J. A., et al. (2005). Enabling proactiveness through context prediction. In *Proceedings of the Workshop on Context Awareness for Proactive Systems, Helsinki*, volume 53. (page 175).
- Nutt, P. C. (2001). A taxonomy of strategic decisions and tactics for uncovering alternatives. *European Journal of Operational Research*, 132(3) :505 – 527. (page 44).
- OASIS (2007). *Web Services Business Process Execution Language Version 2.0*. OASIS Standard. Available at : <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>. (page 144).
- OCDE (2005). *La mesure des activités scientifiques et technologiques Manuel d'Oslo Principes directeurs pour le recueil et l'interprétation des données sur l'innovation, 3e édition : Principes directeurs pour le recueil et l'interprétation des données sur l'innovation, 3e édition*. La mesure des activités scientifiques et technologiques. OECD Publishing. (page 13).
- Oehmen, J. and Rebentisch, E. (2010). *Waste in Lean Product Development*. Lean Advancement Initiative (LAI), Massachusetts Institute of Technology. (pages xiii, 22).
- O'Hare, D. (2003). Aeronautical decision making : Metaphors, models, and methods. In *Principles and practice of aviation psychology*, Human factors in transportation., pages 201–237. Lawrence Erlbaum Associates Publishers, Mahwah, NJ, US. (page 69).

- Oldaker, L. (1995). *Pilot decision making, an alternative to judgment training*. XXIV Organisation Scientific et Technique International du Vol à Voile (OSTIV) Congress, Omanara, New Zealand. (page 71).
- Olsen, G. R., Cutkosky, M., Tenenbaum, J. M., and Gruber, T. R. (1994). Collaborative engineering based on knowledge sharing agreements. (page 19).
- OMG (2011). *OMG Unified Modeling Language (OMG UML), Superstructure, Version 2.4.1*. (pages xv, 94, 96, 97, 100, 107, 113, 114, 144).
- OMG (2013). *Business process model and notation (bpmn) version 2.0*. Technical report. Object Management Group, Inc. (OMG). Available at : <http://www.omg.org/spec/BPMN/2.0.2/PDF>. (pages xiii, xiv, xv, xvii, xviii, xix, 4, 8, 56, 80, 86, 107, 113, 144, 148, 151, 180, 249).
- OMG (2016). *Decision Model and Notation (DMN)*. Object Management Group, Inc. (OMG). V1.1. Available at : <http://www.omg.org/spec/DMN/1.1/PDF>. (pages xiv, 79, 80).
- Orasanu, J., Martin, L., Davison, J., and Null, Cynthia, H. (1998). Errors in aviation decision making : Bad decisions or bad luck ? In *Fourth Conference on Naturalistic Decision Making, Warrenton, Virginia, May 29-31, 1998*. (page 68).
- Ouertani, M. Z. (2007). *DEPNET : une approche support au processus de gestion de conflits basée sur la gestion des dépendances de données de conception*. PhD thesis, Université Henri Poincaré-Nancy I. (pages xv, xix, 107, 109, 112, 113, 114, 124, 244).
- Pahl, G. and Beitz, W. (2013). *Engineering design : a systematic approach*. Springer Science & Business Media. (page 39).
- Pahl, G., Beitz, W., Feldhusen, J., and Grote, K. (2007). *Engineering Design : A Systematic Approach*. Solid mechanics and its applications. Springer London. (page 37).
- Pahl, G., Wallace, K., Blessing, L., Beitz, W., and Bauert, F. (2013). *Engineering Design : A Systematic Approach*. Springer London. (page 15).
- Pahng, F., Senin, N., and Wallace, D. (1998). Distribution modeling and evaluation of product design problems. *Computer-Aided Design*, 30(6) :411 – 423. (page 19).
- Paige, R., Ostroff, J., and Brooke, P. (2000). Principles for modeling language design. *Information and Software Technology*, 42(10) :665 – 675. (pages 106, 111).
- Parmigiani, G. (2001). Decision theory : Bayesian. In Smelser, N. J. and Baltes, P. B., editors, *International Encyclopedia of the Social & Behavioral Sciences*, pages 3327 – 3334. Pergamon, Oxford. (page 33).
- Parnell, G. (2016). *Trade-off Analytics : Creating and Exploring the System Tradespace*. Wiley Series in Systems Engineering and Management. Wiley. (page 68).
- Parnell, G., Driscoll, P., and Henderson, D. (2011). *Decision Making in Systems Engineering and Management*. Wiley Series in Systems Engineering and Management. Wiley. (pages xiii, 15, 44, 45, 76, 77).
- Pavković, N., Štorga, M., Bojčetić, N., and Marjanović, D. (2013). Facilitating design communication through engineering information traceability. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 27(02) :105–119. (page 124).
- PAWLAK, Z. (1998). Rough set theory and its applications to data analysis. *Cybernetics and Systems*, 29(7) :661–688. (page 39).

- Pels, H. J. (2006). Classification hierarchies for product data modelling. *Production Planning and Control*, 17(4) :367–377. (page 103).
- Pereira, J. A., Matuszyk, P., Krieter, S., Spiliopoulou, M., and Saake, G. (2017). A feature-based personalized recommender system for product-line configuration. *ACM SIGPLAN Notices*, 52(3) :120–131. (page 50).
- Perrin, O. (2010). *From data integration to Web services composition*. Habilitation à diriger des recherches, Université Nancy II. (page 19).
- Pham, H. (1999). *Software Reliability*. Springer-Verlag, Berlin, Heidelberg. (page 183).
- Phillips-Wren, G. and Ichalkaranje, N. (2008). *Intelligent Decision Making : An AI-Based Approach*. Studies in Computational Intelligence. Springer Berlin Heidelberg. (pages 129, 260).
- Phillips-Wren, G., Mora, M., Forgionne, G., and Gupta, J. (2009). An integrative evaluation framework for intelligent decision support systems. *European Journal of Operational Research*, 195(3) :642 – 652. (pages xiv, 52).
- Phillips-Wren, G., Mora, M., Forgionne, G. A., Garrido, L., and Gupta, J. N. D. (2006). *A Multicriteria Model for the Evaluation of Intelligent Decision-making Support Systems (i-DMSS)*, pages 3–24. Springer London, London. (pages 189, 205).
- Philpotts, M. (1996). An introduction to the concepts, benefits and terminology of product data management. *Industrial Management & Data Systems*, 96(4) :11–17. (page 18).
- Pinel, J. and Barnes, S. (2017). *Biopsychology, Global Edition*. Pearson Education, Limited. (page 34).
- Pizzuti, T., Mirabelli, G., Sanz-Bobi, M. A., and Gómez-González, F. (2014). Food track & trace ontology for helping the food traceability control. *Journal of Food Engineering*, 120 :17 – 30. (page 122).
- PMI (2018). *Guide to the Project Management Body of Knowledge (PMBOK® Guide)–Sixth Edition*. Project Management Institute. (pages 30, 183).
- Pokrovsky, O. (2009a). Operational research approach to decision making. In Byrnes, J., editor, *Unexploded Ordnance Detection and Mitigation*, pages 235–258, Dordrecht. Springer Netherlands. (page 34).
- Pokrovsky, O. (2009b). Operational research approach to decision making. In Byrnes, J., editor, *Unexploded Ordnance Detection and Mitigation*, pages 235–258, Dordrecht. Springer Netherlands. (page 35).
- Pokinska, B., Dahlgaard, J. J., and Antoni, M. (2002). The state of iso 9000 certification : a study of swedish organizations. *The TQM Magazine*, 14(5) :297–306. (page 183).
- Porter, M. (1990). *The competitive advantage of nations : with a new introduction*. The Michael E. Porter Trilogy. Free Press. (page 10).
- Porter, M. (2008). *Competitive Strategy : Techniques for Analyzing Industries and Competitors*. Free Press. (page 10).
- Porter, M. (2011). *Competitive Advantage of Nations : Creating and Sustaining Superior Performance*. Free Press. (page 13).
- Portugal, I., Alencar, P., and Cowan, D. (2017). The use of machine learning algorithms in recommender systems : a systematic review. *Expert Systems with Applications*. (page 47).
- Power, D. (2002). *Decision Support Systems : Concepts and Resources for Managers*. Quorum Books. (page 51).

- Power, D. J. (2008a). *Decision Support Systems : A Historical Overview*, pages 121–140. Springer Berlin Heidelberg, Berlin, Heidelberg. (page 51).
- Power, D. J. (2008b). Understanding data-driven decision support systems. *Information Systems Management*, 25(2) :149–154. (page 51).
- Power, D. J. and Sharda, R. (2007). Model-driven decision support systems : Concepts and research directions. *Decision Support Systems*, 43(3) :1044–1061. (page 51).
- Prajogo, D. I. (2007). The relationship between competitive strategies and product quality. *Industrial Management & Data Systems*, 107(1) :69–83. (page 13).
- Prince, C. and Salas, E. (1993). Training and research for teamwork in the military aircrew. In *Cockpit resource management.*, pages 337–366. Academic Press, San Diego, CA, US. (page 71).
- Pugh, S. (1981). Concept selection : A method that works. In *Proceedings of International Conference on Engineering Design, Heurista, Zürich, 1981*, pages 497–506. (page 38).
- Putman, V. L. and Paulus, P. B. (2009). Brainstorming, brainstorming rules and decision making. *The Journal of creative behavior*, 43(1) :29–40. (page 39).
- Quarante, D. (1994). *Eléments de design industriel*. Collection Universite De Compiegne. Polytechnica. (page 15).
- Quinlan, J. R. (1993). *C4.5 : Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. (page 173).
- Rao, M. (2012). *Knowledge Management Tools and Techniques*. Taylor & Francis. (pages 18, 257).
- Rashid, A. M., Albert, I., Cosley, D., Lam, S. K., McNee, S. M., Konstan, J. A., and Riedl, J. (2002). Getting to know you : Learning new user preferences in recommender systems. In *Proceedings of the 7th International Conference on Intelligent User Interfaces, IUI '02*, pages 127–134, New York, NY, USA. ACM. (page 47).
- Raymond, K. (1995). *Reference Model of Open Distributed Processing (RM-ODP) : Introduction*, pages 3–14. Springer US, Boston, MA. (page 90).
- Rechtin, E. and Maier, M. (2010). *The Art of Systems Architecting, Second Edition*. Systems Engineering. Taylor & Francis. (pages xxi, 89, 90).
- Reich, Y. (1995). A critical review of general design theory. *Research in Engineering Design*, 7(1) :1–18. (page 39).
- Reimann, P., Calvo, R., Yacef, K., and Southavilay, V. (2010). Comprehensive computational support for collaborative learning from writing. In *International Conference on Computers in Education (ICCE), Putrajaya, Malaysia*. (page 149).
- Renzi, C., Leali, F., and Di Angelo, L. (2017). A review on decision-making methods in engineering design for the automotive industry. *Journal of Engineering Design*, 28(2) :118–143. (pages 37, 38, 83).
- Resnick, P. and Varian, H. R. (1997). Recommender systems. *Commun. ACM*, 40(3) :56–58. (page 47).
- Resulaj, A., Kiani, R., Wolpert, D. M., and Shadlen, M. N. (2009). Changes of mind in decision-making. *Nature*, 461 :263. (page 175).
- Ribeiro, A., Silva, A., and da Silva, A. R. (2015). Data modeling and data analytics : a survey from a big data perspective. *Journal of Software Engineering and Applications*, 8(12) :617. (page 108).

- Ricci, F., Rokach, L., and Shapira, B. (2011). Introduction to recommender systems handbook. In *Recommender systems handbook*, pages 1–35. Springer. (pages 47, 48).
- Riley, G., Culbert, C., and Lopez, F. (1989). C language integrated production system. NASA Tech Briefs (ISSN 0145-319X); 13; 11; P. 56. (page 46).
- Ríos, A. (2001). *Traceability Trackability*, pages 65–77. Springer Berlin Heidelberg, Berlin, Heidelberg. (page 121).
- Roach, G. M., Cox, J. J., and Sorensen, C. D. (2005). The product design generator : a system for producing design variants.(author abstract). *International Journal of Mass Customisation*, 1(1) :83. (page 52).
- Roberts, F. (1979). *Measurement Theory : With Applications to Decision Making Utility and the Social Sciences*. Advanced book program. Addison-Wesley Advanced Book Program. (page 35).
- Rodriguez, M., Posse, C., and Zhang, E. (2012). Multiple objective optimization in recommender systems. In *Proceedings of the Sixth ACM Conference on Recommender Systems, RecSys '12*, pages 11–18, New York, NY, USA. ACM. (page 47).
- Rolland, C., Prakash, N., and Benjamen, A. (1999). A multi-model view of process modelling. *Requirements Engineering*, 4(4) :169–187. (page 124).
- Rolstadås, A. (1995). Enterprise modelling for competitive manufacturing. *Control Engineering Practice*, 3(1) :43 – 50. (page 11).
- Rönnbäck, K. (1992). Product development and competitiveness : The experience of Icab. *Resources Policy*, 18(4) :294. (page 12).
- Roozenburg, N. and Eekels, J. (1995). *Product design : fundamentals and methods*. Wiley series in product development. Wiley. (pages xiii, 14).
- Rosenhead, J. (2006). Past, present and future of problem structuring methods. *Journal of the Operational Research Society*, 57(7) :759–765. (page 38).
- Roucoules, L., Yahia, E., Es-Soufi, W., and Tichkiewitch, S. (2016). Engineering design memory for design rationale and change management toward innovation. {CIRP} *Annals - Manufacturing Technology*, 65(1) :193 – 196. (page 177).
- Rouhani, B. D., Mahrin, M. N., Nikpay, F., and Nikfard, P. (2013). A comparison enterprise architecture implementation methodologies. In *2013 International Conference on Informatics and Creative Multimedia*, pages 1–6. (page 92).
- Roy, R., Hinduja, S., and Teti, R. (2008). Recent advances in engineering design optimisation : Challenges and future trends. *CIRP Annals*, 57(2) :697–715. (page 38).
- Roy, S. (2016). *Decision Making and Modelling in Cognitive Science*. Springer India. (page 34).
- Rozinat, A. and Aalst, W. M. P. (2006). *Decision mining in business processes*. BPM Center Report BPM-06-10, BPMcenter.org,. Available at : <http://bpmcenter.org/wp-content/uploads/reports/2006/BPM-06-10.pdf>. (pages 165, 166).
- Ruhe, G. (2003). Software engineering decision support–methodology and applications. *Innovations in decision support systems*, 3 :143–174. (page 44).
- Ruhe, G. and Ngo-The, A. (2004). Hybrid intelligence in software release planning. *Int. J. Hybrid Intell. Syst.*, 1 :99–110. (page 44).

- Rumbaugh, J., Booch, G., and Jacobson, I. (2017). *The unified modeling language reference manual*. Addison Wesley. (page 90).
- Rumbaugh, J., Jacobson, I., and Booch, G. (1999). *The Unified Modeling Language Reference Manual*. Addison-Wesley Longman Ltd., Essex, UK, UK. (page 106).
- Sahin, S., Tolun, M. R., and Hassanpour, R. (2012). Hybrid expert systems : A survey of current approaches and applications. *Expert Systems with Applications*, 39(4) :4609–4617. (page 47).
- Sandler, T. (1992). *Collective action : theory and applications / Todd Sandler*. University of Michigan Press Ann Arbor. (page 19).
- Sarno, R., Sari, P. L. I., Ginardi, H., Sunaryono, D., and Mukhlash, I. (2013). Decision mining for multi choice workflow patterns. In *Computer, Control, Informatics and Its Applications (IC3INA), 2013 International Conference on*, pages 337–342. IEEE. (pages 165, 166).
- SAS (2015). *How Streaming Data Analytics Enables Real-Time Decisions*. White Paper. Available at : https://www.sas.com/en_us/whitepapers/streaming-analytics-enables-real-time-decisions-107716.html. (pages xiv, 58, 258).
- Sawhney, M., Wolcott, R. C., and Arroniz, I. (2006). The 12 different ways for companies to innovate. <https://sloanreview.mit.edu/article/the-different-ways-for-companies-to-innovate/>. (page 13).
- Schach, S. (1999). *Classical and Object-oriented Software Engineering with UML and Java*. Computer science series. WCB/McGraw-Hill. (page 26).
- Schafer, J. B., Frankowski, D., Herlocker, J., and Sen, S. (2007). Collaborative filtering recommender systems. In *The adaptive web*, pages 291–324. Springer. (page 50).
- Scholz-Reiter, B. and Krause, L. (2001). Integriertes produktdatenmanagement als strategische schlüsseltechnologie in industrieunternehmen. In *Information Age Economy*, pages 905–918. Springer. (pages xv, 104).
- Schonberger, R. (1982). *Japanese Manufacturing Techniques : Nine Hidden Lessons in Simplicity*. Free Press. (page 20).
- Schrijver, A. (1998). *Theory of linear and integer programming*. John Wiley & Sons. (page 38).
- Schumpeter, J. (1934). *The Theory of Economic Development : An Inquiry Into Profits, Capital, Credit, Interest, and the Business Cycle*. Economics Third World studies. Transaction Books. (pages 13, 14).
- Schumpeter, J. (1976). *Capitalism, Socialism and Democracy*. Routledge. (page 14).
- SE (2017). *Activate an Efficient & Sustainable Future*. Schneider Electric. (pages xiv, 73, 74).
- Seepersad, C. C., Pedersen, K., Emblemsvåg, J., Bailey, R., Allen, J. K., and Mistree, F. (2006). The validation square : how does one verify and validate a design method? *Decision making in engineering Design*, pages 303–314. (pages xviii, 188, 189).
- Seletos, T. and Salmon, J. (2017). Analysis of integrating a production design generator with a business decision support system for system-level decision making. *Systems*, 5 :14. (pages xiv, 52, 53).
- Shah, A. K. and Oppenheimer, D. M. (2008). Heuristics made easy : An effort-reduction framework. *Psychological bulletin*, 134(2) :207. (page 36).
- Shaw, N., Bloor, M., and de Pennington, A. (1989). Product data models. *Research in Engineering Design*, 1(1) :43–50. (page 104).

- Shen, L., Peng, X., and Zhao, W. (2009). A comprehensive feature-oriented traceability model for software product line development. In *2009 Australian Software Engineering Conference*, pages 210–219. IEEE. (pages [xix](#), [124](#), [254](#)).
- Shim, J. P., Warkentin, M., Courtney, J. F., Power, D. J., Sharda, R., and Carlsson, C. (2002). Past, present, and future of decision support technology. *Decision support systems*, 33(2) :111–126. (pages [50](#), [52](#), [129](#)).
- SHUBHAM, J. (2017). *Introduction to Pseudo-Labeling : A Semi-Supervised learning technique*. Analytics Vidhya. Available at : <https://www.analyticsvidhya.com/blog/2017/09/pseudo-labeling-semi-supervised-learning-technique/>. (pages [xvi](#), [141](#)).
- Simon, H. (1960). *The New Science of Management Decision*. Ford distinguished lectures, v. 3. Harper & Row. (pages [29](#), [43](#), [50](#)).
- Simon, H. (1990). *Reason in Human Affairs*. Stanford University Press. (page [33](#)).
- Simon, H. A. (1967). Motivational and emotional controls of cognition. *Psychological review*, 74(1) :29. (page [33](#)).
- Simon, H. A. (1980). Cognitive science : The newest science of the artificial. *Cognitive Science*, 4(1) :33 – 46. (page [50](#)).
- Simpson, A. P. (2010). *Decision making in energy : advancing technical, environmental, and economic perspectives*. Ph.d. thesis, Stanford University. (page [73](#)).
- Sirikrai, S. B. and Tang, J. C. (2006). Industrial competitiveness analysis : Using the analytic hierarchy process. *The Journal of High Technology Management Research*, 17(1) :71 – 83. (page [14](#)).
- Smith, J. M. and Smith, D. C. P. (1977). Database abstractions : Aggregation and generalization. *ACM Trans. Database Syst.*, 2(2) :105–133. (page [103](#)).
- Snieder, R. and Lerner, K. (2009). *The Art of Being a Scientist : A Guide for Graduate Students and Their Mentors*. The Art of Being a Scientist : A Guide for Graduate Students and Their Mentors. Cambridge University Press. (page [3](#)).
- Sobek, D., Ward, A., and Liker, J. (1999). "Toyota's Principles of Set-Based Concurrent Engineering". *Sloan Management Review*, 40(2) :67–83. (page [17](#)).
- Sobie, C., Freitas, C., and Nicolai, M. (2017). Simulation-driven machine learning : Bearing fault classification. *Mechanical Systems and Signal Processing*, 99 :403 – 419. (page [42](#)).
- Sobieszczanski-Sobieski, J. and Haftka, R. T. (1997). Multidisciplinary aerospace design optimization : survey of recent developments. *Structural optimization*, 14(1) :1–23. (page [35](#)).
- Sohlberg, B. (1998). *Supervision and Control for Industrial Processes : Using Grey Box Models, Predictive Control, and Fault Detection Methods*. Advances in industrial control. Springer. (pages [40](#), [41](#)).
- Sohlenius, G. (1992). Concurrent engineering. *{CIRP} Annals - Manufacturing Technology*, 41(2) :645 – 655. (pages [18](#), [38](#)).
- Soll, H., Proske, S., Hofinger, G., and Steinhardt, G. (2016). Decision-making tools for aeronautical teams : FOR-DEC and beyond. *Aviation Psychology and Applied Human Factors*, 6(2) :101–112. (page [70](#)).
- Song, M., Günther, C. W., and Van der Aalst, W. M. (2008). Trace clustering in process mining. In *International Conference on Business Process Management*, pages 109–120. Springer. (pages [158](#), [159](#), [160](#)).

- Sriram, D. and Adey, R. (1987). Knowledge based expert systems in engineering : Planning and design. (page 38).
- Sriram, R. D. (1997). Intelligent systems for engineering : A knowledge-based approach. (page 38).
- Stamatis, D. H. (2003). *Failure mode and effect analysis : FMEA from theory to execution*. ASQ Quality Press. (page 39).
- Stanovich, K. (2010). *Decision Making and Rationality in the Modern World*. Fundamentals of cognition series. Oxford University Press. (page 32).
- Steck, H. (2013). Evaluation of recommendations : rating-prediction and ranking. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 213–220. ACM. (page 47).
- Steinwart, I. and Christmann, A. (2008). *Support Vector Machines*. Information Science and Statistics. Springer New York. (page 39).
- Stern, D. H., Herbrich, R., and Graepel, T. (2009). Matchbox : large scale online bayesian recommendations. In *Proceedings of the 18th international conference on World wide web*, pages 111–120. ACM. (page 50).
- Stevens, W., Fenner, B., and Rudoff, A. (2004). *UNIX Network Programming : The Sockets Networking API*. Number v. 1 in Addison-Wesley professional computing series. Addison-Wesley. (pages 132, 260).
- Stoner, J. (1961). *A Comparison of Individual and Group Decisions Involving Risk*. Sloan School of Management, Massachusetts Institute of Technology. (page 29).
- Sudarsan, R., Fenves, S., Sriram, R., and Wang, F. (2005). A product information modeling framework for product lifecycle management. *Computer-Aided Design*, 37(13) :1399 – 1411. (pages xix, 109, 114, 251).
- Suh, N. (2001). *Axiomatic Design : Advances and Applications*. MIT-Pappalardo series in mechanical engineering. Oxford University Press. (pages 17, 37, 38, 39).
- Suhonen, N., Estola, M., Lindén, M., and Saastamoinen, J. (2007). Normative and descriptive theories of decision making under risk : A short review. (page 34).
- Suparno and Carrie, A. S. (1997). *GRAI Method and Simulation : An Approach for Assessment of the Production Planning and Control Systems*, pages 163–168. Macmillan Education UK, London. (page 107).
- Suppes, P. (2012). *Axiomatic Set Theory*. Dover Books on Mathematics. Dover Publications. (page 39).
- Sutton, R. S., Barto, A. G., et al. (1998). *Reinforcement learning : An introduction*. MIT press. (pages xvi, 142).
- Taguchi, G. (1987). *System of experimental design : engineering methods to optimize quality and minimize costs*. Number v. 1 and 2 in System of Experimental Design : Engineering Methods to Optimize Quality and Minimize Costs. UNIPUB/Kraus International Publications. (page 37).
- Takagi, T. and Sugeno, M. (1993). Fuzzy identification of systems and its applications to modeling and control. In *Readings in Fuzzy Sets for Intelligent Systems*, pages 387–403. Elsevier. (page 42).
- Takemura, K. (2014). *Behavioral Decision Theory : Psychological and Mathematical Descriptions of Human Choice Behavior*. Springer Japan. (page 33).
- Takeuchi, N., Wakabayashi, M., and Chen, Z. (2003). The strategic hrn configuration for competitive advantage : Evidence from japanese firms in china and taiwan. *Asia Pacific Journal of Management*, 20(4) :447–480. (page 11).

- Tamiz, M., Jones, D., and Romero, C. (1998). Goal programming for decision making : An overview of the current state-of-the-art. *European Journal of Operational Research*, 111(3) :569 – 581. (page 38).
- Tang, A., Jin, Y., and Han, J. (2007). A rationale-based architecture model for design traceability and reasoning. *Journal of Systems and Software*, 80(6) :918 – 934. (page 124).
- Theroude, F. (2002). *Formalisme et système pour la représentation et la mise en oeuvre des processus de pilotage des relations entre donneurs d'ordres et fournisseurs*. PhD thesis. Thèse de doctorat dirigée par Haurat, Alain et Braesch, Christian Génie industriel Grenoble, INPG 2002. (page 18).
- Thom, L. H., Iochpe, C., and Mitschang, B. (2005). Tmbp : A transactional metamodel for business process modeling based on organizational structure aspects. In *CAiSE Short Paper Proceedings*. Citeseer. (pages xv, xix, 107, 112, 113, 247).
- Thomas, J. (2009). *Data-Exchange Standards and International Organizations : Adoption and Diffusion : Adoption and Diffusion*. Advances in IT Standards and Standardization Research :. Information Science Reference. (page 21).
- Thorpe, M., Holm, J., van den Boer, G., and Redbooks, I. (2014). *Discovering the Decisions within Your Business Processes using IBM Blueworks Live*. IBM Redbooks. (pages xiv, 56, 57).
- Tichem, M. (1997). *A Design Coordination Approach to Design For X*. PhD thesis. (page 17).
- Tomiyaama, T., Gu, P., Jin, Y., Lutters, D., Kind, C., and Kimura, F. (2009). Design methodologies : Industrial and educational applications. *CIRP Annals*, 58(2) :543 – 565. (pages 15, 17, 18, 37, 39).
- Truong, B. A., Lee, Y.-K., and Lee, S.-Y. (2005). Modeling uncertainty in context-aware computing. In *Fourth Annual ACIS International Conference on Computer and Information Science (ICIS'05)*, pages 676–681. (page 175).
- Tulleken, H. J. (1993). Grey-box modelling and identification using physical knowledge and bayesian techniques. *Automatica*, 29(2) :285 – 308. (page 41).
- Turban, E., Aronson, J., and Liang, T. (2005). *Decision Support Systems and Intelligent Systems*. Prentice Hall. (page 45).
- Ullah, I., Tang, D., and Yin, L. (2016). Engineering product and process design changes : A literature overview. *Procedia CIRP*, 56 :25 – 33. The 9th International Conference on Digital Enterprise Technology - Intelligent Manufacturing in the Knowledge Economy Era. (page 174).
- Ullman, D. (2003). *The mechanical design process*. McGraw-Hill Series in Mechanical Engineering. McGraw-Hill Higher Education. (pages 15, 20).
- Ullman, D. G., Dietterich, T. G., and Stauffer, L. A. (1988). A model of the mechanical design process based on empirical data. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 2(1) :33–52. (page 37).
- Ulrich, K. and Eppinger, S. (2015). *Product Design and Development*. McGraw-Hill Education. (page 15).
- Umeda, Y. and Tomiyama, T. (1997). Functional reasoning in design. *IEEE Expert*, 12(2) :42–48. (page 38).
- U.S.NRC (2012). *A Proposed Risk Management Regulatory Framework : A Report to NRC Chairman Gregory B. Jaczko from the Risk Management Taskforce*. United States Nuclear Regulatory Commission. (pages xiv, 70, 71, 72, 73).
- van der Aalst, W. (2011). *Process Mining : Discovery, Conformance and Enhancement of Business Processes*. Springer Berlin Heidelberg. (pages 143, 144, 146, 158).

- Van Der Aalst, W. (2012). Process mining : overview and opportunities. *ACM Transactions on Management Information Systems (TMIS)*, 3(2) :7. (pages xvii, 143, 146).
- van der Aalst, W., Reijers, H., Weijters, A., van Dongen, B., de Medeiros, A. A., Song, M., and Verbeek, H. (2007). Business process mining : An industrial application. *Information Systems*, 32(5) :713 – 732. (page 19).
- Van der Aalst, W. M. (2005). Business alignment : using process mining as a tool for delta analysis and conformance testing. *Requirements Engineering*, 10(3) :198–211. (pages xvi, xvii, 143, 146).
- Van der Aalst, W. M. and de Medeiros, A. K. A. (2005). Process mining and security : Detecting anomalous process executions and checking process conformance. *Electronic Notes in Theoretical Computer Science*, 121 :3–21. (pages xvii, 145).
- van der Aalst, W. M. and Gunther, C. W. (2007). Finding structure in unstructured processes : The case for process mining. In *Application of concurrency to system design, 2007. ACSD 2007. Seventh international conference on*, pages 3–12. IEEE. (pages 143, 145, 146, 260).
- van der Aalst, W. M. P., Rubin, V., Verbeek, H. M. W., van Dongen, B. F., Kindler, E., and Günther, C. W. (2008). Process mining : a two-step approach to balance between underfitting and overfitting. *Software & Systems Modeling*, 9(1) :87. (pages 143, 260).
- van Dongen, B. F., Alves de Medeiros, A. K., and Wen, L. (2009). *Process Mining : Overview and Outlook of Petri Net Discovery Algorithms*, pages 225–242. Springer Berlin Heidelberg, Berlin, Heidelberg. (page 161).
- van Dongen, B. F. and Van der Aalst, W. M. (2005). A meta model for process mining data. *EMOI-INTEROP*, 160 :30. (pages xvii, 153, 155).
- Van Duin, S. and Bakhshi, N. (2017). *Artificial Intelligence Defined : The most used terminology around AI*. Deloitte. (pages xvi, 133).
- Venkatasubramanian, V., Rengaswamy, R., Kavuri, S. N., and Yin, K. (2003). A review of process fault detection and diagnosis : Part iii : Process history based methods. *Computers & chemical engineering*, 27(3) :327–346. (pages xiii, 42).
- Vui-Yee, K. (2015). The impact of strategic human resource management on employee outcomes in private and public limited companies in malaysia. *Journal of Human Values*, 21(2) :75–86. (page 11).
- Vuori, V. (2006). Methods of defining business information needs. *Frontiers of e-Business Research ICEB+ eBRF*, 2006 :311–319. (page 132).
- Wald, A. (1950). *Statistical decision functions*. Wiley publications in statistics. Wiley. (page 35).
- Wald, A. (1992). *Statistical Decision Functions*, pages 342–357. Springer New York, New York, NY. (page 34).
- Wang, Y. (2007). *Software Engineering Foundations : A Software Science Perspective*. Software Engineering Series. Taylor & Francis. (page 44).
- Wang, Y. (2008). *Novel Approaches in Cognitive Informatics and Natural Intelligence*. Advances in cognitive informatics and natural intelligence (ACINI) series. Information Science Reference. (pages 32, 34).
- Wang, Y. and Ruhe, G. (2007). The cognitive process of decision making. *International Journal of Cognitive Informatics and Natural Intelligence*, 1 :73–85. (pages xxi, 32, 36).

- Wang, Y., Wang, Y., Patel, S., and Patel, D. (2006). A layered reference model of the brain (lrmb). *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 36(2) :124–133. (page 34).
- Weske, M. (2012). *Business Process Management : Concepts, Languages, Architectures*. Springer Berlin Heidelberg. (page 19).
- Westgard, J. O. and Lott, J. A. (1981). Precision and accuracy : Concepts and assessment by method evaluation testing. *CRC Critical Reviews in Clinical Laboratory Sciences*, 13(4) :283–330. (pages 74, 258).
- Wielinga, B., Van de Velde, W., Schreiber, G., and Akkermans, H. (1993). Towards a unification of knowledge modelling approaches. In David, J.-M., Krivine, J.-P., and Simmons, R., editors, *Second Generation Expert Systems*, pages 299–335, Berlin, Heidelberg. Springer Berlin Heidelberg. (page 39).
- Wiering, M. and van Otterlo, M. (2012). *Reinforcement Learning : State-of-the-Art*. Adaptation, Learning, and Optimization. Springer Berlin Heidelberg. (page 142).
- Willsky, A. S. (1976). A survey of design methods for failure detection in dynamic systems. *Automatica*, 12(6) :601 – 611. (page 41).
- Wilson, R. and Keil, F. (2001). *The MIT Encyclopedia of the Cognitive Sciences*. Bradford Books. MIT Press. (pages 32, 34).
- Wimmer, A., Mehlaui, J. I., and Klein, T. (2003). Object oriented product meta-model for the financial services industry. In *Proceedings of the 2nd Interdisciplinary World Congress on Mass Customization and Personalization (MCPC'03)*. (pages xix, 110, 114, 253).
- Wingate, G. (2003). *Computer Systems Validation : Quality Assurance, Risk Management, and Regulatory Compliance for Pharmaceutical and Healthcare Companies*. Taylor & Francis. (page 183).
- Witte, E. (1972). Field research on complex decision-making processes - the phase theorem. *International Studies of Management & Organization*, 2(2) :156–182. (page 43).
- Witten, I. H., Frank, E., Trigg, L., Hall, M., Holmes, G., and Cunningham, S. J. (1999). Weka : Practical machine learning tools and techniques with java implementations. (page 172).
- Wohl, J. G. (1981). Force management decision requirements for air force tactical command and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(9) :618–639. (page 71).
- Wu, C.-W., Lin, Y.-F., Yu, P. S., and Tseng, V. S. (2013). Mining high utility episodes in complex event sequences. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 536–544. ACM. (pages 147, 260).
- XIANG, Y., LIU, J., YANG, W., and HUANG, C. (2015). Active energy management strategies for active distribution system. *Journal of Modern Power Systems and Clean Energy*, 3(4) :533–543. (page 74).
- Yannou, B. (2008). *La conception industrielle de produits : Spécifications, déploiement et maîtrise des performances*. Collection Productique. Hermès science publications-Lavoisier. (page 18).
- Yoshikawa, H. (1981). *General Design Theory and a CAD system*. in Sata, T. and Warman, E., editors, *Man-Machine Communication in CAD/CAM*, Proceedings of The IFIP WG5.2 5.3 Working Conference 1980 (Tokyo), pages 35–57. North-Holland, Amsterdam. (page 37).
- Yue, H. H. and Qin, S. J. (2001). Reconstruction-based fault identification using a combined index. *Industrial & Engineering Chemistry Research*, 40(20) :4403–4414. (page 41).

- Z, M. and Lior, R. (2014). *Data Mining With Decision Trees : Theory And Applications (2nd Edition)*. Series In Machine Perception And Artificial Intelligence. World Scientific Publishing Company. (page 39).
- Zachary, W., Wherry, R., Glenn, F., and Hopson, J. (1982). Decision situations, decision processes, and decision functions : Towards a theory-based framework for decision-aid design. In *Proceedings of the 1982 Conference on Human Factors in Computing Systems, CHI '82*, pages 355–358, New York, NY, USA. ACM. (page 30).
- Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3) :276–292. (pages xv, 105, 123, 124, 125).
- Zadeh, L. (1975). The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3) :199 – 249. (page 35).
- Zavadskas, E. K., Turskis, Z., and Kildienė, S. (2014). State of art surveys of overviews on mcdm/madm methods. *Technological and Economic Development of Economy*, 20(1) :165–179. (page 37).
- Zhang, G., Lu, J., and Gao, Y. (2015). *Multi-Level Decision Making : Models, Methods and Applications*. Springer Publishing Company, Incorporated. (page 35).
- Zhang, S., Yao, L., and Sun, A. (2017). Deep learning based recommender system : A survey and new perspectives. *arXiv preprint arXiv :1707.07435*. (page 50).
- Zhao, Q. and Bhowmick, S. S. (2003a). Association rule mining : A survey. *Nanyang Technological University, Singapore*. (page 135).
- Zhao, Q. and Bhowmick, S. S. (2003b). Sequential pattern mining : A survey. *Technical Report CAIS Nanyang Technological University Singapore*, 1 :26. (pages 135, 137).
- Zhu, X. (2006). Semi-supervised learning literature survey. *Computer Science, University of Wisconsin-Madison*, 2(3) :4. (pages 140, 141).
- Ziebart, B. D., Maas, A. L., Dey, A. K., and Bagnell, J. A. (2008). Navigate like a cabbie : Probabilistic reasoning from observed context-aware behavior. In *Proceedings of the 10th International Conference on Ubiquitous Computing, UbiComp '08*, pages 322–331, New York, NY, USA. ACM. (page 175).
- Zimmermann, H. J. (1986). *Fuzzy Sets, Decision Making and Expert Systems*. Kluwer, B.V., Deventer, The Netherlands, The Netherlands. (page 35).
- Zio, E. and Pedroni, N. (2012). Overview of risk-informed decision-making processes . FONSCI. (page 63).
- Zsombok, C. and Klein, G. (2014). *Naturalistic Decision Making. Expertise : Research and Applications Series*. Taylor & Francis. (page 35).

Annexe : Méta-modèles étudiés de processus, de produit et de traçabilité

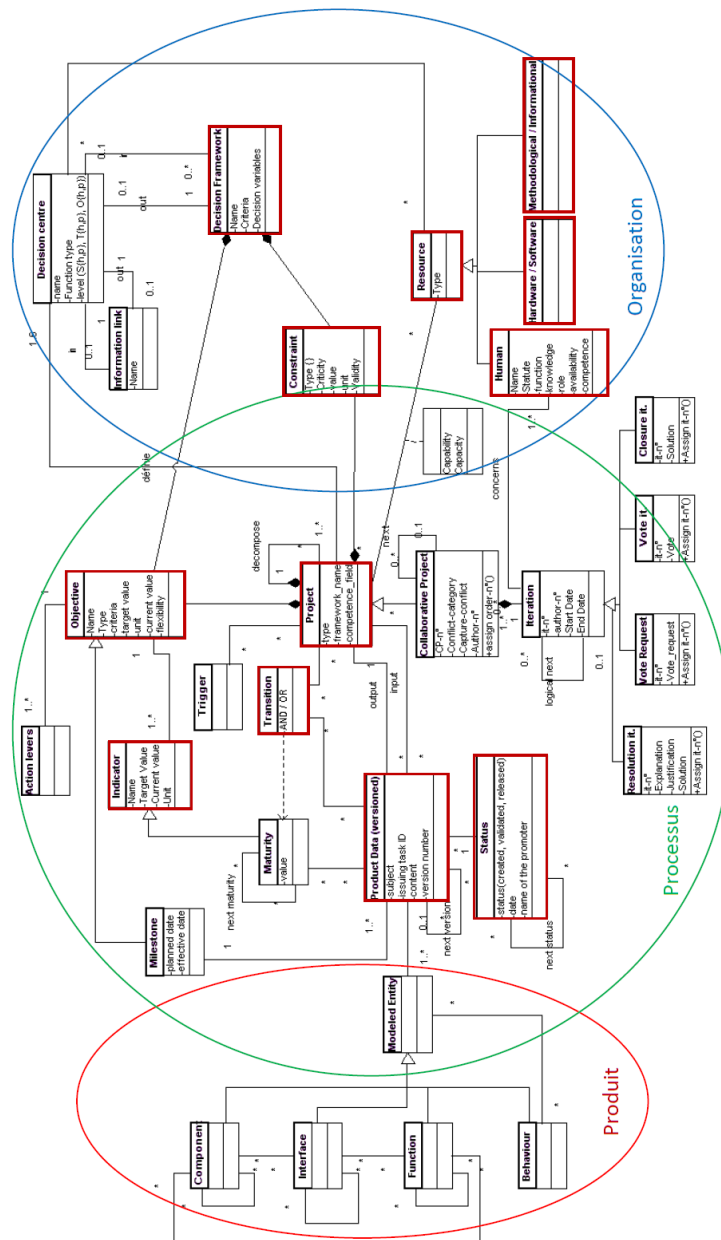


FIGURE 2 – Modèle PPO (Produit-Processus-Organisation) de Nowak et al. [2004]

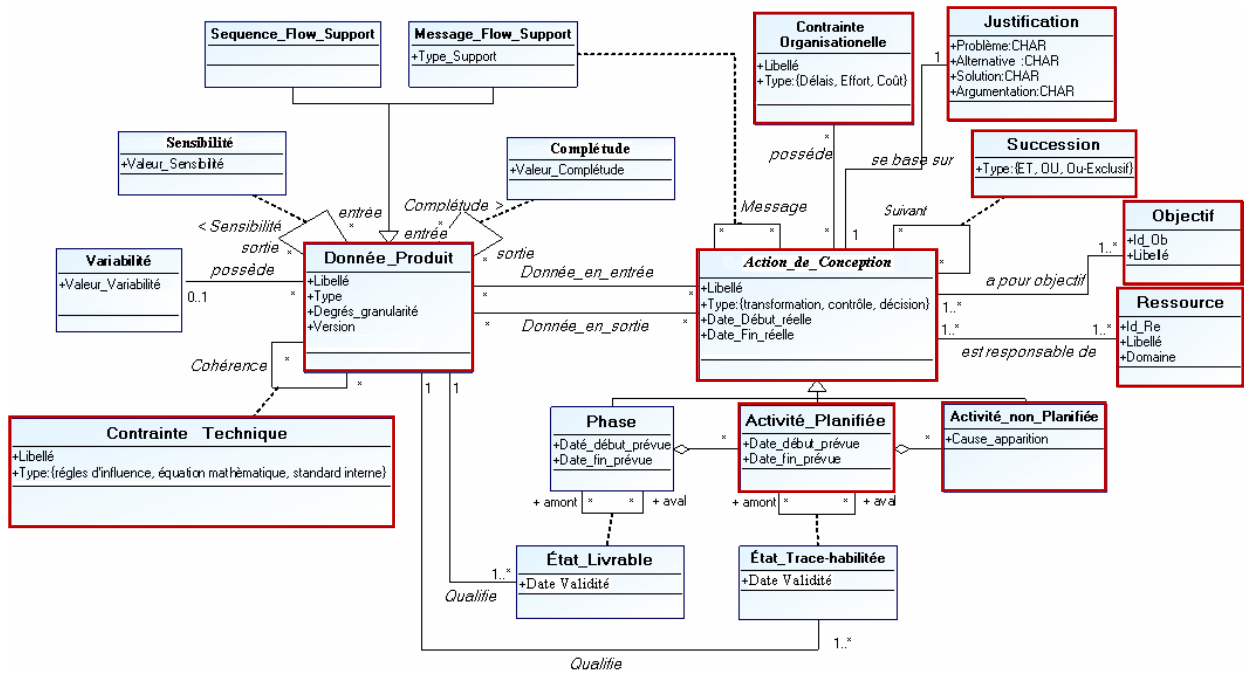


FIGURE 3 – Méta-modèle de processus pour la gestion des conflits [Ouertani, 2007]

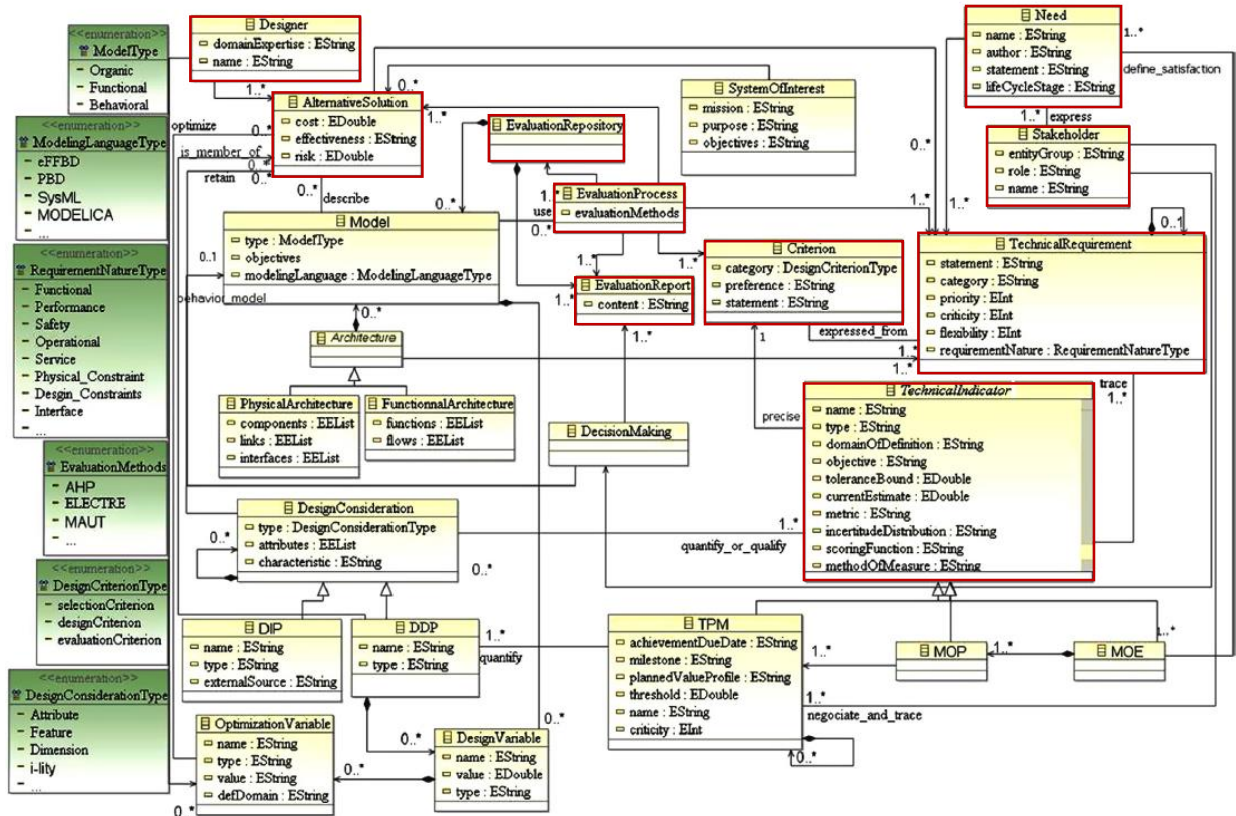


FIGURE 4 – Modèle de donnée pour la caractérisation de l'évolution de donnée [Couturier et al., 2014]

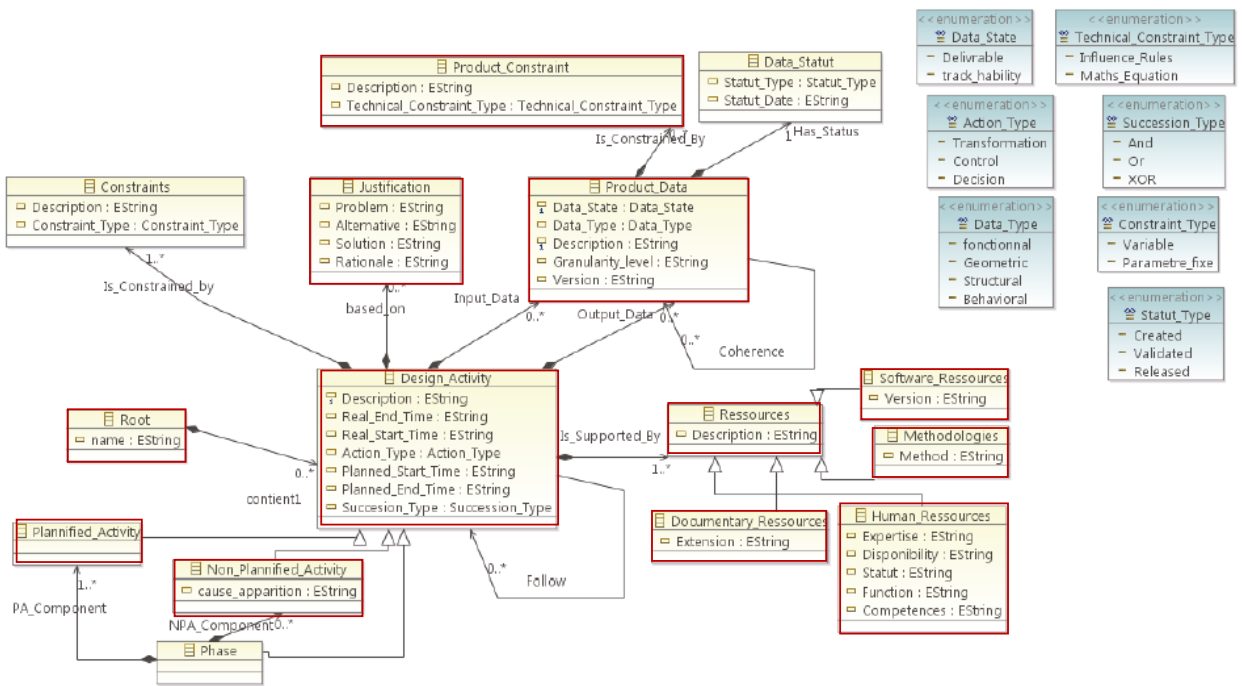


FIGURE 5 – Méta-modèle pour l’identification des changements [Moones et al., 2015]

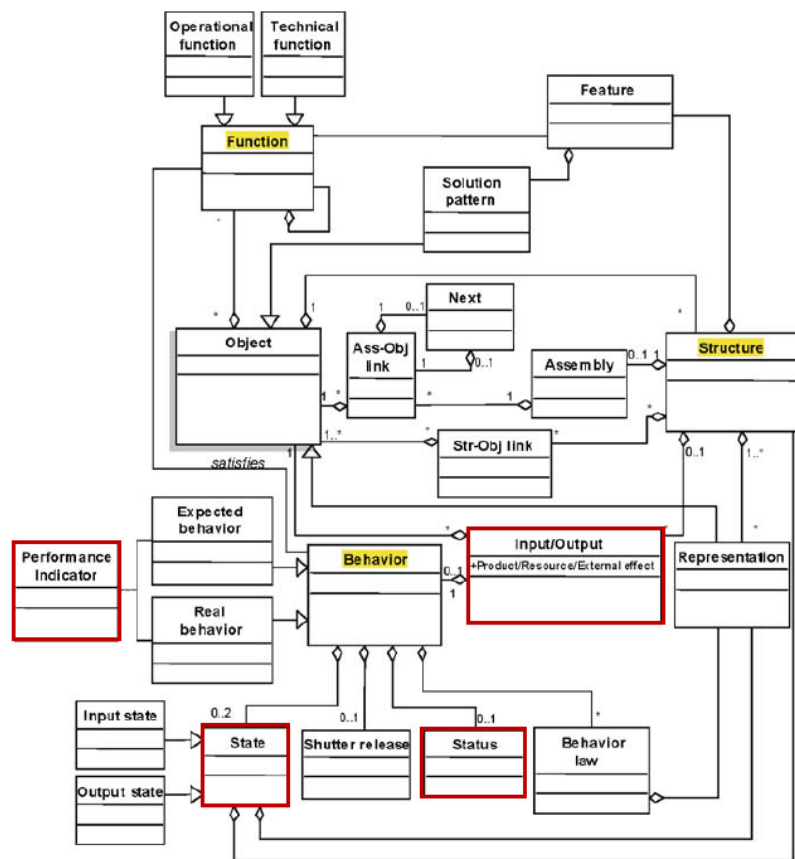


FIGURE 6 – Modèle FBS-PPRE de Labrousse and Bernard [2008]

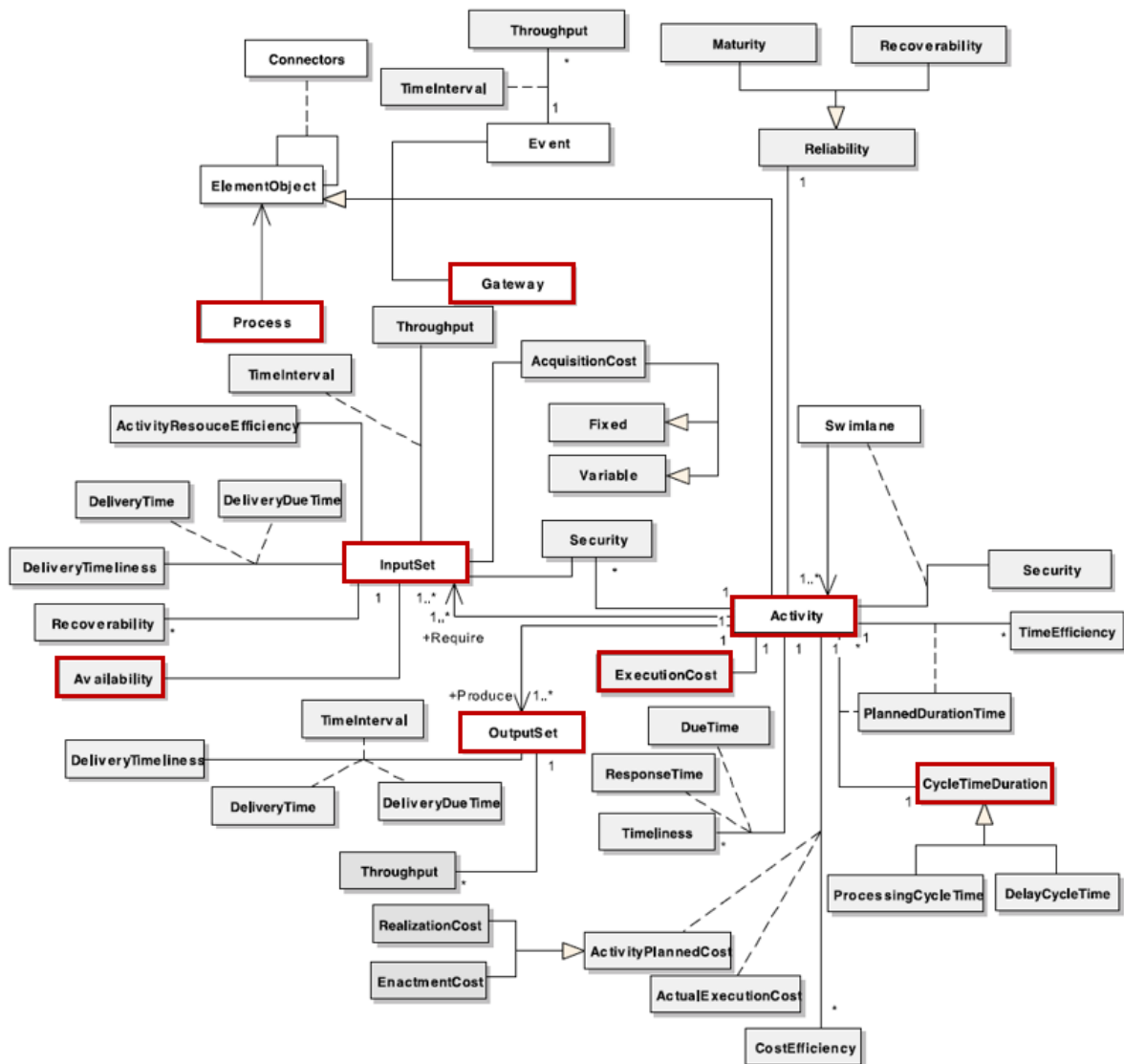


FIGURE 7 – Méta-modèle de processus orienté qualité [Heidari et al., 2011]

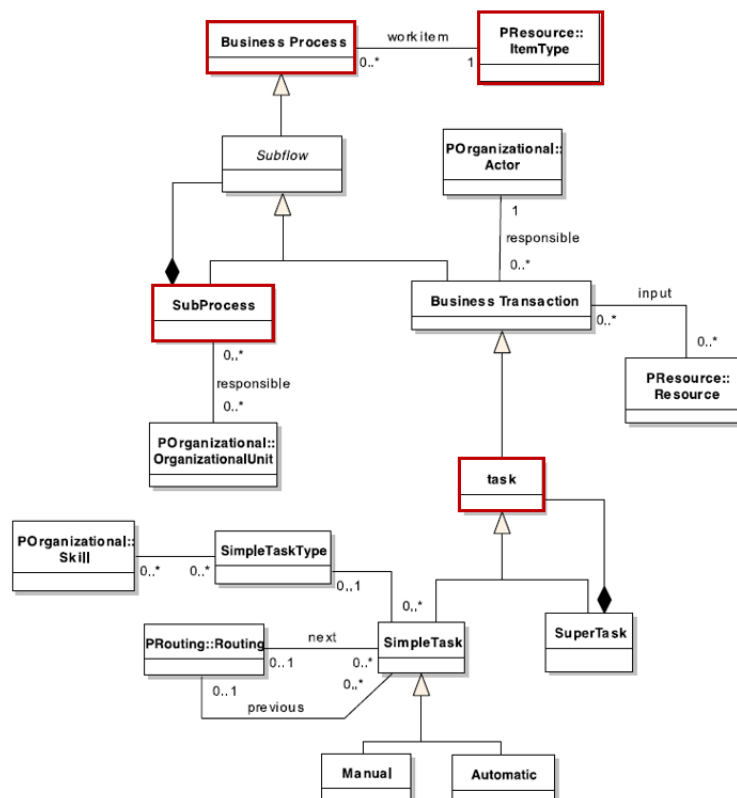
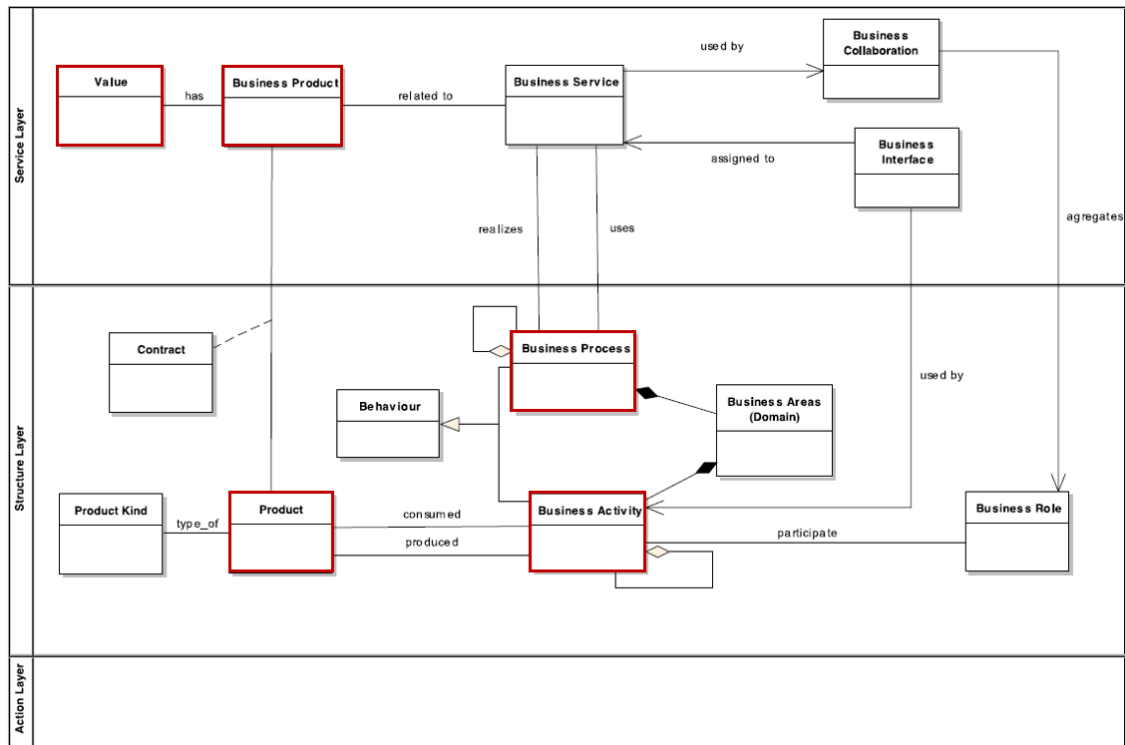
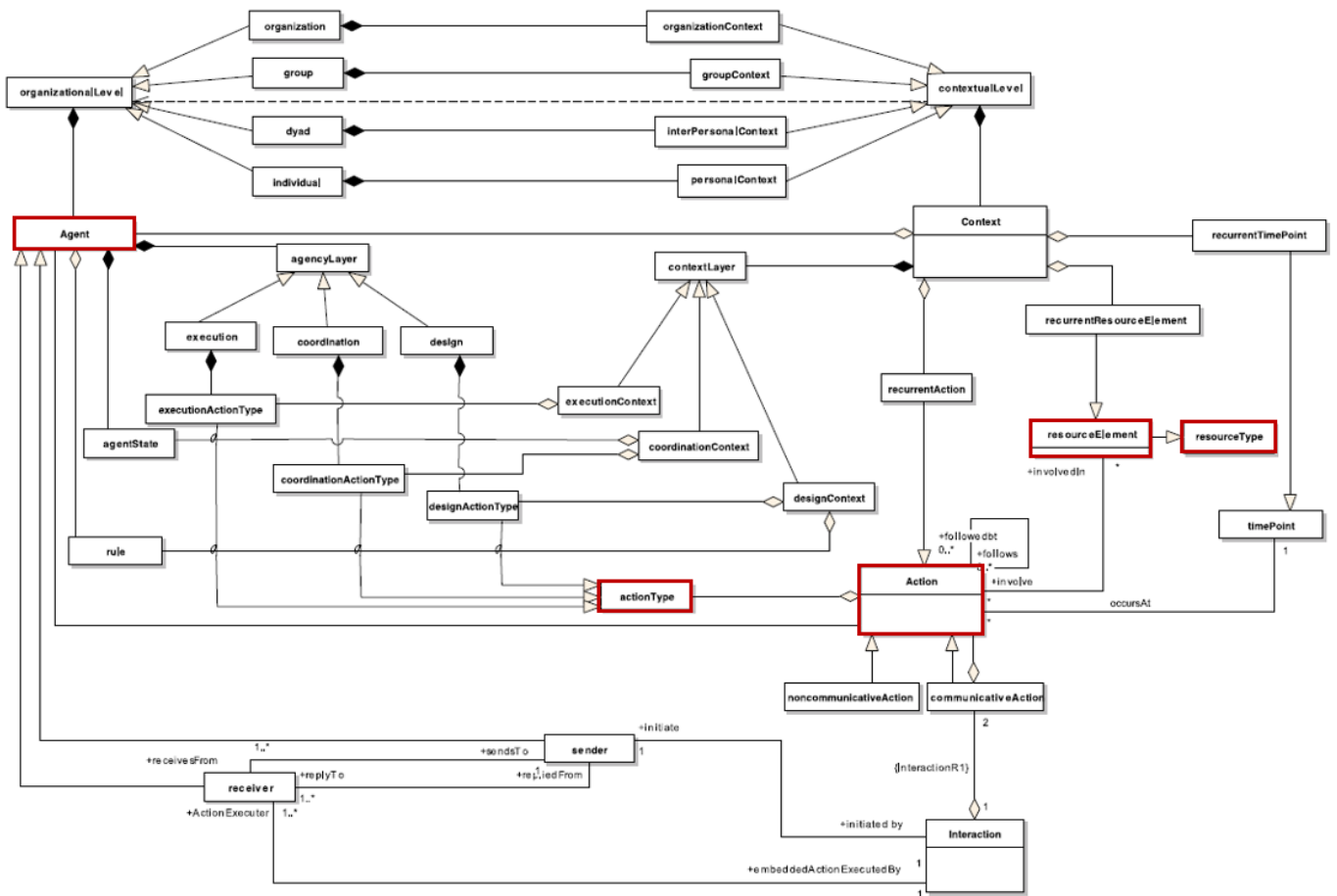


FIGURE 8 – Méta-modèle transactionnel pour la modélisation des processus basée sur la structure organisationnelle [Thom et al., 2005]



9.a. Les trois couches du méta-modèle BPPA [Martins and Zacarias, 2015]



9.b. Couche "Action" du méta-modèle BPPA [Martins and Zacarias, 2015]

FIGURE 9 – Méta-modèle BPPA pour modéliser les processus en se basant sur les pratiques quotidiennes [Martins and Zacarias, 2015]

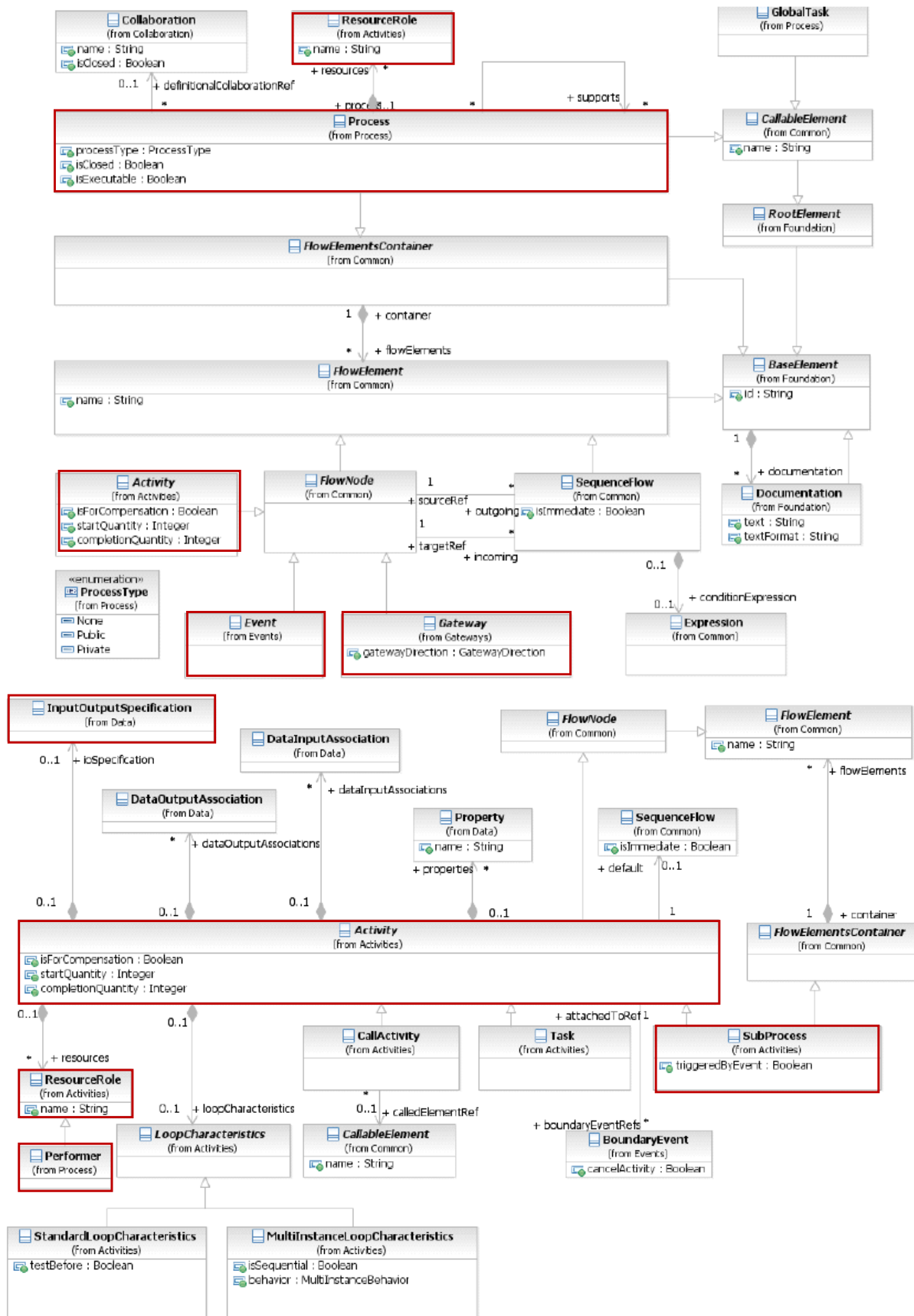


FIGURE 10 – Méta-modèle BPMN [OMG, 2013]

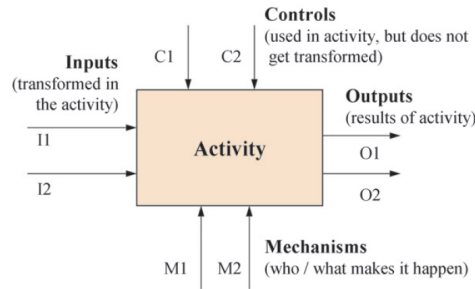


FIGURE 11 – Formalisme de modélisation IDEF0 [Noran, 2007]

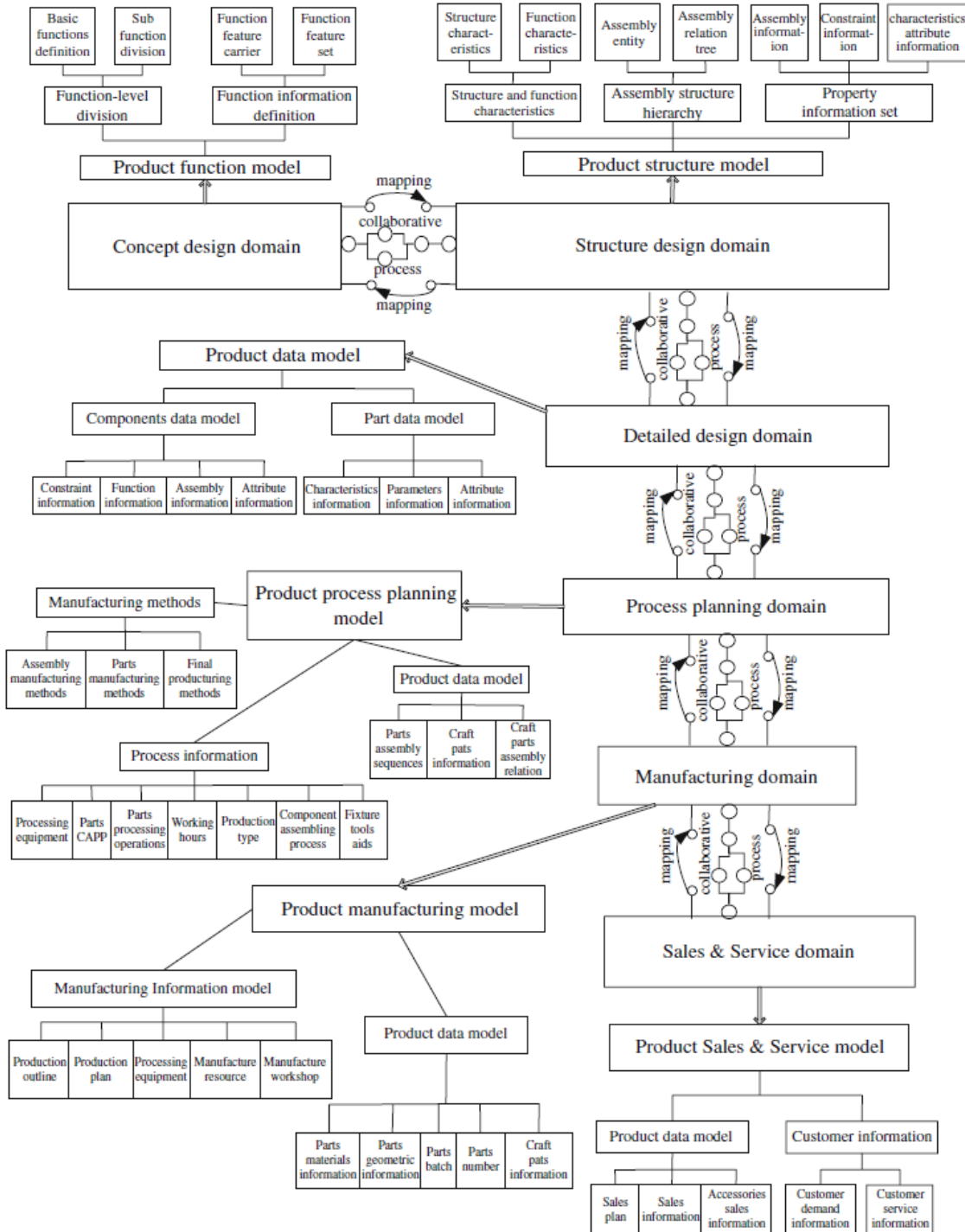
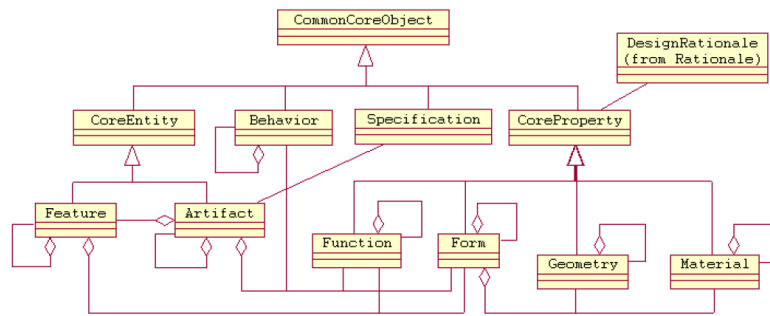
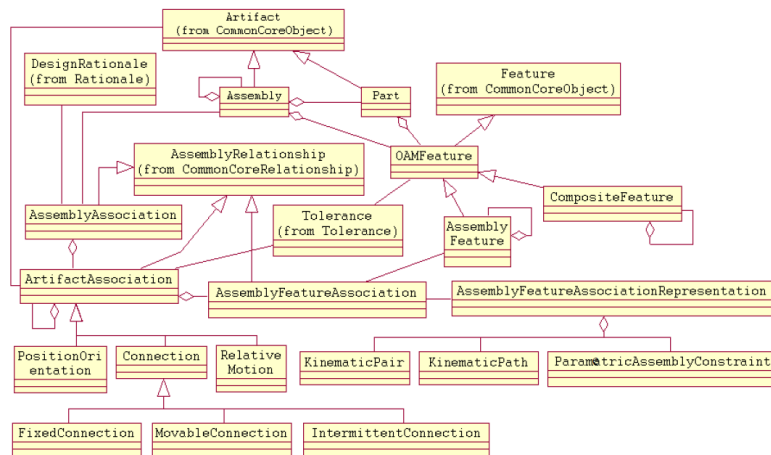


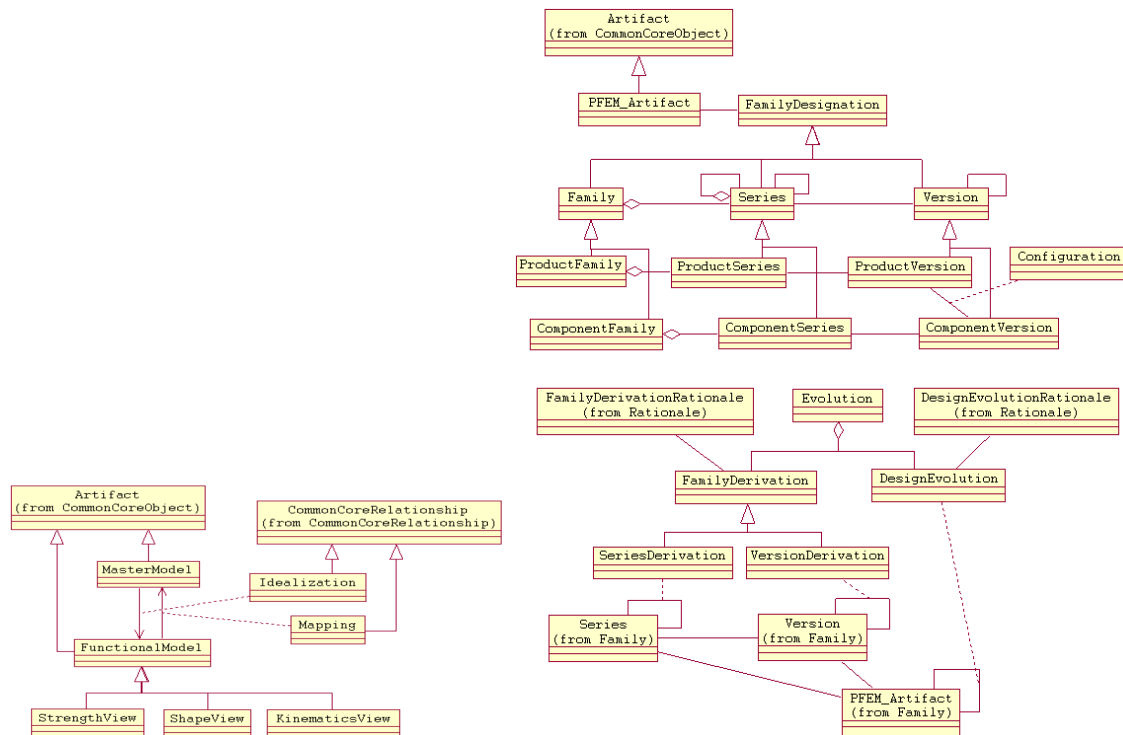
FIGURE 12 – Modèle de données produit basé sur le domaine [Li et al., 2011]



13.a. Premier composant : Modèle produit de base



13.b. Deuxième composant : Modèle d'assemblage ouvert



13.c. Troisième composant : Modèle d'intégration de la conception-analyse

13.d. Quatrième composant : Modèle d'évolution de la famille du produit

FIGURE 13 – Framework de modélisation des données produit de Sudarsan et al. [2005]

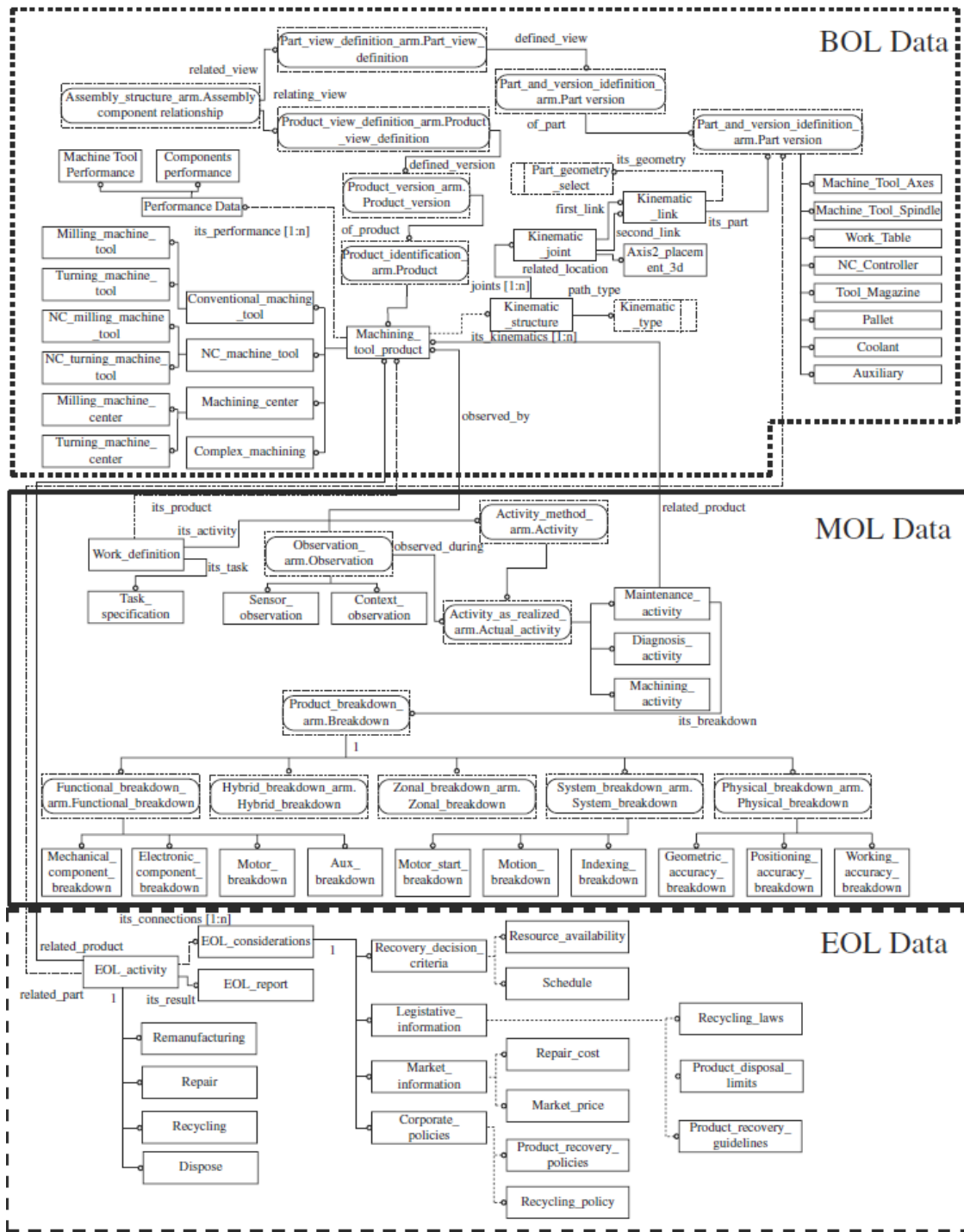


FIGURE 14 – Modèle de produit pour le système UPLS de Lee and Suh [2009]

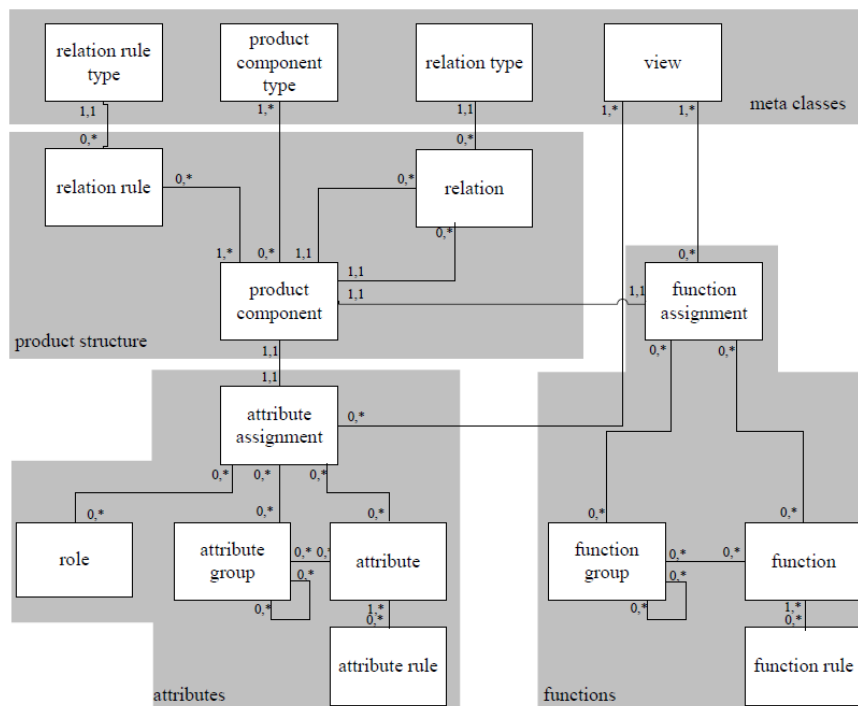


FIGURE 15 – Méta-modèle décrivant les concepts de base pour produire un modèle de produit orienté objet [Wimmer et al., 2003]

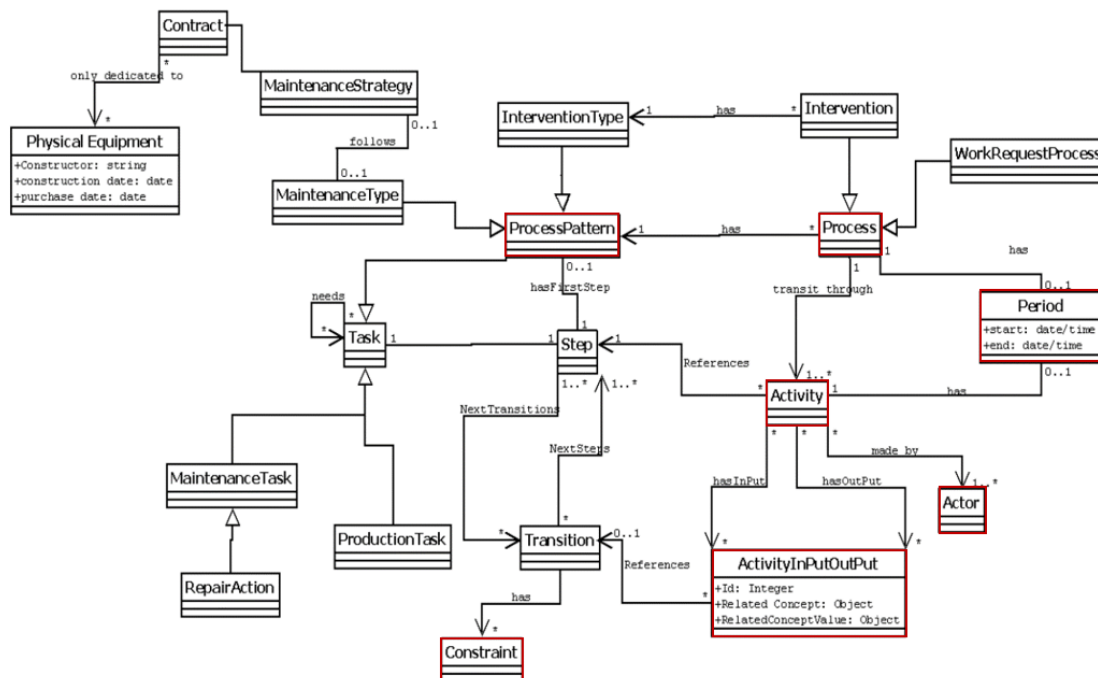


FIGURE 16 – Méta-modèle de trace pour la maintenance [Karray, 2012]

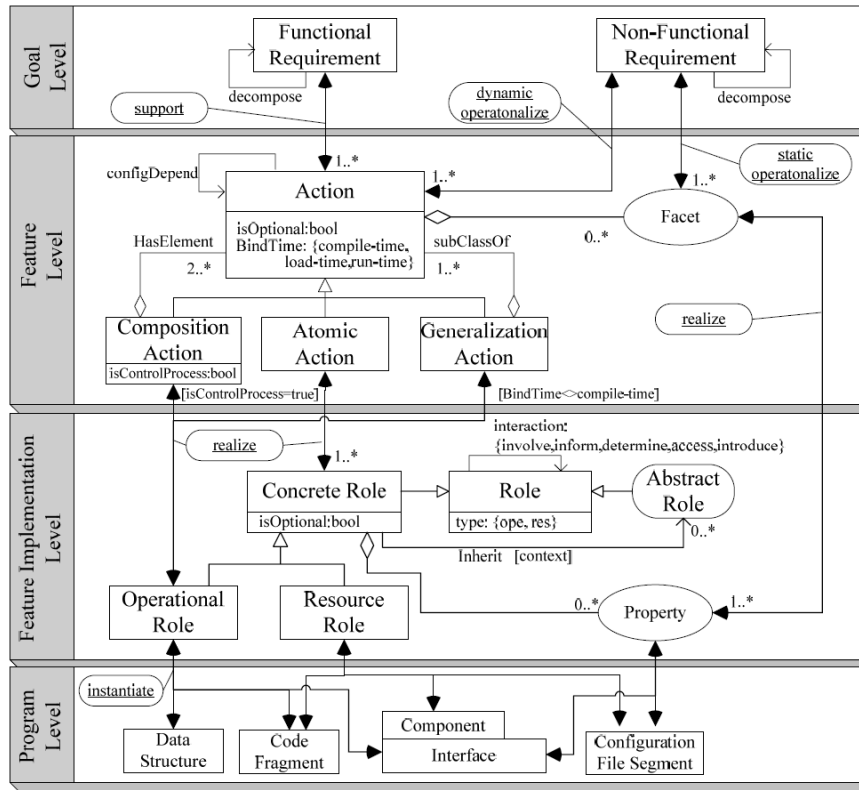


FIGURE 17 – Méta-modèle de trace orienté caractéristique [Shen et al., 2009]

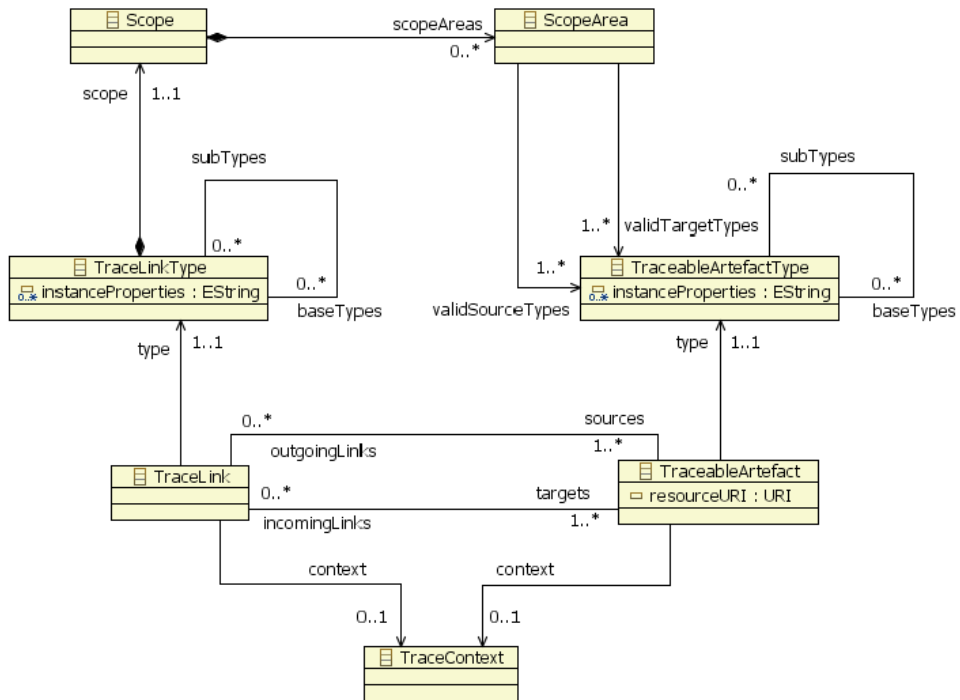


FIGURE 18 – Méta-modèle ATF de traçabilité [Anquetil et al., 2010]

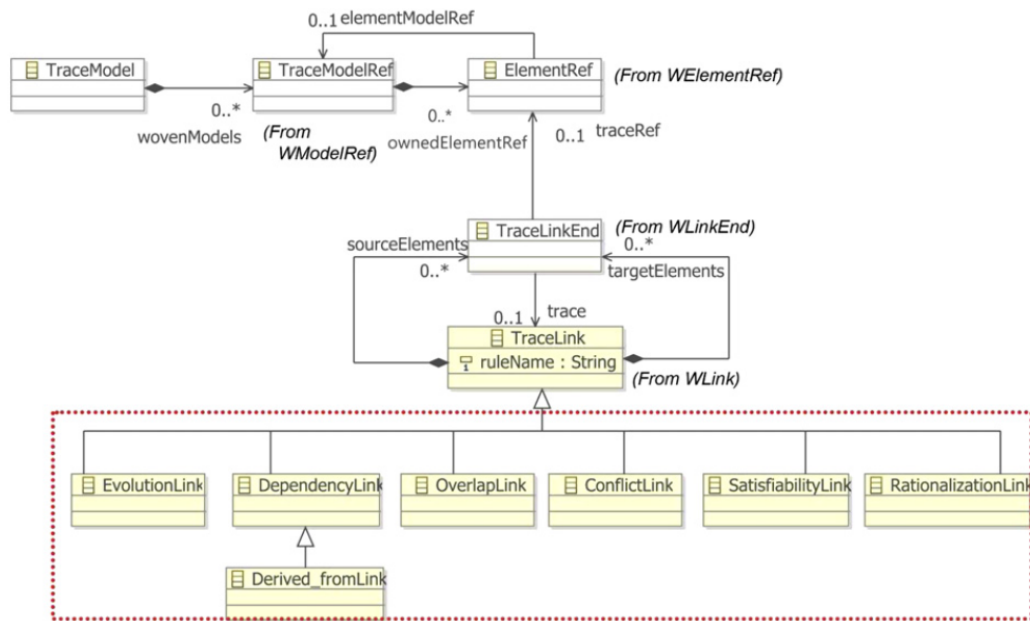


FIGURE 19 – Méta-modèle de traçabilité dédié au développement des entrepôts de données [Maté and Trujillo, 2012]

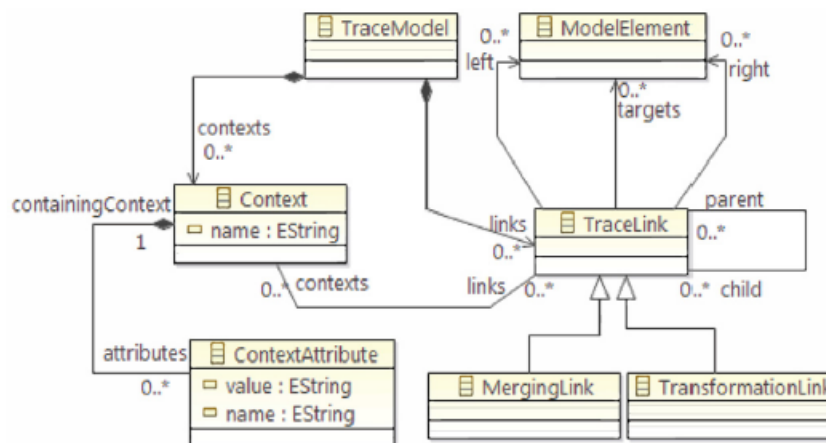


FIGURE 20 – Méta-modèle de traçabilité des compositions de modèles [Laghuaouta et al., 2017]

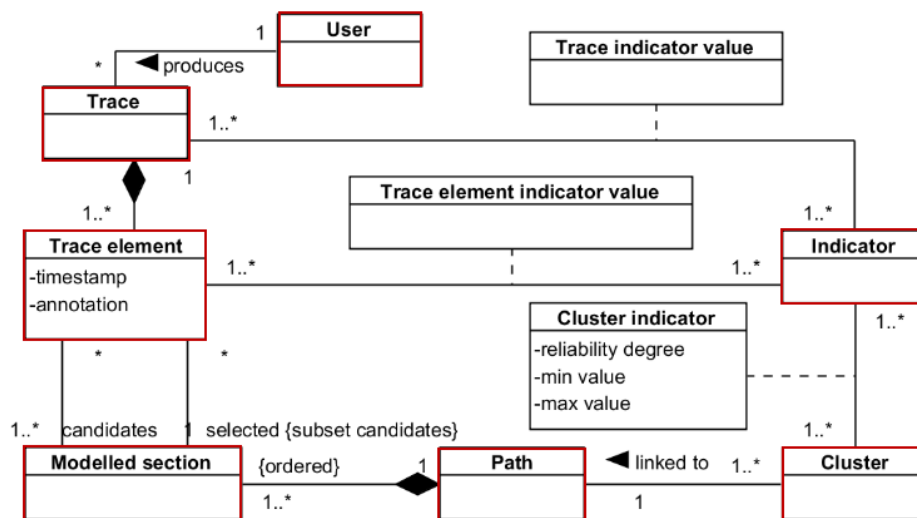


FIGURE 21 – Méta-modèle de trace MAP [Hug et al., 2012]

Liste des notes

1. Il est vrai que dans le cas présent, les articles scientifiques sont très majoritairement lus et synthétisés pour comparer son travail aux ceux des autres équipes de recherche, mais un manuscrit de thèse reste toujours la source la plus précieuse pour comprendre plus en détail le contexte, les objectifs et les contributions, entre autres, des autres travaux. (Page 1).
2. Dans ce travail de recherche, nous nous intéressons plus particulièrement aux processus de conception et de supervision des systèmes, étant les plus critiques du processus de développement produit (voir la Section 2.2.3 pour plus de détails). (Page 2).
3. <http://web.mit.edu/> (Page 10).
4. <https://www.linkedin.com/in/chrisccialosi/> (Page 13).
5. <https://www.forbes.com/sites/chrisccialosi/2015/06/15/how-exceptional-companies-create-high-performance-cultures/#5acad2445fd1> (Page 13).
6. Magazine économique américain qui s'intéresse au business, investissement, technologie, entrepreneuriat, leadership et au style de vie. (Page 11).
7. <https://fr.wikipedia.org/wiki/H%C3%A9raclite> (Page 13).
8. <https://en.oxforddictionaries.com/> (Page 11).
9. Session de formation sur les meilleures pratiques en développement de produits : <https://www.economie.gouv.qc.ca/bibliotheques/formations/formations-mpa-les-meilleures-pratiques-daffaires/meilleures-pratiques-en-developpement-de-produits/> (Page 12).
10. <https://www.lkab.com/en/> (Page 13).
11. <https://www.weforum.org/> (Page 13).
12. <http://corporate.ford.com/homepage.html> (Page 13).
13. <https://www.forbes.com/sites/joannmuller/2011/12/25/the-worlds-most-popular-cars-are-changing/#71b5eb14180e> (Page 13).
14. <https://www.linkedin.com/pulse/what-difference-between-innovation-product-lucy-maclennan/> (Page 14).
15. Produit immatériel, par exemple, le suivi 24h/24 et mise en sécurité des réseaux d'exploitation de gaz ou encore l'exemple des logiciels SaaS - Software as a Service - installés sur des serveurs distants plutôt que sur la machine de l'utilisateur. Ce dernier s'abonne au service et ne paie pas la licence d'utilisation de ces logiciels. (Page 16).
16. Ensemble d'éléments matériels et immatériels interagissant entre eux pour produire un produit ou un service [Bellagamba, 2012], par exemple, le système de production. (Page 16).
17. <http://www.pdma.org/> (Page 17).
18. Attention à la différence sémantique entre concourante et concurrente (Page 17).
19. <http://www.toyota-global.com/> (Page 17).
20. Comme l'application BPR (Business Process Re-Engineering), développée par Ford Motor, ayant pour objectif de supporter le processus manuel existant visant le partage des améliorations de processus entre les usines d'assemblage de véhicules [Rao, 2012]. (Page 18).
21. <https://www.bang-olufsen.com/fr> (Page 22).
22. <https://www.gartner.com/technology/home.jsp> (Page 23).
23. <https://fr.wikipedia.org/wiki/Aristote> (Page 27).
24. <https://fr.wikipedia.org/wiki/Platon> (Page 27).
25. [https://fr.wikipedia.org/wiki/Francis_Bacon_\(philosophe\)](https://fr.wikipedia.org/wiki/Francis_Bacon_(philosophe)) (Page 27).
26. https://en.wikipedia.org/wiki/Ren%C3%A9_Descartes (Page 28).
27. C'est une théorie qui montre comment les décideurs choisissent les alternatives qui impliquent à la fois le risque et l'incertitude. (Page 34).
28. C'est une théorie qui incorpore le regret, qui pourrait se produire en prenant une mauvaise décision, dans la phase de choix de l'alternative pour l'éliminer ou au moins réduire sa probabilité. (Page 34).
29. Il s'agit de tout système pouvant s'écrire sous la forme d'une équation matricielle : $AX = K$ où M est la matrice des coefficients, Y est le vecteur des constantes et X est le vecteur des inconnus. (Page 41).
30. C'est un ex scientifique chercheur en sciences informatiques à l'université de Stanford : <http://jmc.stanford.edu/> (Page 44).

31. PROgrammation en LOGique, c'est un langage de programmation logique : <https://en.wikipedia.org/wiki/Prolog> (Page 47).
32. <https://www.ibm.com/us-en/> (Page 55).
33. https://www.sas.com/fr_fr/home.html (Page 57).
34. Un évènement est défini, selon le SAS [2015], comme étant quelque chose qui se produit à un temps identifiable et qui peut être capturée. (Page 58).
35. C'est une structure ou un motif fréquemment rencontré lors de l'étude des données. Ce concept est détaillé dans les chapitres suivants (Page 58).
36. <https://www.capterra.com/decision-support-software/> (Page 58).
37. <http://www.experian.com/strategy-management/powercurve.html> (Page 59).
38. <https://www.tibco.com/solutions/systems-of-insight> (Page 59).
39. Classement des constructeurs de Formule 1 : <https://www.lequipe.fr/Formule1/f1-classement-ecuries.html> (Page 59).
40. <https://products.office.com/en/excel> (Page 59).
41. C'est un outil intégré dans Excel, qui se voit comme un programme codé en utilisant le langage de programmation VBA (Visual Basic for Applications), visant à automatiser un ensemble de décisions en se basant sur certaines conditions (Page 59).
42. C'est une partie de logiciel permettant de résoudre un problème mathématique donné. (Page 59).
43. <https://www.solver.com/products-overview> (Page 59).
44. En effet, la sélection d'une mauvaise alternative aura un impact majeur sur la qualité du produit, le temps et le coût de son développement, etc. (Page 61).
45. <https://www.nasa.gov/> (Page 63).
46. <https://history.nasa.gov/Apollo204/content.html> (Page 63).
47. <https://history.nasa.gov/rogersrep/genindex.htm> (Page 63).
48. <https://en.wikipedia.org/wiki/Thiokol> (Page 63).
49. <https://history.nasa.gov/columbia/CAIB.html> (Page 63).
50. https://en.wikipedia.org/wiki/Michael_D._Griffin (Page 63).
51. https://en.wikipedia.org/wiki/US_Airways_Flight_1549 (Page 68).
52. Ancien pilote américain et expert en sécurité aérienne : https://en.wikipedia.org/wiki/Chesley_Sullenberger (Page 68).
53. Agence chargée des réglementations de tous les aspects de l'aviation civile aux États-Unis : <https://www.faa.gov/> (Page 68).
54. National Transportation Safety Board (Conseil national de la sécurité des transports en français) est une agence responsable des enquêtes sur les accidents de transport aux États-Unis : <https://www.ntsb.gov/Pages/default.aspx> (Page 69).
55. <https://fr.wikipedia.org/wiki/Lufthansa> (Page 70).
56. <http://www.dlr.de/dlr/en/desktopdefault.aspx/tabid-10002/> (Page 70).
57. <https://www.nrc.gov/> (Page 70).
58. C'est une loi qui déclare que le développement, l'utilisation et le contrôle de l'énergie atomique doivent être orientés de manière à promouvoir la paix dans le monde, améliorer le bien être général, améliorer le niveau de vie et renforcer la libre concurrence entre les entreprises privées : <https://www.nrc.gov/about-nrc/governing-laws.html> (Page 70).
59. C'est une approche qui décrit un système comme étant un ensemble indivisible et le traite d'une manière globale non disjointe pour assurer un bon niveau de cohérence (Page 71).
60. Bien que les notions "objectif" et "but" sont définis de la même façon par les dictionnaires, il existe une grande différence du point de vue business. Un but est l'ensemble des déclarations décrivant les aspirations pour l'avenir d'une entreprise, tandis qu'un objectif est l'ensemble des étapes concrètes que l'entreprise doit suivre pour réaliser ses buts (Page 71).
61. <https://www.schneider-electric.fr/fr/> (Page 73).
62. Bien que les notions de justesse (accuracy) et de précision (precision) sont souvent interchangeable, la différence sémantique est loin d'être négligeable. La justesse fait référence à la proximité d'une valeur mesurée par rapport à une valeur standard et/ou connue. En revanche, la précision est la proximité de deux mesures ou plus, les unes aux autres. Si par exemple, les mesures sont proches de la valeur standard, mais sont très loin les unes des autres ; nous pouvons dire qu'elles sont justes mais imprécises [Westgard and Lott, 1981] (Page 74).
63. <https://www.incose.org/> (Page 75).
64. C'est la version Française de l'INCOSE : <http://www.afis.fr> (Page 75).
65. <https://www.sixsigmadaily.com/input-output-model/> (Page 75).

66. <http://www.bkcase.org/> (Page 76).
67. <https://www.computer.org/> (Page 76).
68. <http://www.sercuarc.org/> (Page 76).
69. <http://www.omg.org/> (Page 78).
70. Ce sont des réseaux constitués d'ordinateurs autonomes connectés à l'aide d'un intergiciel de distribution. Ils permettent de partager différentes ressources et capacités pour fournir aux utilisateurs un réseau cohérent unique et intégré : https://en.wikipedia.org/wiki/Distributed_computing (Page 78).
71. Dans un système orienté objet, les données sont représentées par des objets réutilisables avec lesquels l'utilisateur et les autres objets du système peuvent interagir. Ce type de systèmes permet à l'utilisateur de se concentrer sur les tâches plutôt que sur les outils d'implémentation (vu qu'il existe plusieurs langages et outils permettant d'implémenter les systèmes adoptant une approche orientée objet) : [https://en.wikipedia.org/wiki/Object_\(computer_science\)](https://en.wikipedia.org/wiki/Object_(computer_science)) (Page 78).
72. Toutes les normes OMG sont disponibles sur : <http://www.omg.org/spec> (Page 78).
73. <http://www.decisionmanagementsolutions.com/> (Page 78).
74. C'est un langage définissant le format d'échange de données textuelles, il est compris à la fois par l'homme et la machine : <https://www.w3.org/XML/> (Page 80).
75. <http://www.iiba.org/> (Page 81).
76. <http://www.iso.org/> (Page 82).
77. L'internet des objets connectés (Internet of Things - IoT) fait référence aux dispositifs qui collectent et partagent les données via internet sans interaction de type homme-homme ou homme-machine [Ashton et al., 2009] (Page 88).
78. C'est une transformation cyber-physique de la fabrication, visant à promouvoir la fabrication connectée et la convergence numérique entre l'industrie, les entreprises et d'autres processus [Lasi et al., 2014] (Page 88).
79. Attention à la différence sémantique entre un point de vue et une vue. Le point de vue est un partitionnement du système (parties physique et logique par exemple), tandis que la vue est la représentation (projection) du système selon le point de vue considéré [ISO, 2011] (Page 88).
80. C'est un langage de méta-modélisation basé sur la notion de classe, association et héritage : <https://www.omg.org/spec/MOF/About-MOF/> (Page 95).
81. Ce sont les périphériques ou les points de données constituant un point d'intersection/connexion au sein d'un réseau donné (Page 95).
82. <https://www.eclipse.org/> (Page 95).
83. C'est un langage de programmation informatique orienté-objet basé sur les classes : <https://www.java.com/en/> (Page 96).
84. <http://www.promtools.org/doku.php> (Page 96).
85. <https://www.cs.waikato.ac.nz/ml/weka/> (Page 96).
86. <https://www.eclipse.org/papyrus/> (Page 97).
87. Laboratoire d'Ingénierie dirigée par les modèles pour les Systèmes Embarqués (Page 97).
88. Commissariat à l'Énergie Atomique et aux Énergies alternatives-Laboratory for Integration of Systems and Technology : <http://www-list.cea.fr/en/> (Page 97).
89. Un élément est dit source s'il émet le lien, et est dit cible s'il reçoit le lien. En d'autres mots, la flèche du lien se dirige toujours vers l'élément cible. (Page 97).
90. https://www.nobelprize.org/nobel_prizes/physics/laureates/1932/heisenberg-bio.html (Page 103).
91. https://www.nobelprize.org/nobel_prizes/physics/laureates/1921/einstein-bio.html (Page 103).
92. Il est important de noter que la notion d'entreprise globale est différente de celle d'entreprise multinationale. Les deux types d'entreprise sont présents et investissent dans de nombreux pays, la différence majeure est que le premier type est plus axé sur l'adaptation de ses produits/service à chaque marché local, tandis que le deuxième commercialise ses produits en utilisant la même image/marque sur tous les marchés. Le lecteur peut se référer au travail de Azevedo and Bertrand [2000] pour plus de détails. (Page 104).
93. <https://scholar.google.fr/> (Page 107).
94. C'est un ensemble fini de données produit (Page 108).
95. Ce concept est défini par le Cambridge Dictionary comme : "seeming to be everywhere". En effet, ce sont des systèmes omniprésents intégrés dans l'environnement et dans les objets de tous les jours, au lieu de n'être utilisés que sur des ordinateurs, permettant ainsi une interaction plus naturelle et causale [Carvalho et al., 2003]. Le produit "Apple watch" est un exemple d'une montre basée sur la technologie omniprésente. (Page 109).
96. <http://www.eclipse.org/at1/> (Page 110).
97. En utilisant des techniques de satisfaction des contraintes comme la technique CSP (Constraint Satisfaction Problem) [Kumar, 1992], entre autres. (Page 114).

98. <https://www.mattel.com/en-us> (Page 120).
99. <https://money.cnn.com/2009/06/05/news/companies/cpsc/> (Page 121).
100. <https://www.forbes.com/sites/aarontilley/2017/01/22/samsung-note-7-fire-investigation-battery/#4fbc8f442017> (Page 121).
101. Nous appelons déviation l'écart entre le produit fabriqué et le produit souhaité (Page 121).
102. C'est une science traitant la quête de l'information dans un document, ou la quête des documents, des images, etc. (Page 124).
103. C'est une solution open source basée sur le web : http://www.prowhy.org/prowhy_site/ (Page 124).
104. Plusieurs autres appellations existent dans la littérature, nous citons par exemple : les systèmes intelligents de décision (IDS - Intelligent Decision Systems), les systèmes d'aide à la décision basés sur les connaissances (KBDSS - Knowledge-Based Decision Support Systems), les systèmes d'aide à la décision actifs (Active DSSs) et les systèmes cognitifs conjoints (Join Cognitive Systems) [Phillips-Wren and Ichalkaranje, 2008]. (Page 129).
105. L'adjectif "prétendue" est utilisé ici parce que les décisions venant d'être prises (après une éventuelle analyse décisionnelle) sont jugées être bonnes. Toutefois, l'évaluation objective n'aura lieu qu'après l'implémentation de ces décisions et l'étude de leur performance. (Page 129).
106. Ou patron, c'est une exécution possible et complète du processus, de son début à sa fin. Ce concept sera discuté plus en détails dans les sections suivantes. (Page 130).
107. Plusieurs standards et approches existent, nous avons cité deux pour justifier le choix de ne pas imposer un protocole de communication. (Page 132).
108. C'est un système logiciel permettant de supporter la communication machine-machine sur un réseau informatique. En d'autres termes, une machine peut appeler, sans intervention humaine, les services offerts par d'autres machines via internet et indépendamment des plates-formes et des langages utilisés [Ferris and Farrell, 2003]. (Page 132).
109. Ils permettent d'établir un canal de communication bidirectionnel entre deux machines du réseau. Une des deux machines sera le serveur qui, après avoir accepté la demande de connexion, écoute la demande de l'autre machine appelée client et fournit la réponse appropriée [Stevens et al., 2004]. (Page 132).
110. Deux grandes parties se distinguent quand il s'agit de comparer la fouille de données et l'apprentissage automatique. La première voit que ce dernier est une des techniques utilisées pour fouiller dans les données, tandis que la deuxième équipe assure que l'apprentissage automatique utilise les techniques de la fouille de données pour construire un modèle permettant la prédiction des situations futures. Nous partageons l'avis de la première équipe puisque la fouille de données, ayant été introduite en 1930 sous le nom de "Knowledge discovery", se voit plus ancienne que l'apprentissage automatique ayant été introduit en 1950 [Import.io, 2017]. De ce fait, nous considérons que l'apprentissage automatique est une des techniques utilisées par la fouille de données, comme décrit dans la Figure 7.5.b. (Page 132).
111. Noter que l'existence de l'élément $\langle dg \rangle$ dans la séquence S_4 du Tableau 7.4 ne pose pas de problème si nous cherchons des patterns qui ne sont pas nécessairement contigus. (Page 136).
112. Car l'existence du label dépend de la nature du jeu de données et du type d'apprentissage. En effet, le label existe lorsqu'il s'agit de l'apprentissage supervisé ou semi-supervisé, il est en revanche non présent lorsqu'il s'agit de l'apprentissage non-supervisé. (Page 137).
113. C'est un jeu de société d'origine indienne : <http://www.ludoteka.com/parcheesi.html> (Page 142).
114. Nous citons comme exemple de PAISs, les systèmes ERP (Enterprise Resource Planning) comme le SAP, les systèmes CRM (Customer Relationship Management) comme Microsoft Dynamics CRM et les systèmes PDM (Product Data Management) comme Windchill, entre autres [van der Aalst and Gunther, 2007; van der Aalst et al., 2008]. (Page 143).
115. Ou Business Process Execution Language, c'est un langage basé sur le standard XML, qui permet aux services web d'interconnecter et de partager les données dans une architecture orientée services (Service-Oriented Architecture - SOA). (Page 144).
116. Cette référence n'est pas celle qui a premièrement introduit le concept, nous l'avons citée parce qu'elle définit ce dernier, étant sujet à plusieurs interprétations, d'une façon simple et pertinente. (Page 147).
117. Contrairement à Wu et al. [2013] qui voient la fouille des épisodes comme un ensemble totalement ordonné d'activités, et conformément à Manila et al. [1997] qui considèrent l'ordre partiel dans la définition. (Page 147).
118. Le calcul des indices de support "s" et de confiance "c" n'est pas rigoureux, les valeurs sont données à titre d'exemple. (Page 148).
119. Nous avons construit ce modèle de processus en adoptant BPMN comme langage de modélisation et en utilisant l'outil Signavio : <https://www.signavio.com/>. (Page 151).
120. En effet, notre exemple d'illustration considère toutes les jonctions possibles : And, Or et Xor comme décrit dans le Tableau 6.1, donnant ainsi lieu à plusieurs patterns dont chacun fait référence à un choix d'exécution possible. De plus, notre exemple illustratif considère la possibilité de la présence des mêmes activités dans des patterns différents (activités A_9 et A_{10} dans la Figure 7.22). (Page 151).

121. Les traces (ou patterns) du processus illustratif de la Figure 7.22 sont instanciées à partir du méta-modèle de trace, dans l'environnement Eclipse (Section 6.2) (Page 151).
122. Il est aussi possible que la variable de contexte concernant la ressource humaine ne soit pas son nom mais plutôt sa compétence principale ou son rôle, entre autres. (Page 151).
123. "∧" est le "et" logique : la condition ($a \wedge b$) est vraie si et seulement si a est vrai et b est vrai. "∨" est le "ou" logique : ($a \vee b$) est vraie si un des deux paramètres est vrai. "==" s'utilise pour vérifier l'égalité contrairement à "=" qui s'utilise pour affecter une variable à une autre. ($a \neq b$) revient à dire que $a \neq b$. (Page 152).
124. Il est important de noter la différence entre un verrou scientifique et un verrou technologique. Le premier est lié à une limite conceptuelle (comme l'absence d'une méthode permettant de résoudre une problématique donnée), tandis que le deuxième est lié à une limite technologique (comme l'absence d'une application permettant l'implémentation de la méthode identifiée.) (Page 154).
125. eXtensible Markup Language : <https://www.w3.org/TR/xml/> (Page 154).
126. <http://www.processmining.org/logs/mxml> (Page 154).
127. <http://www.xes-standard.org/> (Page 154).
128. <https://standards.ieee.org/about/sasb/index.html> (Page 154).
129. Comme exemple de traces du processus "Préparer une tarte au chocolat", nous citons : "Préparer une tarte ronde au chocolat et à trois couches en suivant la méthode de Widad la pâtissière" et "Préparer une tarte carrée au chocolat". (Page 154).
130. <http://www.xes-standard.org/lifecycle.xesext> (Page 154).
131. La transformation permet le passage d'un modèle conforme à un certain méta-modèle "x", à un autre modèle conforme au même méta-modèle "x". En revanche, la projection permet de passer d'un modèle conforme à un certain méta-modèle "x", à un autre modèle conforme à un autre méta-modèle "y" [Bézivin, 2005]. (Page 156).
132. Model Driven Architecture, c'est une approche de développement des systèmes logiciels, proposée par l'OMG (<https://www.omg.org/mda/>). La différence entre la MDA et la MDE est bien discutée ici : <https://modeling-languages.com/clarifying-concepts-mbe-vs-mde-vs-mdd-vs-mda/> (Page 156).
133. ou XML Schema Definition, il permet de représenter l'inter-relation entre les attributs et les éléments d'un objet XML (comme un document par exemple) : https://www.w3schools.com/xml/schema_intro.asp. (Page 156).
134. Ce nombre paraît tout à fait acceptable pour tester la proposition et la précision des algorithmes de fouille de processus choisis. Cependant, pour découvrir les processus réels avec un niveau supérieur de certitude, il faut considérer un nombre de trace beaucoup plus important (13087 traces sont considérées pour étudier le processus de demande de prêt : <https://data.4tu.nl/repository/uuid:3926db30-f712-4394-aebc-75976070e91f>). (Page 158).
135. <https://www.my-invenio.com/> (Page 158).
136. <http://www.promtools.org/> (Page 158).
137. <https://fluxicon.com/disco/> (Page 158).
138. En effet, les nouveaux plugins développés peuvent être intégrés à l'environnement ProM. Cette intégration permet de combiner les plugins développés avec les autres plugins proposés par ProM. En outre, elle permet de visualiser, graphiquement, les résultats des plugins en choisissant une des perspectives proposées par ProM. (Page 158).
139. La conformité entre un modèle et un journal des événements est discutée dans Section 7.3.4.2. (Page 161).
140. Qui est introduit pour lever les limites de l'algorithme alpha. (Page 161).
141. Appelée aussi *Tau split*, c'est une activité fantôme qui permet d'exprimer la logique "Or" tout en permettant au modèle réseau de Petri d'être conforme à son méta-modèle. (Page 162).
142. Ce sont les propriétés ou les règles qui ne changent pas dans un journal des événements. (Page 165).
143. <http://sep.cs.ut.ee/Main/BranchMiner> (Page 165).
144. <http://www.cs.waikato.ac.nz/ml/weka/downloading.html> (Page 172).
145. Nous les avons construites à la main, aléatoirement sans suivre une logique définie pour ne pas impacter les résultats de la prédiction. (Page 173).
146. En effet, des études sur la fiabilité humaine ont montré que les personnes formées pour exécuter une activité donnée, commettent environ une à trois erreurs par heure dans les meilleurs scénarios [BKCASE, 2017]. (Page 183).
147. Ou Intelligent Decision-Making Support Systems, c'est une appellation équivalente à l'IDSS que nous utilisons. (Page 189).
148. En Anglais Data accuracy, il s'agit ici de vérifier que les données sont correctes (conformes à la réalité) et présentées d'une façon non-ambiguë : <http://etutorials.org/Misc/data+quality/>. (Page 191).
149. En Anglais : Data relevance. Il s'agit ici de vérifier que la donnée aidera à prendre la bonne décision : <http://leadsift.com/how-to-determine-what-data-is-relevant-and-what-is-distracting/>. Ceci est vérifié parce que les données (traces) que nous considérons sont issues du méta-modèle de trace qui met en évidence les données les plus intéressantes à capturer lors de l'exécution d'un processus. (Page 191).
150. https://europa.eu/european-union/topics/energy_fr (Page 194).
151. <https://www.gontrand.net/projet> (Page 195).
152. Le choix d'utiliser ce protocole est fait par les entreprises et laboratoires impliqués dans le projet. (Page 196).

MODÉLISATION ET FOUILLE DES PROCESSUS EN VUE D'ASSISTER LA PRISE DE DÉCISIONS DANS LE CONTEXTE DE LA CONCEPTION ET LA SUPERVISION DES SYSTÈMES

RÉSUMÉ : L'industrie actuelle est en plein développement suite à la mise en œuvre du concept de l'industrie 4.0 visant à rendre l'usine une entité intelligente, en combinant les processus et pratiques industriels avec les technologies de l'information et de communication récentes comme les systèmes cyber-physiques et l'internet des objets connectés (IoT), entre autres. Ce développement industriel, ainsi que le besoin d'innover pour atteindre et maintenir la compétitivité favorisent une augmentation phénoménale du volume de données (connue sous le nom de Big Data), rendant ainsi (i) les processus de conception et de supervision des systèmes de plus en plus complexes, (ii) les données au sein des organisations de plus en plus difficiles à exploiter et (iii) les ingénieurs de plus en plus perdus lors de la prise de décision. En effet, plusieurs problèmes sont identifiés dans les milieux industriels et qui peuvent être classés en trois catégories: (i) difficultés lors de la recherche, la visualisation et l'échange de l'information, (ii) difficultés lors de la prise de décision et (iii) difficultés lors de la gestion des changements de contexte. À travers ce travail, nous proposons un système d'aide à la décision intelligent et modulaire, dont chacun des quatre modules résout un des problèmes identifiés. Les modules de modélisation et de traçabilité des processus permettent de modéliser les processus et de capturer la façon comment ils sont exécutés. Le module d'aide à la décision permet de proposer le pattern le plus adapté au contexte de la décision ainsi que les paramètres de ses activités les plus significatifs. Le module de gestion des changements contextuels permet de continuellement mettre à jour le module de prise de décision, lui permettant ainsi de tenir compte de l'aspect dynamique du contexte. Le système proposé est entièrement vérifié et à moitié validé dans le contexte du projet Gontrand, visant la supervision intelligente et en temps réel des réseaux de gaz favorisant l'injection du gaz vert. Pour qu'il soit entièrement validé, les performances du système doivent être analysées après l'intégration et l'exploitation de ce dernier dans un milieu industriel réel.

Mots clés: Modélisation des processus, Traçabilité des processus, Aide à la décision, Gestion des changements contextuels, fouille des processus, Apprentissage automatique.

PROCESS MODELLING AND MINING TO SUPPORT DECISION MAKING IN THE CONTEXT OF SYSTEMS DESIGN AND SUPERVISION

ABSTRACT : Data sets are growing rapidly because of two things. First, the fourth industrial revolution that aims to transform factories into smart entities in which cyber physical systems monitor the physical processes of the factory. Second, the need to innovate in order to achieve and maintain competitiveness. Due to this huge volume of data (Big Data), (i) design and supervision processes tend to become more complex, (ii) data within organizations is increasingly becoming difficult to exploit and (iii) engineers are increasingly lost when making decisions. Indeed, several issues are identified in industry: (i) when researching, visualizing and exchanging information, (ii) when making decisions and (iii) when managing contextual changes. Through this research work, we propose an Intelligent and modular Decision Support System (IDSS), where each of the four modules solves one of the identified issues. Process modelling and traceability modules aim to model processes and capture how they are actually executed. The decision support module proposes the process patterns that best fit the decision context, as well as their most significant activity parameters. The contextual change management module continuously updates the decision-making module, in order to handle the dynamic aspect of the decision context. The proposed system is fully verified and half-validated in the context of the Gontrand project, aiming at intelligent and real-time supervision of gas networks favoring the injection of green gas. In order to be fully validated, the performance of the system must be analyzed after integrating and exploiting it in a real industrial environment.

Keywords: Process modelling, Traceability, decision-making support, contextual change management, process mining, machine learning.

